Journal of *Marine Science and Engineering*

MDPI

Article

TriangularSORT: A Deep Learning Approach for Ship Wake Detection and Tracking

Chengcheng Yu and Yanmei Zhang *

School of Integrated Circuits and Electronic Engineering, Beijing Institute of Technology, Beijing 100081, China; 18810995003@163.com

* Correspondence: zhangym0726@163.com

Abstract: Ship wake detection and tracking are of paramount importance for ensuring maritime safety, conducting effective ocean monitoring, and managing maritime affairs, among other critical applications. This paper introduces a novel approach for ship tracking and wake detection utilizing advanced computational techniques, particularly the TriangularSORT algorithm for monitoring vessels. This method enhances effective ship tracking by closely associating the vertices of the triangular wake with the coordinates of the ship. Furthermore, this paper integrates the triangular IoU and attention mechanism, introducing the Triangular Attention Mechanism. This mechanism guides the model's focus to key areas of the image by defining triangular points on the feature map, thereby enhancing the model's ability to recognize and analyze local features in visual tasks. Experimental results demonstrate that the proposed method significantly improves the performance and accuracy of models in object detection and tracking tasks.

Keywords: ship tracking; wake detection; deep learning; triangular IoU

check for updates

Academic Editor: Rafael Morales

Received: 28 October 2024
Revised: 29 November 2024
Accepted: 5 December 2024
Published: 8 January 2025

Citation: Yu, C.; Zhang, Y. TriangularSORT: A Deep Learning Approach for Ship Wake Detection and Tracking . *J. Mar. Sci. Eng.* **2025**, *13*, 108. https://doi.org/10.3390/jmse13010108

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

Ship wake detection and tracking are critical for ensuring maritime safety, conducting effective ocean monitoring, managing maritime affairs, ensuring national defense and military security, carrying out search and rescue operations, and advancing scientific research. Ship tracking technology helps to monitor maritime traffic, prevent collisions and accidents, and protect the safety of crew members and cargo. In ocean monitoring, it helps to oversee vessel activities at sea and is vital for preventing illegal fishing, smuggling, and other maritime crimes. Moreover, ship tracking plays a significant role in maritime management, assisting authorities in regulating sea traffic, optimizing shipping routes, and enhancing efficiency. In the domain of national defense and military security, this technology aids in monitoring and controlling territorial waters, safeguarding national security, and promptly identifying and responding to potential threats. During search and rescue missions, ship tracking technology can rapidly locate distressed vessels or individuals at sea, thereby improving the efficiency of rescue operations, and is essential for reducing loss of life and property in maritime disasters. Finally, ship tracking technology provides a wealth of data for marine scientific research, aiding in the study of changes in the marine environment, marine biodiversity, and the sustainable use of marine resources. As technology progresses, the application of ship tracking technology in these fields is anticipated to become more extensive and profound.

Compared to vessels themselves, the wakes left by ships cover a significantly larger area in satellite imagery. This characteristic enables detection under conditions that require

lower camera altitude and resolution, making it more accessible and practical for a variety of monitoring applications. However, the pursuit of ship wake tracking is riddled with challenges such as trajectory overlap, data deficiency, and the dynamics of various environmental changes. In heavily trafficked sea areas, the wakes of multiple vessels may intersect and overlap, complicating the process of differentiation. The limitations of monitoring equipment along with environmental factors such as adverse weather conditions can lead to incomplete data. The fluctuating marine environment, including changes in currents and wind directions, also impacts tracking of wakes. Variations in a ship's speed and heading contribute to the complexity and variability of wake patterns. Furthermore, sensor noise and rapid dispersion of wakes in the ocean increase the difficulty of tracking. Different lighting and imaging conditions such as nighttime and cloudy weather amplify the difficulty of tracking wakes through satellite or aerial imagery. Moreover, sophisticated data processing and analytical techniques are required for automatic detection and discrimination of wakes. At times, vessels may deliberately take measures to conceal their wakes in order to protect privacy or increase security, adding an extra layer of challenge to the tracking process.

Existing methods for ship tracking face certain challenges and limitations when dealing with spatial and temporal variations. In sea areas with dense ship traffic, such as ports or confluences of shipping lanes, distinguishing and tracking the trajectory of each vessel becomes highly complex. The dynamics of the maritime environment, including changes in ocean currents, wind speed, and wind direction, can affect the movement of ships, and traditional algorithms may fail to accurately anticipate these fluctuations. The collected data may be incomplete or noisy due to weather conditions or sensor malfunction, affecting the continuity and accuracy of tracking. Variations in the size and shape of ships can impact the detection capabilities of algorithms as well. Ships may be obscured by other vessels, islands, or coastlines, leading to interruptions in the tracking process. Traditional tracking algorithms often rely on simple prediction models that may not capture the complex patterns of ship movement. As the number of tracking targets increases, the computational burden on the algorithm increases significantly, especially in applications requiring real-time processing. Some algorithms may perform well on specific datasets but fail to function effectively in new or unknown environments. Furthermore, ship tracking systems must operate while protecting privacy and ensuring data security, which may constrain the design and implementation of algorithms.

Trajectory correlation in maritime surveillance is instrumental in linking consecutive observations of maritime targets into comprehensive trajectories. This process aids in the accurate identification and classification of targets, mitigates false alarms, anticipates target behaviors, and optimizes the allocation of surveillance resources. In addition, it enhances tracking precision, bolsters situational awareness, supports compliance checks, and aids in search and rescue operations. Moreover, trajectory correlation plays a crucial role in environmental protection by monitoring illegal discharge activities. These functionalities are realized through technologies such as data fusion, pattern recognition, and machine learning, thereby enhancing the efficiency and effectiveness of maritime surveillance systems.

2. Related Works

The luminous pixels of ship targets are prone to interference from terrestrial harbors, oceanic islands, and other artificial structures. Furthermore, the patterns of ship wakes often blend with linear structural features such as coastlines, oil spills, and intra-oceanic waves, making them indistinguishable. Such interference significantly undermines the robustness of many conventional ship and wake detection algorithms; hence, there is an urgent need to develop automated techniques that ensure efficient and accurate detection. Despite extensive research and demonstrations of powerful feature learning capabilities by various methods,

it is notable that the focus has primarily been on ship detection rather than wake detection. Researchers including Wang et al. [1], Wei et al. [2], Zhang et al. [3], Zhang et al. [4], and Ding et al. [5] have each generated distinct datasets of remotely sensed ships which serve as a foundation for ship target detection studies [6]. Zhang et al. [7], Shao et al. [8], and Li et al. [9] have explored algorithmic models for ship detection, aiming to construct more sophisticated deep learning networks for this purpose. In the realm of wake detection, Kang and Kim [10] have utilized convolutional neural networks (CNNs) to discern ships and wakes from Synthetic Aperture Radar (SAR) imagery, achieving relatively precise velocity estimations. Although their study was based on a limited sample of 200 ship and wake instances, the results were promising. Del Prete et al. have developed a novel dataset comprising 250 Sentinel-1 wake images for evaluating wake detection with various standard CNNs, confirming the efficacy of CNNs in this domain despite the dataset's constraints [11]. Additionally, there is ongoing research into simulating ship wake data, which is pivotal for enabling deep learning-based wake detection and ship classification tasks [12]. Xue et al. have created the SWIM dataset, an optical remote sensing collection consisting of 15,356 ship wake instances, and proposed WakeNet, which integrates attention to the frequency channel along with the wake's frequency characteristics to highlight the capability of frequency extraction in wake detection features [13]. Ding et al. manually curated and labeled Gaofen-3 images to assemble a SAR remote sensing dataset encompassing 862 ship and wake pairs [14]. Through the observation and documentation of a ship's wake, valuable insights into the ship's position, velocity, trajectory, and conduct can be gleaned. Analyzing the wake's attributes and patterns allows inferences to be made regarding the ship's category, dimensions, functionality, propulsion system, maneuverability, and hydrodynamics. Moreover, it aids in assessing and monitoring the environmental impact of maritime activities, including the discharge of wastewater and oil pollution [15,16].

The goal of target detection is to anticipate a series of bounding boxes and categorical labels for each object of interest. Modern detectors address this challenge indirectly by formulating regression and classification problems for numerous proposals [17], anchor boxes [18], or window centers [19,20]. The effectiveness of these detectors is substantially influenced by postprocessing procedures, anchor box design, and heuristics for associating target boxes with anchor boxes. To streamline these processes, Carion et al. introduced a direct ensemble prediction approach known as DEtection with TRansformers (DETR), which obviates the need for intermediary agent tasks [21]. DETR simplifies the training process by treating target detection as a straightforward ensemble prediction task, and employs an encoder–decoder architecture based on transformers [22]. Such architectures explicitly model interactions between sequence elements through the transformer's self-attention mechanism, making them particularly well suited for handling ensemble prediction-specific constraints such as avoiding duplicate predictions. However, DETR faces challenges due to the limitations of the transformer's attention module when dealing with image feature mapping, exhibiting slow convergence, limited feature space resolution, and unclear query interpretation. To address these issues, extensive research has focused on enhancing DETR's performance through innovations in pairwise attention, querying, positional embedding, and bipartite matching.

Due to the high computational complexity associated with attention mechanisms, especially when dealing with a large number of image pixels, Zhu and Dai et al. introduced Deformable DETR [23]. This method builds upon the concept of deformable convolution and implements deformable attention [24], merging the sparse spatial sampling of Deformable convolution with the relational modeling capabilities of transformers. This innovative approach addresses the issues of slow convergence and high complexity encountered with DETR, introducing multiscale deformable attention and various versions

of Deformable DETR to tackle the challenge of limited detection performance for small targets. Dai et al. introduced Dynamic DETR, which implicitly incorporates a coarse-to-fine box coding approach by introducing dynamic attention and stacking dynamic decoders at the codec stage [25]. To address the computational expense and slow convergence issues associated with DETR, Yao et al. proposed Efficient DETR [26], which improves target query initialization and reduces the number of required decoders. Furthermore, the number of proposals can be dynamically adjusted during the training process, further reducing DETR's computational demands. Zhang et al. presented DDQ DETR [27], which introduces dense distinguished queries into the framework to generate dense queries, which are then iteratively processed to produce the final sparse output. To tackle the issue of poor-quality content embedding in DETR, which hampers precise object localization, Meng et al. introduced Conditional DETR [28]. This approach decouples content information from location information in the original cross-attention, enabling the decoder to search for the target's end region on display, in turn improving object localization and reducing DETR's reliance on content embedding during training. To address the issue of unclear and ambiguous learnable embeddings in DETR, Wang et al. introduced Anchor DETR [29]. This approach draws inspiration from the concept of anchors in CNN target detectors, with each query in Anchor DETR based on a specific anchor, allowing queries to focus on targets located near their respective anchors. Liu et al. expanded on this approach by introducing DAB DETR, which adds scale information to the anchor points to make DETR's queries more interpretable [30]. This scaling data enhances query interpretability and acts as a positional guide for faster model convergence, while also adjusting the attention graph by incorporating scaling information related to bounding boxes. Sun et al. attributed the slow convergence of DETR to its use of the Hungarian loss applied to DETR's bipartite matches, arguing that noisy environments during different training periods lead to unstable matches. To address this, they introduced TSP-FCOS and TSP-RCNN [31]. These methods eliminate stochasticity by using a trained stationary teacher model, with the teacher model's predicted bipartite matches serving as ground truth labels. This approach helps to eliminate randomness in the training process. Li et al. first tackled the problem of unstable bipartite matching in DETR by conducting denoising training, leading to the development of DN DETR [32]. This approach not only accelerates model convergence but also significantly improves detection results. DN DETR focuses on predicting anchor boxes with nearby ground truths, but struggles when there are anchor boxes without nearby objects. To overcome this limitation, Zhang et al. merged DN DETR, DAB DETR, and Deformable DETR to create DINO [33]. DINO employs comparative denoising training to discard unnecessary anchor boxes, thereby enhancing overall performance. The notion of end-to-end target detection has garnered significant attention and research thanks to its simplified structure and efficient performance. Ongoing efforts to enhance model convergence and reduce computational complexity have highlighted the impressive capabilities and potential of the end-to-end target detection approach. Within this paradigm, research on oriented target detection methods is essential for addressing the challenges of DETR, including slow convergence and high computational requirements. This investigation holds immense importance for the field of oriented target detection.

Target tracking has evolved significantly with the advent of advanced computational techniques, becoming a cornerstone in the domain of computer vision and artificial intelligence [34]. Target tracking involves the localization and motion prediction of objects within a sequence of frames, which is crucial for a myriad of applications such as surveillance, autonomous driving, and human–computer interaction [35]. Over the past decades, research into target tracking has transitioned from traditional algorithms based on Kalman filters and means-shifting to more complex models that leverage machine learning and

deep learning [36]. These modern approaches have shown remarkable improvements in handling challenges such as occlusion, scale variations, and fast motion [37]. In particular, deep learning has revolutionized the way in which features are extracted and learned for tracking tasks. Convolutional neural networks (CNNs) have become instrumental in automatically capturing rich representations from data, which has led to significant enhancements in tracking accuracy and robustness [38]. Moreover, the integration of recurrent neural networks and attention mechanisms has further improved the ability of trackers to model complex object behaviors and interactions [39].

SORT, an acronym for Simple Online and Realtime Tracking, has emerged as a pivotal algorithm in the field of object tracking due to its efficiency and simplicity [40]. Since its inception, SORT has been widely adopted for applications where real-time performance is crucial, such as surveillance systems and autonomous vehicles. SORT operates on the principle of tracking-by-detection, utilizing a detection model to identify objects in each frame and a Kalman filter to predict the motion of these objects [41]. This approach allows SORT to handle multiple targets simultaneously while maintaining low computational overhead. Despite its effectiveness, SORT faces challenges in complex scenarios with heavy occlusion or rapid movements, where its reliance on detection accuracy becomes a limiting factor [42]. To address these issues, researchers have proposed enhancements to the SORT framework, including the incorporation of deep learning-based detectors and more sophisticated state estimation techniques [43]. The SORT algorithm also stands out for its ability to provide a strong baseline in multi-object tracking benchmarks, often serving as a starting point for more advanced algorithms that aim to improve tracking accuracy while preserving real-time capabilities [44]. As the field of object tracking continues to evolve, SORT remains a foundational algorithm, inspiring various extensions and improvements; the pursuit of more robust and adaptive tracking solutions is an ongoing endeavor, with SORT being a key reference in this dynamic research landscape.

Despite these advancements, there are numerous outstanding challenges that the research community aims to address. These include the need for real-time tracking capabilities, the generalization of trackers to diverse datasets, and the development of more efficient algorithms that can operate on resource-constrained platforms [45]. The pursuit of more accurate and efficient tracking algorithms is an active area of research, with ongoing efforts focused on improving the computational efficiency, adaptability, and robustness of tracking models [46]. As the field progresses, it is expected that future trackers will become more intelligent, requiring less manual intervention while providing more reliable tracking results in various scenarios [47].

In conclusion, ship wake tracking is a vital area of research with significant implications for maritime safety, environmental monitoring, and security. The integration of advanced machine learning techniques, particularly deep learning, has revolutionized the field, providing more accurate and efficient methods for tracking and analyzing ship wakes [48].

This paper introduces a novel approach to ship tracking that leverages the TriangularSORT algorithm for employing triangular wake tracks to monitor vessels. This method addresses the issue in traditional bounding box detection of the center of the rectangle lacking any specific physical meaning. By closely associating the vertices of the triangular wake with the coordinates of the ship, the proposed approach provides enhanced ship tracking ability. The method shifts the focus from rectangular bounding boxes to triangular tracks, which better represent the physical characteristics of a ship's wake. The vertices of the triangle are used to pinpoint the key points of the ship's movement, providing a more accurate and physically meaningful representation of the ship's trajectory. This innovative tracking technique offers several advantages over conventional rectangular bounding box methods, particularly in scenarios where the dynamics of the ship's wake

are critical to understanding its movement. The TriangularSORT algorithm is renowned for its efficiency in handling multiple targets in real-time tracking scenarios. By integrating this algorithm with our triangular wake-tracking approach, it is possible to significantly improve the precision and reliability of ship tracking. This integration allows for the effective management of complex maritime traffic in situations where multiple vessels and their wakes must be monitored simultaneously. In summary, the use of triangular wake tracks in conjunction with the TriangularSORT algorithm presents a promising advancement in the field of ship tracking. This method not only overcomes the limitations of traditional detection methods but also opens up new possibilities for enhanced maritime surveillance and vessel management.

3. The Principle of TriangularSORT

This section initially introduces detection-based tracking methods. These rely on independent object detection in each frame of the video and subsequent association of detection results to form continuous tracking trajectories. The rest of this section elaborates on the combination of DETR with Triangular Intersection over Union (IoU), an evaluation metric specifically designed to measure the similarity between predicted and actual triangular bounding boxes. Furthermore, we introduce the Triangular Attention Mechanism, which guides the model's focus to key areas of the image by defining triangular points on the feature map. This enhances the model's ability to recognize and analyze local features in visual tasks. Lastly, the design of the TriangularSORT multi-object tracking algorithm is described. This algorithm significantly improves the accuracy and robustness in handling multi-object tracking tasks through the integration of deep learning detectors, optimized feature extraction techniques, and sophisticated data association strategies. In addition, it incorporates the AFLink and GSI algorithms, with Triangle IoU serving as the matching evaluation metric. This section presents the methodological foundations of our proposal for achieving efficient and accurate ship tracking.

3.1. Detection-Based Tracking

Detection-based tracking is a method for tracking objects in video sequences. It relies on independent object detection in each frame and association detection results to form continuous tracking trajectories. The main steps of the method are as follows:

1. Initialization : Object detection is used to identify and locate objects in the initial frame of the video and assign a unique tracking ID to each object.
2. Frame-wise Detection: Object detection algorithms are run in each frame to identify and locate objects and to obtain an associated bounding box, category, and confidence level.
3. Data Association: The correspondence between the detection results of the current frame and the tracked objects in the previous frames is determined using the nearest neighbor, Kalman filter, or Hungarian algorithm method.
4. Tracking Maintenance: The state of the objects is updated to handle any object occlusion, loss, or reappearance and to merge or split closely adjacent objects.
5. Tracking Evaluation: Tracking performance is evaluated using metrics such as accuracy, robustness, and computational efficiency, with the parameters then adjusted based on the results.
6. Feedback and Adjustment: The detector parameters are adjusted based on tracking results to improve the model's accuracy and robustness.

Detection-based tracking methods have significant advantages in the field of video analysis. The simplicity and flexibility of this method makes it easy to adapt to various scenarios, including the sudden appearance and disappearance of objects. Because it

can independently utilize powerful object detection algorithms, it can handle complex backgrounds and changing environmental conditions. However, this method also has some disadvantages, mainly due to the need for full object detection in every frame of the video, which may lead to high computational costs and affect real-time performance. In addition, the stability and continuity of tracking may be affected if the accuracy of the detection algorithm is insufficient or if errors occur in frame-to-frame association. Therefore, it is necessary to improve the stability and continuity of the detector. In this paper, we propose a triangular IoU evaluation metric for DETR-based detection of triangular ship wakes. This not only improves the model's detection accuracy for triangular objects but also enhances the model's robustness in complex scenarios. At the same time, triangular detection boxes provide a more intuitive physical meaning than rectangular detection boxes during the subsequent tracking processes.

3.2. DETR with Triangular IoU

Detection with Transformer (DETR) is a cutting-edge object detection framework constructed upon the transformer architecture. DETR treats the task of object detection as a direct set prediction problem. Unlike traditional anchor-based methods, which rely on region proposal networks (RPNs) or anchors, DETR predicts a set of candidate objects' categories and locations without such dependencies. This approach theoretically allows DETR to sidestep the complexities and limitations found in conventional detection models.

3.2.1. Triangular IoU

The Triangular IoU is a variant of the IoU metric that is specifically designed to measure the similarity between predicted and ground-truth triangular bounding boxes. The IoU is a widely used metric in object detection for assessing the degree of overlap between predicted and actual bounding boxes. For triangular targets, including certain types of ship wakes, the traditional rectangular IoU may not adequately describe the shape and position of the target. Thus, the triangular IoU serves as an assessment indicator. This loss function measures the overlap between the predicted triangular bounding box and the actual triangular bounding box. By minimizing this loss, the model learns to accurately predict the position and size of triangular targets.

In summary, combining DETR with the triangular IoU offers a flexible and robust framework for detecting targets with complex shapes, such as triangular ship wakes. The advantage of this method is its ability to directly handle shape variations of targets within the model without the need for predefined anchors or region proposals.

Calculating the triangular IoU loss involves the following steps.

Given the following predicted and ground truth parameters of the bounding box and landmark:

- Predicted bounding box width and height $w_{\text{bbox}}^{\text{pred}}, h_{\text{bbox}}^{\text{pred}}$
- Predicted landmark coordinates and angles $x_{\text{mark}}^{\text{pred}}, y_{\text{mark}}^{\text{pred}}, \theta_1^{\text{pred}}, \theta_2^{\text{pred}}$
- Ground truth bounding box width and height $w_{\text{bbox}}^{\text{gt}}, h_{\text{bbox}}^{\text{gt}}$
- Ground truth landmark coordinates and angles $x_{\text{mark}}^{\text{gt}}, y_{\text{mark}}^{\text{gt}}, \theta_1^{\text{gt}}, \theta_2^{\text{gt}}$

first, calculate the lengths of the predicted and ground truth triangles by

$$\text{length}^{\text{pred}} = 0.5 \times \sqrt{w_{\text{bbox}}^{\text{pred}} + h_{\text{bbox}}^{\text{pred}}}, \tag{1}$$

$$\text{length}^{\text{gt}} = 0.5 \times \sqrt{w_{\text{bbox}}^{\text{gt}} + h_{\text{bbox}}^{\text{gt}}}. \tag{2}$$

Next, use the landmark angles and lengths to compute the coordinates of the triangle vertices for both the predicted and ground truth triangles by

$$\begin{bmatrix} x_1^{\text{pred}} & x_2^{\text{pred}} \\ y_1^{\text{pred}} & y_2^{\text{pred}} \end{bmatrix} = \begin{bmatrix} x^{\text{pred}} \\ y^{\text{pred}} \end{bmatrix} + \text{length}^{\text{pred}} \cdot \begin{bmatrix} \cos(\theta_1^{\text{pred}}) & \cos(\theta_2^{\text{pred}}) \\ \sin(\theta_1^{\text{pred}}) & \sin(\theta_2^{\text{pred}}), \end{bmatrix} \tag{3}$$

$$\begin{bmatrix} x_1^{\text{gt}} & x_2^{\text{gt}} \\ y_1^{\text{gt}} & y_2^{\text{gt}} \end{bmatrix} = \begin{bmatrix} x^{\text{gt}} \\ y^{\text{gt}} \end{bmatrix} + \text{length}^{\text{gt}} \cdot \begin{bmatrix} \cos(\theta_1^{\text{gt}}) & \cos(\theta_2^{\text{gt}}) \\ \sin(\theta_1^{\text{gt}}) & \sin(\theta_2^{\text{gt}}) \end{bmatrix}. \tag{4}$$

Form the oriented bounding boxes for both triangles:

$$obb^{\text{pred}} = [x_1^{\text{pred}}, y_1^{\text{pred}}, x_2^{\text{pred}}, y_2^{\text{pred}}, y_3^{\text{pred}}, y_3^{\text{pred}}, x_4^{\text{pred}}, y_4^{\text{pred}}], \tag{5}$$

$$obb^{\text{gt}} = [x_1^{\text{gt}}, y_1^{\text{gt}}, x_2^{\text{gt}}, y_2^{\text{gt}}, y_3^{\text{gt}}, y_3^{\text{gt}}, x_4^{\text{gt}}, y_4^{\text{gt}}]. \tag{6}$$

Finally, calculate the IoU loss as

$$L_{\text{Triangular IoU}} = 1 - \text{IoU}(obb^{\text{pred}}, obb^{\text{gt}}). \tag{7}$$

Minimizing the loss function based on the triangular IoU helps the DETR model to refine its predictions, ensuring that the detected triangular targets closely match their actual shapes and positions.

3.2.2. Triangular Attention

Attention mechanisms are computational approaches that mimic the human visual attention system, enabling models to focus on the most relevant parts when processing information. By assigning varying weights to different parts of the input data, these mechanisms allow models to selectively concentrate on important features amid a wealth of information. Attention mechanisms have been widely applied In natural language processing and computer vision to enhance model performance, especially when dealing with long sequences or complex imagery.

Unlike traditional neural networks, in which all input features are typically considered to have equal importance, attention mechanisms enable dynamic adjustment of focus based on context. Specifically, they calculate the similarity between queries, keys, and values to generate a weight distribution that indicates which input features are more significant for the current task. This process can be achieved through dot-product or other similarity metrics, followed by a softmax function to normalize the weights.

In practical applications, attention mechanisms have been shown to improve model performance, particularly in handling long-range dependencies; they enhance not only a model's expressive power but also its ability to capture long-distance relationships. In this way, attention mechanisms allow models to understand and generate information more effectively, propelling the success of transformers and similar architectures while leading to significant advancements across multiple tasks.

This paper integrates the previously proposed triangular IoU and a new triangular attention mechanism, introduced below. As shown in Figure 1, the triangular attention mechanism guides the model's focus to key areas of the image by defining triangular points on the feature map. The attention mechanism initially generates a set of triangular points using a query landmark guide, identifying local regions that require the model's special attention; then, it calculates sampling offsets for each point to extract local features related to these points from the feature map. Under the multihead attention framework, each head independently weights the local features to create attention weights. By subsequently aggregating these weighted features, the model's ability to capture key information is

enhanced. Ultimately, after integrating the outputs of different heads through a linear layer, the model produces the final response, achieving more accurate local feature recognition and analysis on visual tasks. This mechanism is particularly suitable for tasks that require precise localization, such as object detection and tracking, where it significantly improves model performance and accuracy.

The core concept of the proposed triangular attention mechanism is to integrate a deformable structure into the attention computation process to allow for more precise modeling of the feature regions. Given an input feature map $x \in \mathbb{R}^{c_i \times h_i \times w_i}$ and a 2D reference point p_{tri}, the triangular attention can be computed as follows:

$$\mathrm{TriAttn}(p_{tri}, \theta, x) = \sum_{j=1}^{J} W_j \left[\sum_{k=1}^{K} A_{jqk} \cdot W'_j x\Big(p_{tri} + \Delta p_{jqk}\Big) \cdot M(\alpha) \right] \tag{8}$$

where the rotation matrix is

$$M(\alpha) = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) \\ -\sin(\alpha) & \cos(\alpha) \end{bmatrix}, \tag{9}$$

J denotes the number of attention heads, $W'_j \in \mathbb{R}^{C_v \times C}$ and $W_j \in \mathbb{R}^{C \times C_v}$ are learnable weights (with $C_v = C/M$ by default), and the attention weights $A_{jqk} \propto \exp\left(\left\{\frac{z_q^T U_m^T V_m x_k}{\sqrt{C_v}}\right\}\right)$ are normalized to $\sum_{k \in \Omega_k} A_{mqk} = 1$. Here, $U_m, V_m \in \mathbb{R}^{C_v \times C}$ are learnable weights, Δr_{jqk} represents the sampling point offset, and z_q and x_k correspond to the concatenation/sum of element content embeddings and position embeddings, respectively.

Introduction of the triangular attention mechanism provides deep learning models with a novel tool for handling tasks involving sequential data, potentially advancing the field of sequential data processing. By understanding and applying this mechanism, it is possible to uncover additional model optimization strategies and enhance project performance.

3.3. The TriangularSORT Multi-Object Tracking Algorithm

In the realm of ship tracking, learning about a vessel's motion characteristics and potential intentions from historical movement data requires continuous surveillance of that particular vessel. Therefore, in this paper we construct a multi-object ship tracking model from a space-based perspective. The challenge of ship tracking lies in associating the same ship that appears in the field of view at any given moment with a unique identifier (ID). There are two main difficulties: matching moving targets, and avoiding ID switching phenomena when targets are occluded and reappear over time. The SORT algorithm and its variants effectively address both of these issues.

3.3.1. Target Tracking Design

Multi-object tracking involves assigning unique identifiers to multiple targets of interest, each labeled with distinct IDs such as 0, 1, 2, etc., and continuously tracking these identified targets throughout a video or image sequence. This process allows for the acquisition of temporal information related to the targets.

The typical tracking process generally encounters the following three scenarios:

1. Target Initialization: At the beginning of a video sequence or when a target first enters the field of view, the tracking algorithm must detect the target and initiate a new tracking trajectory. This typically involves detecting the presence of a target in the first frame or several consecutive frames and assigning it a unique identifier (ID).
2. Target Matching: In each frame, the algorithm must match newly detected targets with existing tracking trajectories. If an existing trajectory finds a matching target

in the current frame, the trajectory is considered to have successfully continued to track the target. The matching process may involve calculating the distance, similarity, or other relevant correlation measures between the detection box and the predicted trajectory.

3. Target Termination: When a target leaves the field of view, is occluded, or cannot be detected for other reasons, the tracking algorithm must recognize this situation and appropriately terminate or pause the corresponding tracking trajectory. This may involve setting a time threshold; if a trajectory is not detected for several consecutive frames, then the target is considered to have disappeared, and its corresponding trajectory can be terminated or marked as 'lost'.

The multi-object tracking model based on DeepSORT effectively tracks multiple targets within a video. This streamlined and efficient process is capable of handling multi-object tracking issues within videos, and can operate effectively even when targets undergo occlusions or rapid movements.

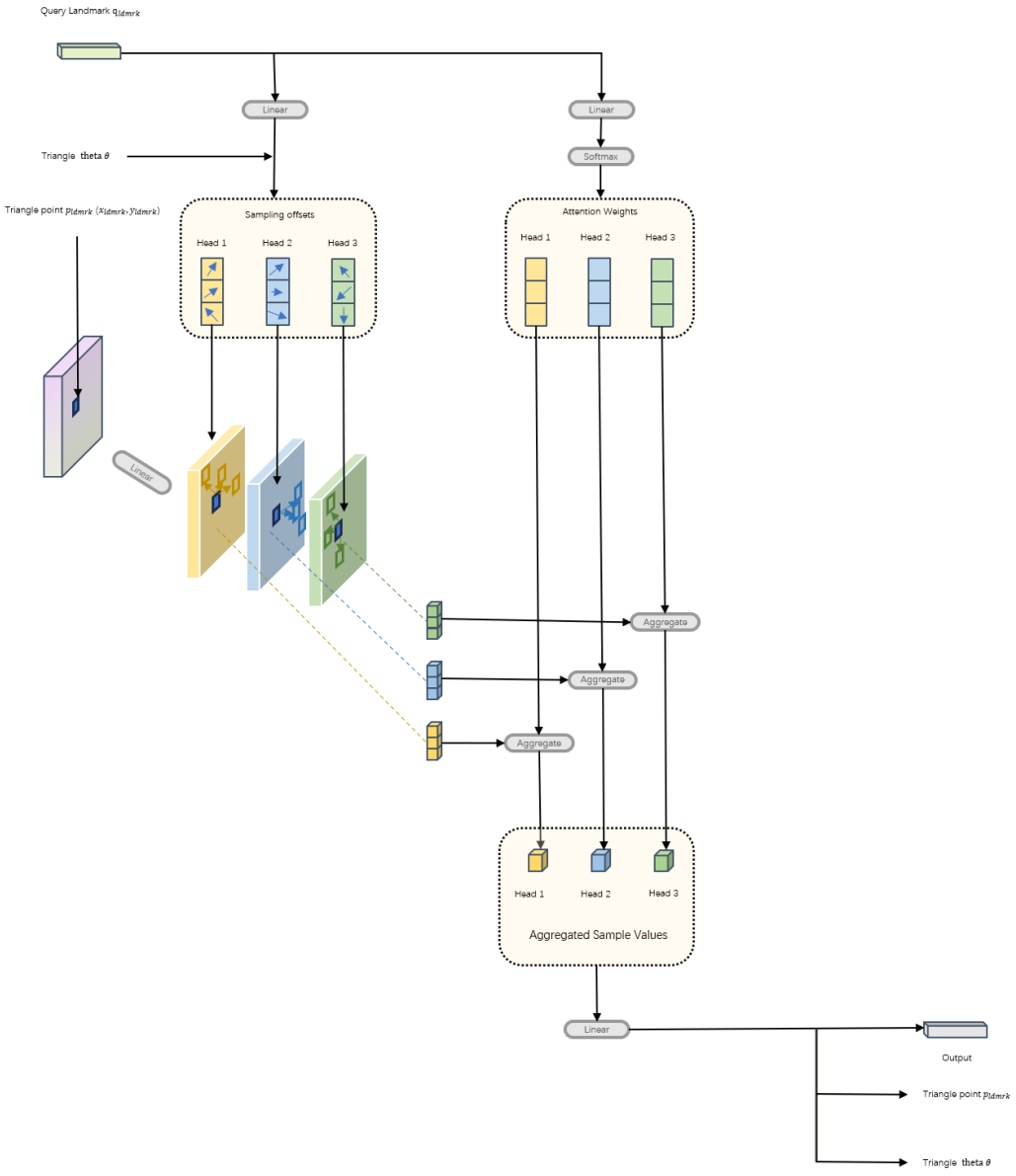

Figure 1. Illustration of the triangular attention mechanism.

The StrongSORT multi-object tracking algorithm is an innovative approach that enhances the traditional SORT (Simple Online and Realtime Tracking) methodology. SORT is known for its simplicity and effectiveness in real-time tracking based on detection results from object detection algorithms; however, the initial version did not account for the appearance features of targets, leading to tracking failures or ID switches, particularly for occluded targets. DeepSORT, an improved iteration of SORT, mitigates these issues by incorporating appearance features and a cascading matching mechanism, resulting in significantly reduced ID switches and enhanced tracking performance.

StrongSORT further elevates the performance of DeepSORT through the integration of advanced detectors, optimized feature extraction techniques, refined data association strategies, and incorporation of the AFLink and GSI algorithms. These advancements notably bolster the accuracy and robustness of DeepSORT in handling multi-object tracking tasks, providing more reliable and continuous tracking even in complex scenarios such as occlusion and swift movements.

Kalman filtering is widely applied in dynamic systems. It utilizes the system's linear state equations to observe system inputs and outputs, thereby achieving optimal state estimation. In practice, inaccuracies in the observed data due to interference make this estimation process also a filtering one.

To make a prediction, the state equation of the Kalman filter is first constructed based on the motion model acquired by the detection algorithm. Let the state of the target at time $k-1$ be $\overrightarrow{X_{k-1}} = \begin{bmatrix} p_{k-1} \\ v_{k-1} \end{bmatrix}$, where p denotes position and v denotes velocity. Within the neighborhood of time $k-1$, the target can be approximated as moving at a constant speed; hence, the state information at time k can be found by

$$\overline{X_k} = \begin{bmatrix} p_k \\ v_k \end{bmatrix}, \quad \text{where} \quad p_k = p_{k-1} + (t_k - t_{k-1})v_{k-1}, \quad v_k = v_{k-1}. \tag{10}$$

In addition, the possible acceleration a_k during the target's motion should also be considered. Let Δt represent the difference between times k and $k-1$; then, the state at time k can be expressed as

$$\begin{cases} p_k = p_{k-1} + v_{k-1}\Delta t + \frac{1}{2}a_k\Delta t^2 \\ v_k = v_{k-1} + a_k\Delta t. \end{cases} \tag{11}$$

Rewriting the above in matrix multiplication form, we have

$$\hat{X}_k = \begin{bmatrix} p_k \\ v_k \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_{k-1} \\ v_{k-1} \end{bmatrix} + \begin{bmatrix} \frac{\Delta t^2}{2} \\ \Delta t \end{bmatrix} a_k. \tag{12}$$

The state transition equation for the Kalman filter during the prediction process can be written as

$$\hat{X}_{k|k-1} = F_k \hat{X}_{k-1} + B_k \vec{u}_k, \tag{13}$$

where

$$F_k = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \quad B_k = \begin{bmatrix} \frac{\Delta t^2}{2} \\ \Delta t \end{bmatrix}. \tag{14}$$

Here, F_k is known as the state transition matrix, which is used to transfer the estimated value from the previous moment to the new predicted position, B_k is known as the control matrix, and $\vec{u}_k$ represents the acceleration a_k. In actual situations, the motion of the target is affected by various factors; $\vec{u}_k$ represents the control input vector, and the product $B_k\vec{u}_k$

represents known external influencing factors, while $\hat{X}_{k|k-1}$ represents the current state estimated using the state estimate from the previous moment, which is an estimated value rather than the true value.

From the analysis above, it is known that the estimates obtained at each moment during prediction have a certain degree of uncertainty. Therefore, it is necessary to represent the uncertainty of the estimated state and the correlation between state quantities. The covariance matrix is used to represent the correlation between the two, with the following calculation formula:

$$P_{k|k-1} = F_k P_{k-1} F_k^T + Q_k. \tag{15}$$

The mathematical model is generally described by the covariance matrix of the noise. Considering the prediction uncertainty and the impact of noise comprehensively, the second prediction formula of the Kalman filter can be obtained as follows:

$$P_{k|k-1} = F_k P_{k-1} F_k^T + Q_k. \tag{16}$$

The system observation equation is then obtained through the transformation matrix H:

$$Z_k = Hx_k + v_k \tag{17}$$

where Z_k represents the observable measurement of the system and v_k represents the noise in the observation process.

In actual measurements, the predicted noise and the actual measured noise are assumed to both follow a Gaussian distribution and to be independent of each other; thus, their product will follow a Gaussian distribution. The gain matrix K is calculated as follows:

$$K_k = P_{k|k-1} H_k^T \left(H_k P_{k|k-1} H_k^T + R_k \right)^{-1}. \tag{18}$$

Finally, using the above formula to correct the prediction formula, the optimal estimate of the target can be obtained as

$$\hat{X}_k = \hat{X}_{k|k-1} + K_k \left(Z_k - H_k \hat{X}_{k|k-1} \right), \tag{19}$$

$$P_k = P_{k|k-1} - K_k H_k P_{k|k-1}. \tag{20}$$

3.3.2. Data Association Design

Data association refers to the process of matching targets detected in each frame with those detected in the previous and even earlier frames while ensuring that the associated targets are the same vehicle and assigning the same ID. Matching is primarily based on the similarity between targets, including motion similarity and representation feature similarity, which are measured by different cost matrices. To ensure that the inter-frame association is as complete as possible, two rounds of matching are conducted; the first consists of cascaded matching of representation and motion, while the second is based solely on the IoU metric.

(a) Cascaded Matching

The representation features are extracted from the detected 2D bounding boxes of the targets. The cost matrix is calculated based on the representation features. The representation matching involves two steps: computing the cosine distance cost matrix, and computing the motion feature similarity. The cosine similarity represents the cosine of the

angle between two feature vectors, while the cosine distance is the unit value minus the cosine similarity:

$$D_{\cos}(i,j) = \min_{k} \left(1 - \frac{f_{t,i} \cdot e_{t-1,j}^{(k)}}{\|f_{t,i}\| \|e_{t-1,j}^{(k)}\|} \right) \quad (k \in R_{t-1,j}) \tag{21}$$

where $e_{t-1,j}^{(k)}$ represents the k-th representation feature of the j-th tracked target at time $t-1$ and $R_{t-1,j}$ is the set of representation features retained for the last 40 frames according to the sequence.

The motion feature similarity measures the spatial proximity between the detected position in the current frame and the predicted position from the previous frame, assuming that the target follows a hypothetical motion model. The Mahalanobis distance is used for this purpose, as it corrects for scale inconsistencies across different dimensions and is invariant to the units of measurement:

$$D_{\text{mv}}(i,j) = (d_i - p_j)^T \Sigma_j^{-1} (d_i - p_j) \tag{22}$$

where d_i represents the detected position of the i-th bounding box, p_j represents the predicted position of the j-th tracked target, and Σ_j is the covariance matrix from the Kalman filter.

The motion feature similarity can be thresholded using the cosine distance cost matrix to prevent targets from matching with vehicles that are too far away in space. The threshold is set to the 95th percentile of the χ^2 distribution with four degrees of freedom, $D_{\text{threshold}} = 9.4877$:

$$D_{\cos} = D_{\cos}[D_{\text{mv}} < D_{\text{threshold}}]. \tag{23}$$

By combining the motion similarity, which provides information about the target's short-term predicted position, with the appearance similarity, which provides information about the target's long-term motion features, we obtain the final cascaded matching similarity matrix through linear weighting:

$$D = \lambda D_{\cos} + (1-\lambda) D_{\text{mv}} \tag{24}$$

where λ is the weight assigned to appearance features and motion features, set as 0.98 in this paper.

(b) Triangular IoU Matching

IoU matching is a method employed within the SORT (Simple Online and Realtime Tracking) algorithm to assess the similarity between two bounding boxes. This approach takes into account not only the overlapping area between the bounding boxes but also their shape and size.

In this work, we deviate from traditional SORT's utilization of GIoU and instead adopt the triangular IoU, a novel evaluation metric proposed in this paper. The triangular IoU provides a more precise description of the target's shape and position, enabling more accurate localization of ship coordinates. This enhancement leads to improved accuracy in target matching, allowing for the more effective association between targets detected in the current frame and those tracked in the previous frame. This improvement is particularly beneficial for SORT in maintaining accurate tracking in complex scenarios, such as when targets are occluded, moving rapidly, or undergoing shape changes.

The use of triangular IoU in SORT offers several advantages; notably, it enhances the algorithm's robustness in handling various challenging conditions that traditional IoU

metrics might fail to address adequately. By incorporating the geometric properties unique to triangular shapes, the triangular IoU better captures the nuances of targets such as ship wakes, leading to more reliable tracking results.

(c) Hungarian Algorithm

The Hungarian algorithm is a combinatorial optimization technique utilized to identify an optimal matching within a bipartite graph, aiming to maximize or minimize the sum of the weights of all matched edges. In the context of the SORT (Simple Online and Realtime Tracking) algorithm, this method addresses the challenge of how to optimally assign the best matches between a set of targets detected in the current frame and those tracked in the previous frame by minimizing the total distance of matches. By minimizing the cost of assignments, the Hungarian algorithm effectively enhances the stability and precision of tracking, reducing the loss or errors in tracking that can result from incorrect matches.

In summary, the Hungarian algorithm offers SORT a highly efficient and reliable approach to managing the problem of target assignment, which is especially crucial in scenarios involving multi-object tracking. This algorithm is pivotal in maintaining the continuity and accuracy of target tracking.

3.4. Overall Framework of the TriangularSORT Algorithm

The TriangularSORT algorithm addresses several distinct situations when matching detection boxes with tracks, which dictate how the algorithm handles the relationship between detected targets and existing tracks. As shown in Figure 2, the following are detailed explanations of these scenarios, applicable to detection-based multi-object tracking methods:

1. Matched Tracks: In this scenario, the detection boxes obtained from the detection algorithm successfully match the predicted boxes in a determined state by the Kalman filter. This implies that the algorithm can continuously track the target and the detection box is associated with an existing track. This is the most common situation in the tracking process, indicating that the tracking system can effectively maintain tracking of the target.
2. Handling of Unmatched Tracks and Detections: For tracks that are initialized and in an unconfirmed state (Unconfirmed Tracks), tracks that fail to match successfully through cascaded matching (Unmatched Tracks), and unmatched detection boxes (Unmatched Detections), the TriangularSORT algorithm attempts to match using an appearance model such as deep learning-based feature extraction. If a match is successful, then the track is updated using the Kalman filter. If the matches fail consecutively, these detection boxes may be initialized as new tracks or deleted according to certain strategies to avoid redundant or erroneous tracking.
3. Mismatched Tracks: If a track fails to match through IoU and is in a determined state, but has fewer mismatches than a set threshold, then the algorithm continues to predict the track and attempts to match it again in the next frame. If the number of mismatches exceeds the threshold, the track may be deleted to prevent tracking loss or erroneous tracks. This mechanism helps the algorithm to maintain effective tracking of other targets even when a target is occluded or temporarily leaves the field of view.

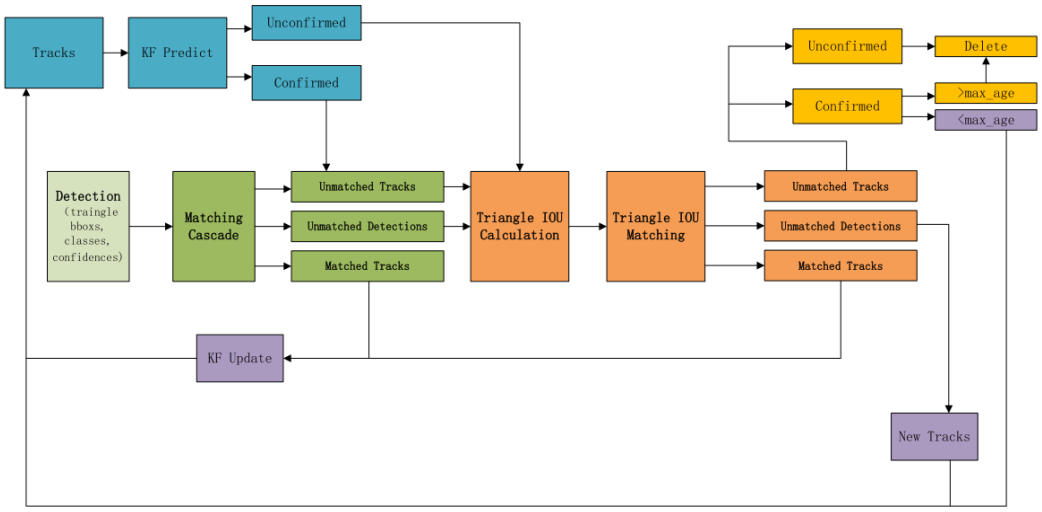

Figure 2. Framework of the multi-object tracking system.

These scenarios cover various situations that the TriangularSORT algorithm may encounter when matching detection boxes with tracks, ensuring the robustness and accuracy of the algorithm. By dynamically handling matched, unmatched, and mismatched situations, TriangularSORT can effectively track multiple targets in a video even in cases of occlusion or rapid movement between targets.

The overall process of the TriangularSORT algorithm is as follows:

- Step 1: Initialization and Prediction: In the initial phase of the TriangularSORT algorithm, object detection algorithms are first used to process the first frame of the video stream to detect targets in the scene and create corresponding tracks. Then, a Kalman filter is initialized for each detected target to predict the target's motion status. In subsequent frames, the Kalman filter predicts the position of the target in the current frame based on the target's historical motion information, a step that does not rely on the detection results of the current frame.
- Step 2: Object Detection and Matching: In each frame, the TriangularSORT algorithm uses object detection algorithms to detect targets, resulting in a series of detection boxes. Then, appearance features are extracted from these detection boxes using a feature embedding model. Next, these detection results are matched with the predicted positions by the Kalman filter. The matching process typically involves computing matching costs such as the Mahalanobis distance, cosine similarity, etc., to determine the best track–detection match pair.
- Step 3: Track Update and Interpolation Processing: For successfully matched tracks, the TriangularSORT algorithm uses detection box information to update the Kalman filter and track status to reflect the target's new position and possible new velocity or direction. For missing detection cases, i.e., targets not detected in the current frame, the algorithm uses the Gaussian-Smoothed Interpolation (GSI) algorithm for track interpolation to fill gaps in the track, thus maintaining continuous tracking of the target.
- Step 4: Global Association and Optimization: For tracks and detection boxes that fail to match successfully, the TriangularSORT algorithm uses the AFLink algorithm for global association to solve the problem of missing associations. The AFLink algorithm uses spatiotemporal information to predict whether two track segments belong to the same target, thereby improving the accuracy of global association. In addition, optimization techniques such as the Hungarian algorithm are used to find the optimal track–detection match pair, ensuring the optimization of tracking results.

- Step 5: Result Output and Iteration: After completing all matching and updating steps, the TriangularSORT algorithm outputs the tracking results of the current frame, including the track and identity information of each target. Then, the algorithm iteratively processes the next frame, repeating the above steps until the end of the video stream. This iterative processing ensures continuous tracking of all targets in the video even in cases of occlusion or rapid movement between targets.

4. Experiments

This section provides a detailed introduction to the experimental environment based on DETR for triangular IoU, detection results, ablation studies, and the tracking results of TriangularSORT on real-world datasets.

4.1. DETR for Triangular IoU

4.1.1. Data Source

The input to the algorithm is a video sequence taken 10 m above the sea surface with a wind speed of 5 m/s and a ship speed of 20 m/s. It is assumed that the size of the sea surface in the original data is 2500 m × 2500 m, and the diffusion range of the ship's wake is set to 1000 m × 1000 m. Other parameters are as shown in Tables 1 and 2.

Table 1. Ship modeling parameters.

Parameter Category	Parameter Value
Ship Speed	5 m/s
Ship Length	100 m
Ship Width	15 m
Draft	7.5 m
Motion Type	Uniform Linear

Table 2. Sea surface modeling parameters.

Parameter Category	Parameter Value
Wave Model	Elfouhaily Spectrum
Wind Speed	10 m/s
Wave Age	0.84
FFT Points	1024
Sea Surface Size	2500 m × 2500 m
Resolution	1 m
Frame Number	1000
Time Interval	0.1 s

Our detection experiments for ship wakes utilized the SWIM dataset, which served as the benchmark in this paper. The training images were maintained at their original dimensions. To ensure reliable performance metrics and equitable comparisons, in the absence of prior research on wake datasets, we utilized all 11,600 positive images in the dataset for training and evaluation purposes, while negative images and 365 instances that were tagged as "difficult" were set aside. Data augmentation techniques encompassed random color distortion, random horizontal and vertical flipping, and random affine transformations, each applied with a probability of 0.5.

In addition, our experiments on ship wake tracking also employed real-world ship wake video sequences from the Macau Science Satellite website, from which a single frame of the video was extracted, as shown in Figure 3.



Figure 3. Selected frame from the video sequence used for tracking performance evaluation.

4.1.2. Evaluation Metric

The standard average precision (AP) as defined in PASCAL-VOC was the primary metric used to assess the performance of WakeDETR and the other CNN-based methodologies. The AP represents the mean precision at various recall levels. The definitions for the precision and recall metrics are as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{25}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{26}$$

where true positive (TP) signifies the number of correctly identified objects, false positive (FP), also known as false alarms, represents the number of incorrectly identified objects, and false negative (FN) denotes the number of missed objects. It is important to note that because other CNN-based methods do not predict the wake tip location and the Kelvin arm direction, the aforementioned metrics are still based on whether the IoU of the predicted bounding box and the ground-truth box exceeds 0.5. The detection outcome of WakeNet on the wake tip and Kelvin arms is currently only demonstrated in its output visualizations.

4.1.3. Training Settings

All models were developed within the PyTorch framework and trained using a single NVIDIA RTX 3090 (24 GB) GPU. The software version number for NVIDIA RTX 3090 (24 GB) GPU is 566.14, manufactured by NVIDIA in Santa Clara, CA, USA. The batch size was set to 4.

4.1.4. Detection Results

The standard average precision (AP) as defined in PASCAL-VOC was primarily utilized to assess the performance of WakeDETR and the other CNN-based methodologies. The AP represents the mean precision at various recall levels. The definitions for the precision and recall metrics are as follows:

$$\text{Precision} = \frac{TP}{TP + FP} \tag{27}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{28}$$

where true positive (TP) signifies the count of correctly identified objects, false positive (FP) represents the count of incorrectly identified objects, and false negative (FN) denotes the count of missed objects. It is important to note that as other CNN-based methods do not predict the wake tip location and the Kelvin arm direction, the aforementioned metrics remain based on whether the Intersection over Union (IoU) of the predicted bounding box and ground-truth box exceeds 0.5.

In analyzing the experimental results listed in Table 3, it can be observed that the detection performance significantly improves with the increase in model complexity, particularly after introducing the "Triangular IOU" and "Triangular Attention" mechanisms. These results suggest that both mechanisms effectively enhance the model's ability to capture key features, increasing its detection accuracy.

Specifically, the baseline model achieves an AP of 62.40%, which can be considered the performance benchmark without additional mechanisms. Upon incorporating "Triangular IOU", the AP increases to 66.26%, indicating that this mechanism improved the model's capability to locate targets to some extent. Further enhancement with the "Triangular Attention" mechanism boosts the AP to 71.50%, demonstrating that this mechanism plays an active role in feature processing, possibly by more accurately simulating the process of visual attention focus. These results show that both mechanisms lead to enhanced detection performance.

When "Triangular Attention" and "Triangular IOU" are both introduced, the AP reaches 73.53%, which is the highest performance among all configurations. This indicates that integration of these two mechanisms comprehensively elevates the model's recognition and localization abilities for targets, possibly through more intricate feature interactions and refined attention allocation that make for more precise target detection.

Table 3. Comparison of detection performance based on backbone depth and configuration.

Backbone Depth	Configuration	AP (%)
50-layer	Baseline	62.40
50-layer	Baseline+Triangular IoU	66.26
50-layer	Baseline+Triangular attention	71.50
50-layer	Baseline+Triangular attention+Triangular IoU	73.53

In Table 4, our proposed TriDETR detection method utilizing the ResNet-50 backbone network attains an AP of 73.53%, further validating the effectiveness of our approach. Compared to other methods, TriDETR exhibits a notable advantage in detection performance, likely due to its unique strengths in feature extraction and target recognition.

In summary, the experimental results demonstrate that the the TriDETR method introducing the triangular attention and triangular IoU mechanisms can significantly enhance the performance of detection models, providing a better foundation for subsequent processing tasks.

Table 4. Comparison of different detection methods.

Method	Baseline	Backbone	AP (%)
R2CNN	Faster R-CNN	ResNet-50	63.24
R-YOLO	YOLO	DarkNet-53	54.87
BBAVectors	CenterNet	ResNet-50	65.19
Our method	DETR	ResNet-50	73.53

4.2. Tracking Results Based on TriangularSORT

TriangularSORT is adaptable to both SAR and optical imagery. For convenience, our subsequent validations were conducted using optical imagery. An example frame from the optical video sequence with a moving ship is shown in Figure 4.

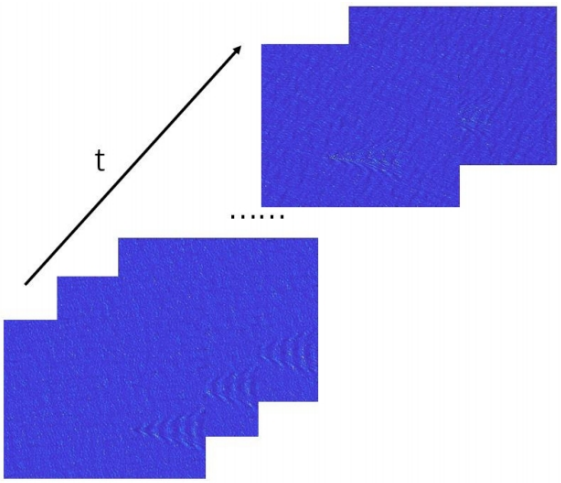

Figure 4. Video sequence of the sea surface with wind speed at 5 m/s and ship speed at 20 m/s.

Applying TrianglarDETR to the sequence of ship wake videos yielded the detection results shown in Figure 5.

If the center of the ship's wake is considered as the actual position of the target, the tracking error of the ship's target is defined as the absolute value of the distance from the center of the ellipse at that moment to the actual wake of the ship. If the center coordinates of the ellipse obtained at moment k are $(m_k, n_k), k = 1, 2, \ldots, 8$, then the actual position of the wake center is (x_k, y_k); then the tracking error is defined as

$$e = \sqrt{(m_k - x_k)^2 + (n_k - y_k)^2}. \tag{29}$$

The minimum distance resolution of the error is related to the block size, and the tracking is generally performed according to the detection results. The higher the detection accuracy, the smaller the tracking error. The error with a wind speed of 5 m/s at an altitude of 10 m above the sea surface and a ship speed of 20 m/s is shown in Table 5.

Table 5 illustrates the tracking error, measured in meters (m), at different observation times; the tracking error refers to the distance between the center of the ship wake and its actual position at a specific time point. At the beginning of the observation (0 s), the tracking error is 6.10 m, which may be due to inaccurate initial position estimation or calibration errors at system startup. From 0 to 20 s, the tracking error decreases from 6.10 m to 4.47 m, indicating that the tracking system quickly adjusts in the initial stage to reduce the error. The tracking error continues to decrease between 20 and 60 s, dropping from 4.47 m to 3.33 m; this reduction in error over time suggests that the tracking system is gradually stabilizing and improving its accuracy. Between 60 and 100 s, the tracking error further decreases, reaching 1.90 m at 100 s, indicating that the tracking system has achieves higher precision and minimizes error during this period. Overall, the tracking error gradually decreases over time, showing that the performance of the tracking system is continuously optimized. This trend is ideal for a ship wake tracking system, as it indicates that the system can track the target more accurately over time.

To assess the tracking performance, we implemented TriangularSORT on a video sequence depicting a vessel navigating along a coastline. The video encompassed a total of 260 frames. One of the extracted frames is shown in Figure 3.

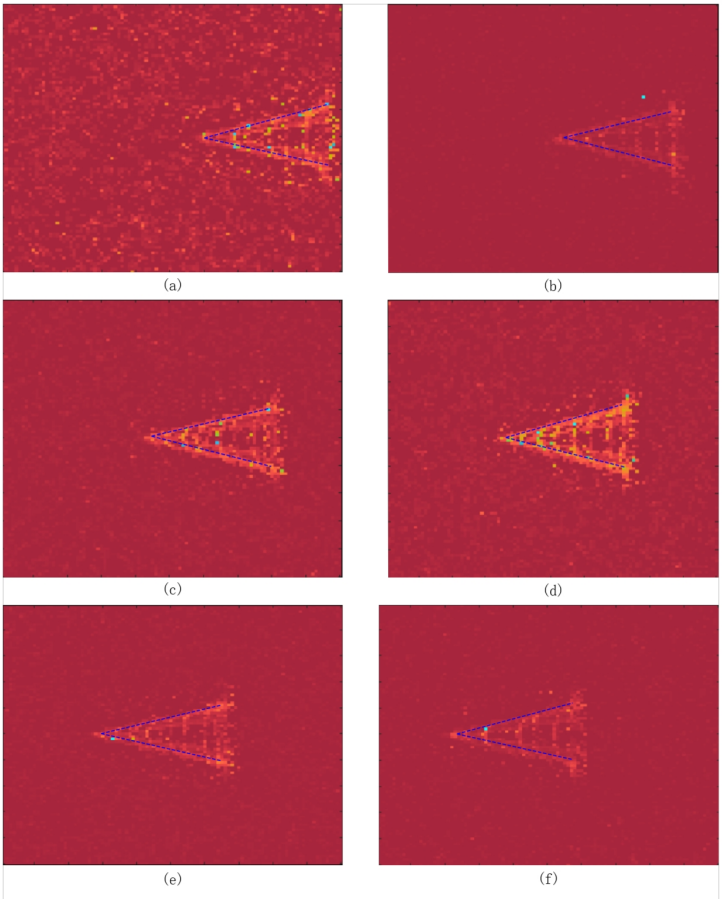

Figure 5. Tracking results using TrianglarDETR on ship wake video sequences; (**a**–**f**) display the results of applying the TriangularDETR detection algorithm to images captured from the ship's wake video sequence at intervals of 16 frames. The two blue lines represent the two Kelvin wakes of the ship and their intersection point indicates the position of the target ship.

Table 5. Tracking error for a Wind Speed of 5 m/s at 10 m above sea level and a ship speed of 20 m/s.

Observation Time/s	Tracking Error/m
0	6.10
20	4.47
40	4.42
60	3.33
80	2.26
100	1.90

The outcomes of applying TriangularSORT to this video segment are depicted in Figure 6, with frames 110, 120, 130, and 140 from the video selected for presentation.

The second set of images illustrates the output of the TriangularSORT algorithm applied to the same video frames. The red arrows indicate the tracked wake patterns, showcasing the algorithm's efficacy in detecting and following the moving wake trails left by the vessel. This visual representation offers a clear demonstration of the algorithm's ability to identify and sustain tracking of dynamic patterns across consecutive frames.

The juxtaposition of the two sets of images reveals the enhancement afforded by the TriangularSORT method. While the original frames capture the wake patterns, the processed images with red lines provide a more accurate and consistent tracking of the wake trails. This enhancement is vital for applications that require precise tracking over time, as it facilitates superior analysis and comprehension of the vessel's movement and the resulting wake patterns.

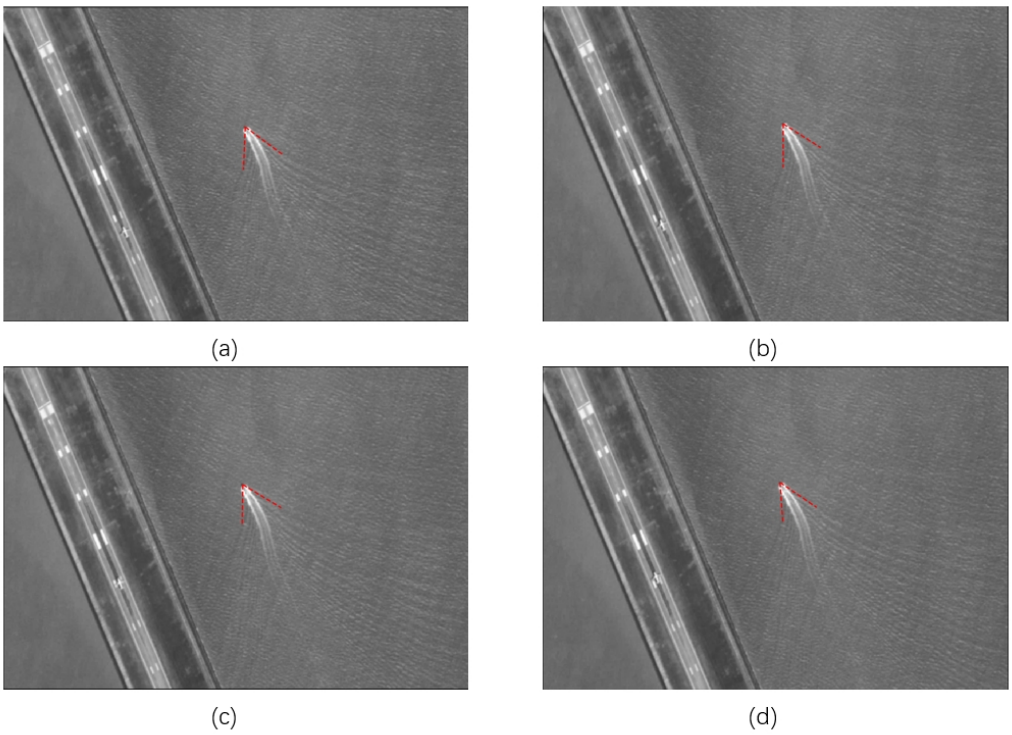

Figure 6. Results from TriangularSORT processing for selected video frames. (**a**) The 30th frame image in the video (**b**) the 50th frame (**c**) the 70th frame (**d**) the 90th frame.

This experiment substantiates the utility of the TriangularSORT method in maritime surveillance or environmental monitoring, where tracking moving objects and their interactions with the surrounding environment is imperative. The findings advocate for the algorithm's potential for integration into extensive monitoring systems and contributing to augmented observational capabilities and data precision.

5. Conclusions

In conclusion, ship wake tracking is a vital area of research with significant implications for maritime safety, environmental monitoring, and security. The integration of advanced machine learning techniques, particularly deep learning, has revolutionized the field, providing more accurate and efficient methods for tracking and analyzing ship wakes. This paper introduces a novel approach to ship tracking that leverages the TriangularSORT algorithm, employing triangular wake tracks to monitor vessels. To substantiate our conclusions, we present a comparison of detection performance based on backbone depth and configuration. The baseline model, with a 50-layer backbone, achieved an average precision (AP) of 62.40%. Our introduction of the triangular IoU mechanism alone improved the AP to 66.26%. Further enhancement was observed with the incorporation of the proposed triangular attention mechanism, which increased the AP to 71.50%. The most significant improvement was achieved when combining the triangular attention and triangular IoU, resulting in an AP of 73.53%. These quantitative results underscore the effectiveness of

our proposed method. The data clearly demonstrate the advantages of our approach over traditional bounding box methods, particularly in scenarios where the dynamics of the ship's wake are critical to understanding its movement. As evidenced by the performance metrics, integration of the proposed triangular attention mechanism guides the model's focus to key areas of the image, significantly improving local feature recognition and analysis in visual tasks. This mechanism is particularly suitable for tasks that require precise localization, further validating the achievements of our research.

Looking ahead, we plan to further refine the TriangularSORT algorithm to accommodate a broader spectrum of marine environments and larger variety of vessel types in order to provide a more rigorous validation of its effectiveness. Additionally, the development of a real-time ship wake monitoring system that can promptly respond to maritime emergencies is a focal point of our future research endeavors.

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

Funding: This work received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used and analysed during the current study are available from the corresponding author upon reasonable request.

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

References

1. Wang, Y.; Wang, C.; Zhang, H.; Dong, Y.; Wei, S. A SAR Dataset of Ship Detection for Deep Learning under Complex Backgrounds. *Remote. Sens.* **2019**, *11*, 765. [CrossRef]
2. Wei, S.; Zeng, X.; Qu, Q.; Wang, M.; Su, H.; Shi, J. HRSID: A High-Resolution SAR Images Dataset for Ship Detection and Instance Segmentation. *IEEE Access* **2020**, *8*, 120234–120254. [CrossRef]
3. Zhang, T.; Zhang, X.; Ke, X.; Zhan, X.; Shi, J.; Wei, S.; Pan, D.; Li, J.; Su, H.; Zhou, Y.; et al. LS-SSDD-v1.0: A Deep Learning Dataset Dedicated to Small Ship Detection from Large-Scale Sentinel-1 SAR Images. *Remote. Sens.* **2020**, *12*, 2997. [CrossRef]
4. Zhang, T.; Zhang, X.; Li, J.; Xu, X.; Wang, B.; Zhan, X.; Xu, Y.; Ke, X.; Zeng, T.; Su, H.; et al. SAR Ship Detection Dataset (SSDD): Official Release and Comprehensive Data Analysis. *Remote. Sens.* **2021**, *13*, 3690. [CrossRef]
5. Lei, S.; Lu, D.; Qiu, X.; Ding, C. SRSDD-v1.0: A High-Resolution SAR Rotation Ship Detection Dataset. *Remote. Sens.* **2021**, *13*, 5104. [CrossRef]
6. Huo, Y.; Cheng, X.; Lin, S.; Zhang, M.; Wang, H. Memory-Augmented Autoencoder With Adaptive Reconstruction and Sample Attribution Mining for Hyperspectral Anomaly Detection. *IEEE Trans. Geosci. Remote. Sens.* **2024**, *62*, 5518118. [CrossRef]
7. Zhang, T.; Zhang, X.; Shi, J.; Wei, S. HyperLi-Net: A hyper-light deep learning network for high-accurate and high-speed ship detection from synthetic aperture radar imagery. *ISPRS J. Photogramm. Remote. Sens.* **2020**, *167*, 123–153. [CrossRef]
8. Shao, Z.; Zhang, X.; Zhang, T.; Xu, X.; Zeng, T. RBFA-Net: A Rotated Balanced Feature-Aligned Network for Rotated SAR Ship Detection and Classification. *Remote. Sens.* **2022**, *14*, 3345. [CrossRef]
9. Li, X.; Li, D.; Liu, H.; Wan, J.; Chen, Z.; Liu, Q. A-BFPN: An Attention-Guided Balanced Feature Pyramid Network for SAR Ship Detection. *Remote. Sens.* **2022**, *14*, 3829. [CrossRef]
10. Kang, K.M.; Kim, D.J. Ship Velocity Estimation From Ship Wakes Detected Using Convolutional Neural Networks. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* **2019**, *12*, 4379–4388. [CrossRef]
11. Del Prete, R.; Graziano, M.D.; Renga, A. First Results on Wake Detection in SAR Images by Deep Learning. *Remote. Sens.* **2021**, *13*, 4573. [CrossRef]
12. Jones, B.; Ahmadibeni, A.; Shirkhodaie, A. Simulated SAR imagery generation of marine vehicles and associated wakes using electromagnetic modeling and simulation techniques. In Proceedings of the Applications of Machine Learning 2021, Pasadena, CA, USA, 13–16 December 2021. [CrossRef]

13. Xue, F.; Jin, W.; Qiu, S.; Yang, J. Rethinking Automatic Ship Wake Detection: State-of-the-Art CNN-based Wake Detection via Optical Images. *IEEE Trans. Geosci. Remote. Sens.* **2022**, *60*, 1–22. [CrossRef]
14. Ding, K.; Yang, J.; Lin, H.; Wang, Z.; Wang, D.; Wang, X.; Ni, K.; Zhou, Q.; Wang, M. Towards real-time detection of ships and wakes with lightweight deep learning model in Gaofen-3 SAR images. *Remote. Sens. Environ.* **2022**, *284*, 113345. [CrossRef]
15. Luofeng, H.; Blanca, P.; Yuanchang, L.; Enrico, A. Machine learning in sustainable ship design and operation: A review. *Ocean. Eng.* **2022**, *266*, 112907. [CrossRef]
16. Panda, J. Machine learning for naval architecture, ocean and marine engineering. *J. Mar. Sci. Technol.* **2023**, *28*, 1–26. [CrossRef]
17. Cai, Z.; Vasconcelos, N. Cascade R-CNN: High Quality Object Detection and Instance Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2021**, *43*, 1483–1498. [CrossRef]
18. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollar, P. Focal Loss for Dense Object Detection. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017. [CrossRef]
19. Zhou, X.; Wang, D.; Krähenbühl, P. Objects as Points. *arXiv* **2019**, arXiv:1904.07850.
20. Tian, Z.; Shen, C.; Chen, H.; He, T. FCOS: Fully Convolutional One-Stage Object Detection. In Proceedings of the 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Republic of Korea, 27 October–2 November 2019. [CrossRef]
21. Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-End Object Detection with Transformers. In Proceedings of the Computer Vision—ECCV 2020, Glasgow, UK, 23–28 August 2020; pp. 213–229; Lecture Notes in Computer Science. [CrossRef]
22. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; Polosukhin, I. Attention is All you Need. In Proceedings of the Neural Information Processing Systems, Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017.
23. Zhu, X.; Su, W.; Lu, L.; Li, B.; Wang, X.; Dai, J. Deformable DETR: Deformable Transformers for End-to-End Object Detection. *arXiv* **2020**, arXiv:2010.04159.
24. Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; Wei, Y. Deformable Convolutional Networks. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017. [CrossRef]
25. Dai, X.; Chen, Y.; Yang, J.; Zhang, P.; Yuan, L.; Zhang, L. Dynamic DETR: End-to-End Object Detection With Dynamic Attention. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021. [CrossRef]
26. Yao, Z.; Ai, J.; Li, B.; Zhang, C. Efficient DETR: Improving End-to-End Object Detector with Dense Prior. *arXiv* **2021**, arXiv:2104.01318.
27. Zhang, S.; Xinjiang, W.; Wang, J.; Pang, J.; Lyu, C.; Zhang, W.; Luo, P.; Chen, K. Dense Distinct Query for End-to-End Object Detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023.
28. Meng, D.; Chen, X.; Fan, Z.; Zeng, G.; Li, H.; Yuan, Y.; Sun, L.; Wang, J. Conditional DETR for Fast Training Convergence. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021. [CrossRef]
29. Wang, Y.; Zhang, X.; Yang, T.; Sun, J. Anchor DETR: Query Design for Transformer-Based Object Detection. In Proceedings of the AAAI Conference on Artificial Intelligence, AAAI 2022, Virtual, 22 February–1 March 2022.
30. Liu, S.; Li, F.; Zhang, H.; Yang, X.; Qi, X.; Su, H.; Zhu, J.; Zhang, L. DAB-DETR: Dynamic Anchor Boxes are Better Queries for DETR. In Proceedings of the 2022 International Conference on Learning Representations, Virtual, 25–29 April 2022.
31. Sun, Z.; Cao, S.; Yang, Y.; Kitani, K. Rethinking Transformer-based Set Prediction for Object Detection. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), Montreal, BC, Canada, 11–17 October 2021. [CrossRef]
32. Li, F.; Zhang, H.; Liu, S.; Guo, J.; Ni, L.; Zhang, L. DN-DETR: Accelerate DETR Training by Introducing Query DeNoising. In Proceedings of the 2022 Conference on Computer Vision and Pattern Recognition, New Orleans, LO, USA, 19–24 June 2022.
33. Zhang, H.; Li, F.; Liu, S.; Zhang, L.; Su, H.; Zhu, J.; Ni, L.; Shum, H.Y. DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection. In Proceedings of the 2023 International Conference on Learning Representations, Kigali, Rwanda, 1–5 May 2023.
34. Chen, Z. From Traditional Methods to Deep Learning: An Overview of Target Tracking Methods. In *Machine Intelligence*; Springer: Warsaw, Poland, 2020.
35. Cao, D.; Fu, C.; Jin, G. A Survey on Target Tracking Algorithms Based on Machine Learning. *Comput. Sci.* **2016**, 1–7.
36. Gao, J.; Zhang, T.; Xu, C. Graph Convolutional Tracking. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Virtual, 10–17 June 2020.
37. Dong, X.; Shen, J.; Wang, W.; Liu, Y.; Shao, L.; Porikli, F. Hyperparameter Optimization for Tracking with Continuous Deep Q-Learning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Virtual, 10–17 June 2020.
38. Voigtlaender, P.; Luiten, J.; Torr, P.H.; Leibe, B. Siam R-CNN: Visual Tracking by Re-Detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Virtual, 10–17 June 2020.

39. Zhang, Z.; Peng, H.; Fu, J.; Li, B.; Hu, W. Ocean: Object-Aware Anchor-Free Tracking. In Proceedings of the European Conference on Computer Vision, Virtual, 10–17 June 2020.
40. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. *arXiv* **2016**, arXiv:1602.00763.
41. Bertinetto, L.; Valmadre, J.; Leal-Taixé, L.; Frossard, P.; Henriques, J.F. Distractor-aware channel selection for multiple object tracking. *arXiv* **2016**, arXiv:1602.03488.
42. Valmadre, J.; Bertinetto, L.; Henriques, J.F.; Vedaldi, A.; Torr, P.H. End-to-end representation learning for correlation filter based tracking. In Proceedings of the European Conference on Computer Vision, Venice, Italy, 22–29 October 2017.
43. Wang, Q.; Gao, J.; Xing, K.; Zhang, P.; Lu, H. DeepSORT: Deep learning based single object tracking and multi-object tracking. *arXiv* **2019**, arXiv:1909.05339.
44. Held, D.; Thrun, S.; Savarese, S. End-to-end people detection in crowded scenes. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016.
45. Yang, T.; Chan, A. Learning Dynamic Memory Networks for Object Tracking. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
46. Park, E.; Berg, A. Meta-Tracker: Fast and Robust Online Adaptation for Visual Object Trackers. In Proceedings of the European Conference on Computer Vision, Munich, Germany, 8–14 September 2018.
47. Huang, J.; Zhou, W. Re2EMA: Regularized and Reinitialized Exponential Moving Average for Target Model Update in Object Tracking. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019.
48. Cao, X.; Zhang, L.; Ma, J. First Results of Ship Wake Detection by Deep Learning Techniques in Multispectral Spaceborne Images. In Proceedings of the IEEE International Conference on Image Processing, Anchorage, AK, USA, 19–22 September 2021; pp. 1735–1739.

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.