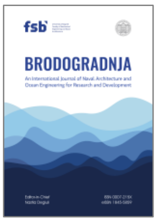
University of Zagreb
Faculty of Mechanical
Engineering and Naval
Architecture

journal homepage: www.brodogradnja.fsb.hr

Brodogradnja

An International Journal of Naval Architecture and
Ocean Engineering for Research and Development



A PointPillars-based 3D point cloud object detector of USVs for small target detection in dynamic aquatic environments

Xue Fan[1], *Shaolong Yang*[1,2,3*], *Xianbo Xiang*[1,2,3], *Shuo Sun*[1], *Shimhanda Daniel Hashali*[1,2,3]

Affiliation

[1] *School of Naval Architecture and Ocean Engineering, Huazhong University of Science and Technology, 430074, Wuhan, China*

[2] *International Science and Technology Cooperation Offshore Center for Ship and Marine Intelligent Equipment and Technology, Hubei, Wuhan 430074, China*

[3] *Wuhan Belt & Road Joint Lab of Ship and Marine Intelligent Equipment and Technology, Wuhan 430074, China*

ARTICLE INFO

Keywords:
Unmanned Surface Vehicles
LiDAR
3D Point Cloud detector
PointPillars

ABSTRACT

LiDAR, a crucial sensor for Unmanned Surface Vehicles (USVs), allows for precise 3D modelling but encounters challenges in real-time target detection due to sparse point clouds. Current 3D point cloud detectors struggle to effectively capture fine-grained details and dynamic water surface features, while high-performance models often rely on custom operators, making deployment more complicated. Additionally, current water surface datasets lack the resolution necessary for small target detection. To tackle these issues, this study enhances the PointPillars model with the Voxel-Guided Label Assignment (VGLA) strategy, improving feature extraction through adaptive label assignment. A high-resolution point cloud dataset focused on small aquatic objects has also been developed based on 128-beam LiDAR. The proposed PointPillars-VGLA achieves 3D AP scores of 89.50%, 83.70%, and 75.20%, as well as BEV AP scores of 95.20%, 91.00%, and 86.70% across three target categories. Ablation experiments confirm the effectiveness of the VGLA module, with accuracy gains of up to 2.27% over CenterPoint. Deployed on the Jetson AGX Orin with TensorRT, the model achieves real-time inference at 30 FPS, enabling efficient detection and tracking in dynamic aquatic environments.

1. Introduction

Unmanned Surface Vehicles (USVs) serve as intelligent and compact platforms that are widely used in various maritime operations, including meteorological monitoring, offshore exploration, and waterborne patrols [1-3]. Those autonomous platforms rely on the coordinated functioning of multiple modules to perform complex tasks [4], with the perception module playing a critical role in ensuring safe and efficient operation [5-7]. By collecting environmental data and executing real-time visual tasks such as target detection and scene segmentation, the perception module [8] enables USVs to interpret their surroundings accurately [9]. Among perception technologies [10], LiDAR stands out for its ability to emit laser pulses and measure the distance,

[*] Corresponding author.

E-mail address: yangsl@hust.edu.cn

https://doi.org/10.21278/brod77202
Received 26 February 2025; Received in revised form 19 July 2025; Accepted 10 August 2025
Available online 14 October 2026
ISSN 0007-215X; eISSN 1845-5859

shape, and position of objects with high precision. The lightweight design, and low power consumption make it particularly suitable for small USVs, where space and energy efficiency are critical [11]. Additionally, LiDAR's adaptability to all-weather conditions further enhances its attractiveness as a perception solution compared to cameras [12].

Despite its advantages, the LiDAR integration in USV applications remains less mature and widespread compared to cameras, primarily due to several challenges. First, LiDAR is substantially more expensive than cameras, with its price increasing exponentially as the number of vertical beams (e.g., 16, 32, or 6) rises. Affordable options, such as 16-beam LiDAR, offer limited resolution, reducing the effectiveness in detecting small targets. Second, there is a notable scarcity of high-performance and easy-deployment 3D point cloud detectors. Mainstream detectors like PointPillars [13] often exhibit suboptimal accuracy in complex multi-target environments. Finally, the field suffers from a lack of high-quality annotated water surface point cloud datasets for object detection. Unlike image datasets, 3D object detection requires annotating seven-degree-of-freedom bounding boxes, making the process labour-intensive and time-consuming. Existing water surface datasets are predominantly based on radar or cameras [14], while LiDAR datasets often rely on 16-beam sensors, which are primarily effective for detecting large vessels. For the small targets, the distribution density of the point cloud is insufficient, making it difficult to clearly describe the detailed features.

Based on the above issues, the main contributions of this paper can be summarized in three aspects:

- **Construct a high-resolution water surface point cloud dataset with 128-beam LiDAR.** The dataset encompasses three categories: 175 USV, cube buoys, and triangular buoys. Additionally, reflective panels are incorporated as a data augmentation strategy to enhance the intensity dimension.
- **Design an easily deployable high-performance 3D object detector: PointPillars-VGLA.** Based on VGLA, PointPillars-VGLA achieves state-of-the-art detection accuracy on the custom dataset, and its effectiveness of VGLA is validated through ablation experiments.
- **Conduct comprehensive field tests for target detection and tracking in dynamic aquatic environments.** The PointPillars-VGLA model, trained on the custom dataset, is deployed on a Jetson AGX Orin-equipped USV, achieving a real-time inference speed up to 30 FPS.

2. Related work

2.1 Water surface datasets

In the field of water surface target detection, multiple datasets have been constructed and widely applied to various scenarios and tasks. For instance, the water surface unmanned vehicle detection dataset [15] comprises 1,550 images, with target categories including ships, floating objects, and obstacles, annotated in Pascal VOC format, making it suitable for unmanned vehicle environmental perception tasks. USVInland [16] is the first inland unmanned vessel dataset under real-world scenarios with multiple sensors and weather conditions, encompassing three tasks: Simultaneous Localization and Mapping, stereo matching, and water's edge segmentation. The FloW dataset [17] is the world's first water surface floating debris detection dataset from the perspective of unmanned vessels, utilizing a combination of image and millimetre-wave radar data to detect floating debris on the water surface, while also supporting small target detection.

These datasets, with their unique characteristics in terms of target types, data volume, and applicable scenarios, provide valuable resources for research on water surface target detection. However, current studies are still mainly focused on scenarios such as USVs' navigation and ocean monitoring, relying heavily on images as the primary sensing modality. In contrast, publicly available datasets for high-precision small target detection on water surfaces using LiDAR remain scarce [18], leaving room for further exploration in this area.

2.2 3D Water surface object detection

Water surface point cloud object detection has become an important research direction in intelligent shipping and uncrewed vessels in recent years. With the widespread adoption of 3D sensors like LiDAR, researchers have proposed various point cloud-based object detection methods, including traditional clustering and segmentation techniques, as well as deep learning models such as PointNet [19] and SECOND [20]. To improve detection accuracy and robustness, techniques such as multi-scale feature fusion, dynamic label assignment, and multi-sensor fusion have also been introduced [21, 22]. Recent studies [23, 24] further focus on the aggregation of local geometric relation features for small object detection. For example, Yang et al. [25] proposed a geometric relation-based feature aggregation method, which significantly improves small object detection performance. Despite these advances, challenges such as data sparsity, noise interference, and the difficulty of detecting small objects remain, and related datasets and evaluation standards need further improvement.

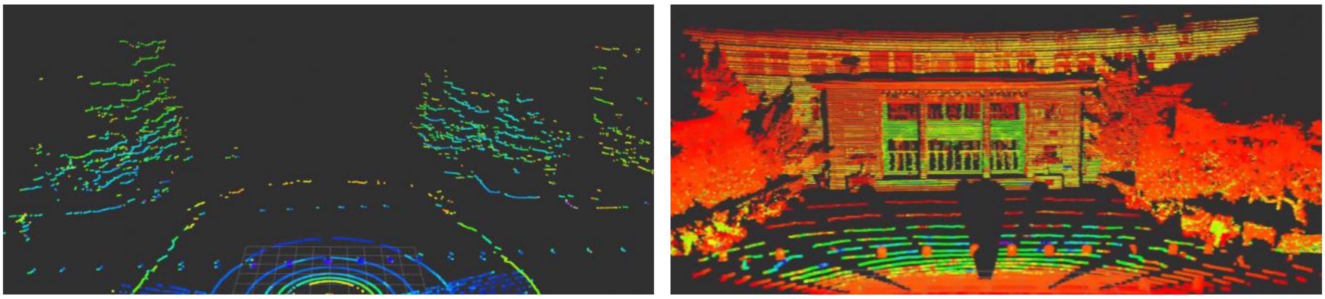
2.3 3D Label assignment strategy

In the domain of 3D object detection using point clouds, the label assignment strategy plays a pivotal role in model training, significantly impacting the selection of positive and negative samples as well as the computation of loss functions. Given the inherent sparsity of point cloud data and the complexity of 3D detection tasks, the development of effective label assignment strategies has garnered considerable attention. Initial approaches relied on anchor-based methods, where positive and negative samples were determined by predefined Intersection over Union (IoU) thresholds between anchor boxes and Ground Truth boxes (GT boxes). For instance, the SECOND [20] employed fixed IoU thresholds, although this method is highly dependent on the design of anchor boxes and involves substantial computational overhead.

In recent advancements, dynamic label assignment strategies have gained traction, offering improved performance through adaptive adjustment of assignment criteria. PV-RCNN [26], for instance, dynamically modifies IoU thresholds based on the feature similarity between candidate boxes and GT boxes. Meanwhile, H3DNet [27] integrates attention mechanisms to utilize local feature semantic information, thereby refining the selection of positive samples and making the assignment process more rational. These innovations underscore the ongoing evolution and sophistication of label assignment strategies in point cloud object detection.

3. High-resolution water surface 3D point cloud object datasets

3.1 Water surface datasets

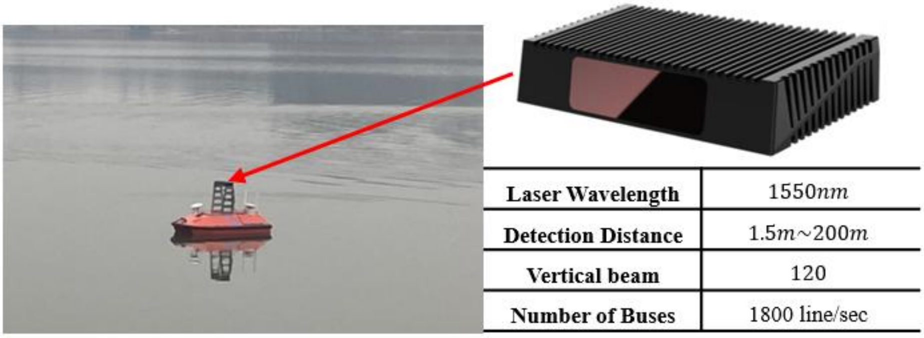


(a) 16-beam mechanical LiDAR (b) 128-beam solid-state LiDAR

Fig. 1 The Comparison of the LiDAR outputs on the same building

For comparative analysis, we utilize both the widely accessible 16-beam mechanical LiDAR and the 128-beam solid-state LiDAR to scan the same building, as depicted in Fig. 1, highlighting the enhanced capabilities of our LiDAR from the same building. As shown in Fig. 1, the scanning result of the 128-line solid-state LiDAR demonstrates significantly higher point cloud density and richer detail compared to the 16-line mechanical LiDAR. Specifically, a 16-line LiDAR typically collects around 320,000 points per second, while a 128-line LiDAR can reach up to 1.3 million points per second, resulting in nearly a fivefold increase in point cloud density. With similar costs, the 128-line LiDAR shows clear performance advantages, making it an ideal choice for scenarios requiring high-precision environmental perception.

The 128-line solid-state LiDAR, detailed in Fig. 2, offers a detection span of 1.5 to 250 meters, with a horizontal coverage of 120°, and a vertical sweep of 25°. It delivers an angular precision of up to 0.13° × 0.1° and can produce 1.3 million points per second at a frame rate of 10 FPS. To guarantee the acquisition of ample point cloud data, this study establishes an effective scanning range: in the *x*-axis, from -30 meters to 30 meters; in the *y*-axis, from 0 meters to 60 meters; and in the *z*-axis, from -3 meters to 1 meter.



Laser Wavelength	1550*nm*
Detection Distance	1.5*m*~200*m*
Vertical beam	120
Number of Buses	1800 line/sec

Fig. 2 A USV equipped with an LS180S2-B LiDAR

This study is carried out in natural aquatic environments, employing a USV integrated with an LS180S2-B LiDAR for data collection. As illustrated in Fig. 3, the study dynamically captures point cloud data for three categories of targets across diverse relative angles, distances, and velocities, ensuring a comprehensive dataset. Specific details regarding the classifications and dimensions of these target types are outlined in Table 1. To date, a total of 3,978 frames have been meticulously annotated to facilitate this research. Real-time data visualization and recording are achieved by integrating LiDAR with the ROS topic interface.

Table 1 The categories and frames of datasets

Category	Dimension(cm)	Frames
Triangular buoy	100 ×100 × 100	1000
Cube buoy	80 × 80 × 80	1000
175 USV	175 × 60 × 80	1978

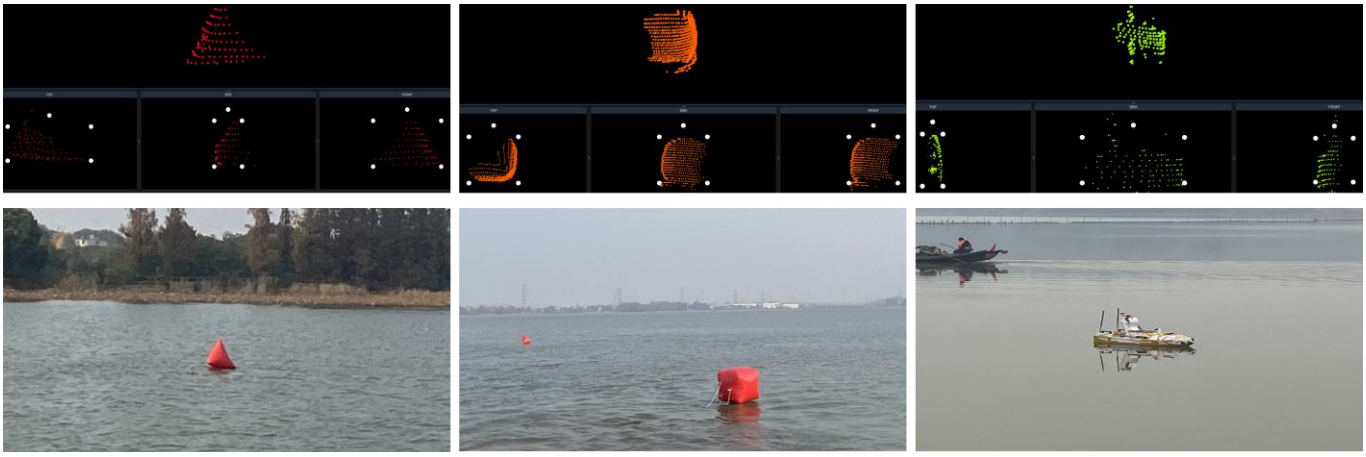


Fig. 3 The point cloud object detection dataset with three types of targets. From left to right: a triangular buoy, a cubic buoy, and a USV with a length of 1.75m

During the data playback stage, the ROS bag technique is utilized to capture data at one frame per second, enabling efficient subsequent target annotation. The annotation format and dataset management approach are designed to align with the KITTI benchmark [28], ensuring smooth compatibility with the majority of 3D point cloud object detectors and training frameworks. However, the abstract nature of point cloud, in contrast to images, makes the annotation process significantly more laborious and introduces several complexities.

A primary challenge lies in the inherent constraints of LiDAR technology, which limit its capacity to capture all target features comprehensively. To mitigate this, we design annotation boxes with sufficient margins to accommodate targets observed from various angles, thereby ensuring the consistency of the 3D bounding boxes. For cubic and triangular buoys, the orientation of the annotation boxes is determined by identifying the surface direction with the highest point cloud density, as detected by the LiDAR, to fulfill the seven-degree-of-freedom annotation requirement. Moreover, since the apparent size of targets in the point cloud remains constant regardless of distance, the annotation boxes for the same category are kept as consistent as possible throughout the annotation process.

3.2 Dataset evaluation metrics

To enhance the dataset's reliability and scientific rigor, it is essential to establish comprehensive, equitable, and practical evaluation criteria for assessing the performance of 3D object detection models in terms of regression, localization, and classification. In object detection datasets, key evaluation metrics such as Average Precision (AP) and Mean Average Precision (mAP) are widely adopted, consistent with practices in image-based detection tasks. In the 3D domain, to accurately quantify detection precision, 3D Intersection over Union (IoU) and Bird's Eye View (BeV) IoU are often combined with AP/mAP. These metrics evaluate positive and negative samples based on intersection-over-union calculations across different spatial dimensions, thus providing a comprehensive assessment of model performance [29]. Additionally, supplementary metrics—including localization error, orientation error, velocity error, and frame rate—are employed to further elucidate the overall effectiveness and robustness of the model.

To ensure the precision of dataset evaluation models and minimize computational complexity during the evaluation process, this paper utilizes evaluation criteria that combine AP with 3D IoU and BeV IoU. A detailed explanation is provided below:

AP is a widely adopted metric for evaluating the performance of single-class detection tasks, defined as the area under the Precision-Recall (PR) curve. The computation process, as illustrated in Fig. 4, involves several steps: First, the model's predicted bounding box information—including position parameters (e.g., center coordinates, width, and height), class labels, and confidence scores—is extracted. All predicted bounding boxes are then sorted in descending order according to their confidence scores, as higher scores indicate a greater likelihood of the prediction aligning with the correct target. Subsequently, a confidence threshold is applied to filter out predictions, retaining only those with confidence scores meeting or exceeding the threshold. The confidence threshold is gradually decreased from 0.9 to 0.1, with intervals of 0.05 used as the set of thresholds.

To match predicted boxes with GT boxes, the 3D/BEV IoU threshold is utilized. IoU quantifies the degree of overlap between the predicted box and the GT box, with its calculation formula as follows:

$$\mathrm{IoU} = \frac{\text{Area of Intersection}}{\text{Area of Union}} \tag{1}$$

3D IoU is calculated within the three-dimensional spatial domain, while BEV IoU is assessed on the bird's-eye view plane, specifically the *X-Y* plane. For predicted bounding boxes that satisfy the confidence threshold, they are paired with GT boxes in a one-to-one correspondence.A successful match is considered a True Positive (TP) if the IoU between a predicted box and a GT box meets or surpasses the predefined threshold. If a GT box remains unmatched to any predicted box, it is labeled as a False Negative (FN). On the other hand, if a predicted box fails to match any GT boxes or if the IoU is below the threshold, it is classified as a False Positive (FP). At each confidence threshold, the metrics Precision and Recall are derived as follows:

$$\mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{2}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{3}$$

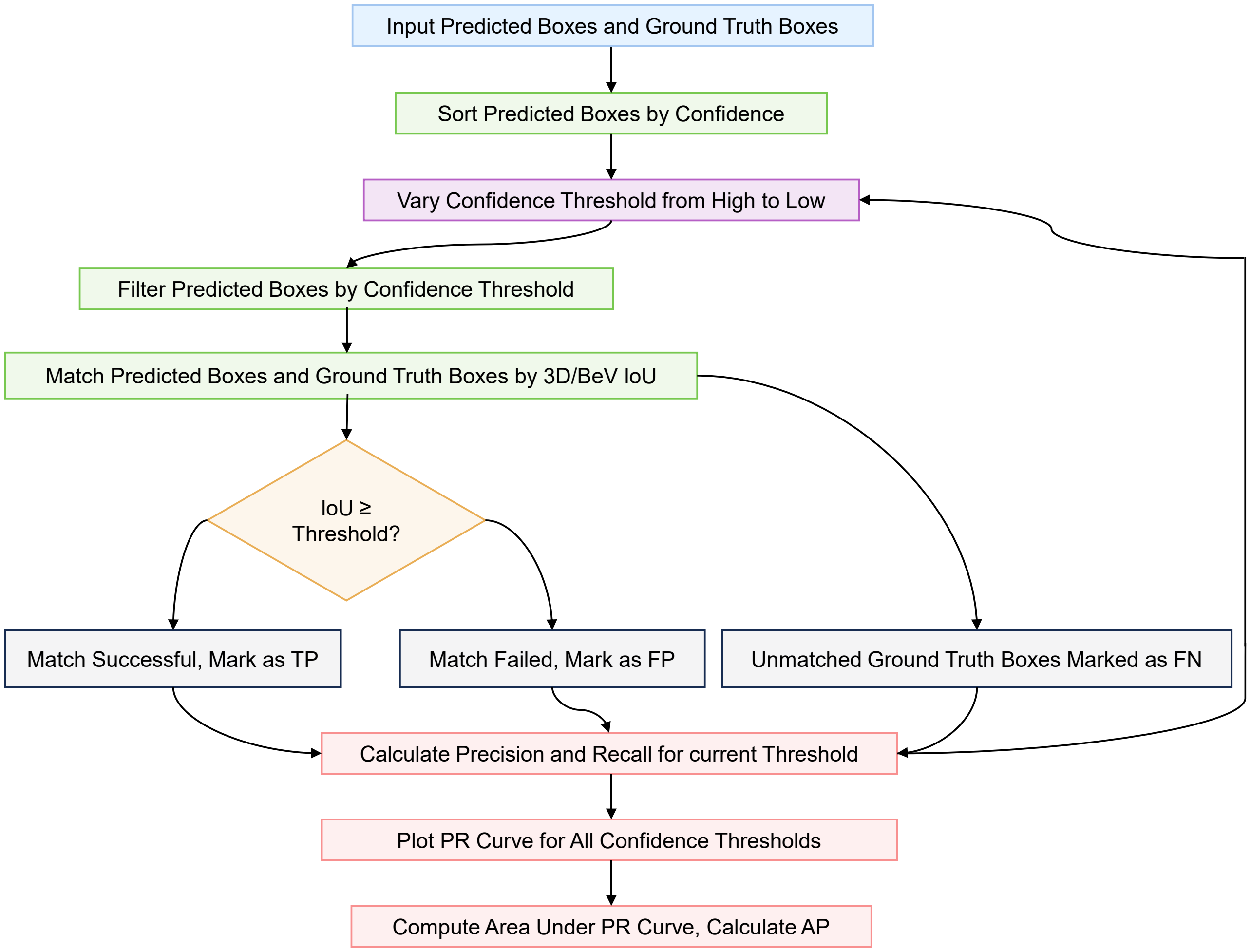
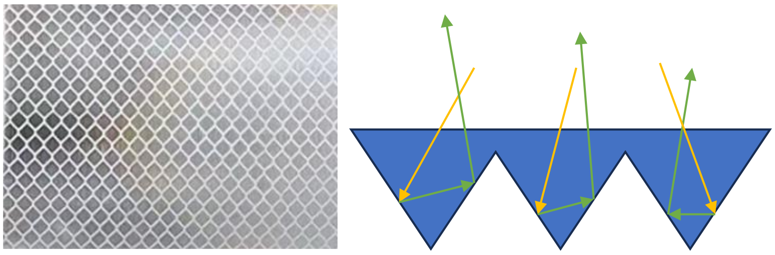

Input Predicted Boxes and Ground Truth Boxes
Sort Predicted Boxes by Confidence
Vary Confidence Threshold from High to Low
Filter Predicted Boxes by Confidence Threshold
Match Predicted Boxes and Ground Truth Boxes by 3D/BeV IoU
IoU ≥ Threshold?
Match Successful, Mark as TP
Match Failed, Mark as FP
Unmatched Ground Truth Boxes Marked as FN
Calculate Precision and Recall for current Threshold
Plot PR Curve for All Confidence Thresholds
Compute Area Under PR Curve, Calculate AP

Fig. 4 The evaluation process of the dataset

Precision measures the ratio of correctly predicted boxes to the total number of predicted boxes, while Recall evaluates the proportion of GT boxes that are successfully identified.

By systematically iterating through all possible confidence thresholds, the corresponding Precision and Recall values are computed, and a Precision-Recall (PR) curve is generated, with Recall plotted on the x-axis and Precision on the y-axis. As the confidence threshold decreases, recall improves due to the detection of a greater number of GT boxes. However, precision declines as the increase in FP leads to a reduction in overall precision. This evaluation framework ensures a comprehensive and comparable assessment of model performance and is widely regarded as the standard methodology in the field of 3D object detection.

3.3 Data augmentation method based on reflective panels



(a) Reflective panels (b) micro-prism structure

Fig. 5 Reflective Panels Characteristics

In addition to employing the time-of-flight measurement technique to capture the positional coordinates of the laser beam, LiDAR also records the energy value of the reflected beam as the intensity value. This intensity value is primarily influenced by four key factors: the reflectivity of the scanned object's surface, the material composition of the object's surface, the angle of incidence and reflection of the laser beam, and the distance between the LiDAR and the object. Generally, higher surface reflectivity, lower material absorption, more perpendicular incidence angles, and shorter distances result in higher intensity values for the returned beam.

The design of the reflective panels [30] is meticulously optimized to fulfill the requirement for high-intensity LiDAR reflection. Firstly, as illustrated in Fig. 5(a), the surface of the reflective panels is typically coated with a fluorescent material, which minimizes light absorption and maximizes the amount of light energy reflected. Additionally, the reflective panels are often constructed from materials with high reflectivity, such as the micro-prism structure depicted in Fig. 5(b). This material enables directional reflection, ensuring that regardless of the angle from which the laser beam is emitted, the majority of the incident light is redirected back to the LiDAR receiver, thereby significantly boosting the return signal.

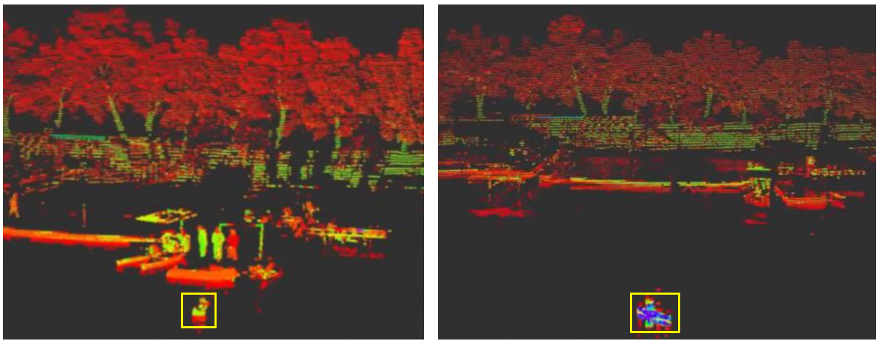

(a) with reflective panels (b) without reflective panels

Fig. 6 Enhance the point cloud intensity values by attaching reflective panels

To further validate the enhancement in intensity value provided by the reflective panels, a comparative experiment is conducted on the same USV, both with and without the reflective panel, under identical environmental conditions. As depicted in the Fig. 6, by replaying the scan data using ROS and RVIZ software, it becomes evident that in the absence of the reflective panels, the point cloud color of the target USV matches that of the shoreline behind it, and the intensity values, as measured by the PCL library, are predominantly below 100 (with a maximum value of 255). However, upon installing the reflective panels, the point clouds color of the target USV distinctly contrasts with that of the shoreline, and the intensity values exceed 200.

4. Improved PointPillars with VGLA

PointPillars is a pillar-based point cloud object detector characterized by its fewer parameters and easy deployment, making it one of the most commonly used models in the industrial field. However, it adopts the same anchor-based label assignment strategy as SECOND, which results in high computational overhead and leaves significant room for improvement. In addition to issues such as heavy reliance on prior information and high computational costs, it also suffers from an imbalance in the number of assigned samples for different objects. Generally, larger objects occupy a greater proportion of the feature map and are more likely to obtain more positive samples, while smaller objects may lack positive samples altogether, leading to significant differences in detection performance for targets of varying sizes. To address the aforementioned issues, the detection head based on PointPillars is modified, and a Voxel-Guided label assignment method is proposed. It should be noted that, compared to the standard PointPillars, only the detection head is modified.

4.1 Voxel-Guided Label Assignment

Since point clouds are typically confined to the object's surface, there is usually no voxel present at the center of the GT box. To enhance the utilization of positive samples, we introduce VGLA, an extension of the CenterPoint [31] framework, as illustrated in Fig. 7. This approach leverages the relative positions between the GT box and all surrounding voxels to improve the effectiveness of positive sample utilization. Notably, this strategy is exclusively applied during the training phase. The predicted values are extracted from the corresponding voxel positions within the candidate regions.

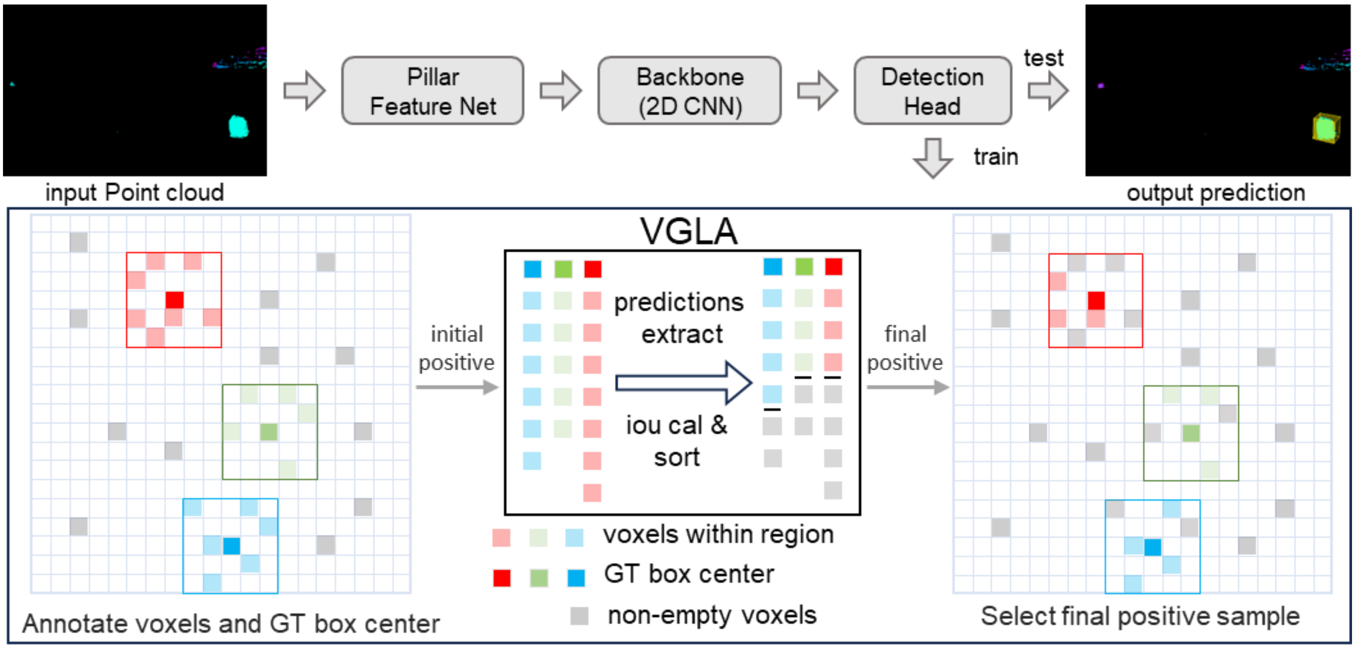


Fig. 7 PointPillars with VGLA

The specific implementation details are outlined in **Algorithm 1**: for all GT boxes and voxels in a certain batch, given a feature map $F \in \mathbb{R}^{C\times H\times W}$, where C represents the number of feature channels, and H and W are the height and width, respectively. Construct a mask $M \in \{0,1\}^{H\times W}$, as shown in the Fig. 7. All voxel positions $V = \{v_i = [x_i, y_i] \in \mathbb{R}^{N\times 2}\}$ and the center coordinates of the GT box $C = (x_t, y_t)$ are identified. Taking the GT box coordinates as the center define a square region P as the interest region. The interest region P, is formally defined as:

$$P = \left\{(x, y) \in \mathbb{R}^2 \middle| |x - x_t \right| \leq \frac{L}{2}, |y - y_t| \leq \frac{L}{2}\right\} \tag{4}$$

L represents the side length of the region. Determine if voxels are within the region of the GT box and update the mask:

$$M(\lfloor x_i \rfloor, \lfloor y_i \rfloor) = \begin{cases} 1 (\lfloor x_i \rfloor, \lfloor y_i \rfloor) \in P \\ \quad 0 \text{ otherwise} \end{cases} \tag{5}$$

Assign a value of 1 to the voxel positions located within the region P, designating them as potential candidate positions for positive samples. On the feature map, retrieve the predicted values at the positions where the mask equals 1 and compute the IoU. The number of positive samples for the corresponding GT box is determined by the total IoU between the predicted values and the target values, with a minimum threshold:

$$k = min\left(\alpha * max\left(\sum_{j \in P} \mathrm{IoU}\left(G^t, P_j^t\right) \right), R \right) \tag{6}$$

where α serves as a hyperparameter, R is the minimum threshold, which is usually set to 1. G^t represents a certain GT Box t, and P_j^t represents the candidate positive samples for it. The top k predicted values, determined by the IoU results, are chosen as the final positive samples. Unlike CenterPoint [31], where k is fixed at 1, VGLA maximizes the utilization of contextual information surrounding the GT box for prediction.

This method intuitively captures the features around the GT box. If a greater number of voxels are present within the GT box, more samples will be available to represent contextual information. By designing CUDA kernels, **Algorithm 1** can be implemented to enable the simultaneous processing of all GT Boxes and voxels within a batch.

Algorithm 1 Voxel-guided label assignment

```
Input: voxels coordinates $V = \{v_i \in \mathbb{R}^{N\times2}\}$, GT boxes $C = \{c_i \in \mathbb{R}^{M\times8}\}$, feature map $F \in [C, H, W]$
Output: the $k$ positive samples for every GT Box $c_i$.
Initialize the mask matrix $M = zeros(H, W)$
// step 1. Update $M$ based on if voxels are within $P$.
parallel for each $c_i \in C$ do
  define interest region $P$ on mask $M$ accounting to Eq. (4)
  parallel for each $v_i \in V$ do
    if $v_i \in P$, then
      $M(v_i) = 1$ accounting to Eq. (5)
    end if
  end parallel
// step 2. Extract the prediction results based on $M$ as the initial positive samples.
  Initialize IoUs to store the Intersection over Union between the predicted values and $c_i$.
  parallel for $(x_i, y_i) \in P$ do
    if $M(x_i, y_i) == 1$, then
      pred ← $F(:, x_i, y_i)$
      IoU ← IoU(pred, $c_i$)
      IoUs.append((IoU, $(x_i, y_i)$))
    end if
  end parallel
// step 3. Obtain the final positive samples according to the IoU value.
  sort IoUs in descending order by IoU
  Obtain the final number $k$ of positive samples for $c_i$ according to Eq. (6).
  return: the top k predicted values as the final positive samples.
end parallel
```

The PointPillars-VGLA framework as shown in Fig. 7, introduces a sophisticated methodology for point cloud object detection, commencing with the ingestion of raw point cloud data as input. The model's operational sequence is divided into several pivotal phases, initiating with the 3D backbone network where the Pillar Feature Net encodes and processes the point cloud data. This encoding strategy adeptly manages the inherent sparsity of point cloud data while ensuring computational efficiency.

Progressing through the framework, the 2D backbone network assumes control, initially translating the encoded features into a Bird's Eye View BEV representation. The 2D CNN backbone subsequently processes these BEV features, distilling extensive contextual information via multiple convolutional layers. This conversion from 3D to 2D space markedly simplifies the processing workflow while retaining essential spatial information.

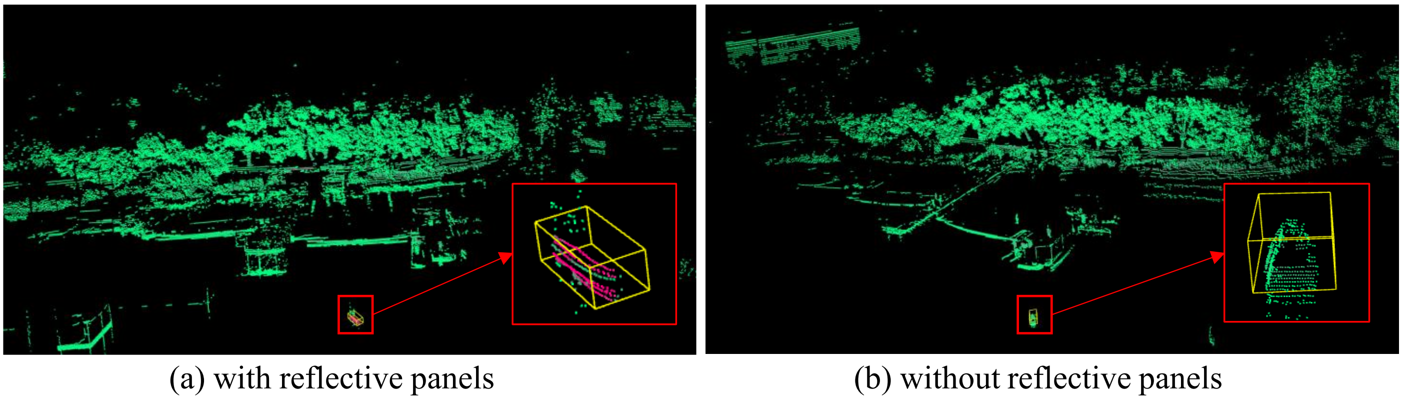
In the detection head, multiple feature layers are generated based on different dimensions of predicted targets. The predicted information for category, size, position, and orientation at that position is concatenated to form the prediction values in the feature map. VGLA focuses on using the projected positions of the GT Box and voxels on the feature map to calculate the IoU of the prediction values, thereby determining the positive samples.

5. Experiments

The experimental section is mainly divided into three parts. First, it validates the effectiveness of the data augmentation method based on reflective panels in improving object detection accuracy. Second, it evaluates the overall performance of PointPillars-VGLA on the dataset and designs ablation experiments to verify VGLA. Finally, the trained PointPillars-VGLA model is deployed on the Jetson AGX Orin using TensorRT, connected to real-time LiDAR data, to conduct object detection and tracking experiments in a real-world scenario.

5.1 Validation of the data augmentation method based on reflective panels

To validate the efficacy of the data augmentation technique utilizing reflective panels in enhancing object detection accuracy, we employ a selection of representative models. These models encompass a variety of backbone architectures, such as standard convolution, sparse convolution [32], and Transformer, along with diverse voxel encoding methods, including both voxel-based and pillar-based approaches. All models undergo training with uniform strategies and configurations, including batch size, number of epochs, and optimizer settings.



(a) with reflective panels (b) without reflective panels

Fig. 8 Dataset annotation process: The colors range from green to red, representing intensity values from 0 to 255

For dataset preparation, 300 frames are randomly selected from the same scene for training purposes, while 100 frames are reserved for validation. As illustrated in Fig. 8, it is evident that only the USV equipped with reflective panels exhibits red or light red hues in the entire scene, signifying that these regions possess markedly higher reflection intensity compared to other areas. Conversely, the remainder of the scene (such as the water surface and background) predominantly displays green or light green tones, indicating lower reflection intensity in these regions.

From the results, the Focals Conv [33] method performs the best in most cases, achieving a 3D AP of 77.59% without reflective panels and 82.29% with reflective panels, and a BEV AP of 86.33% and 89.00%, respectively. The DCDET [34] method slightly outperforms in the BEV AP metric in the scenario with reflective panels, reaching 89.36%. VoxelRCNN [35] shows relatively stable overall performance, while VoxelNext [32] has comparatively lower performance but still maintains a certain level of competitiveness.

These results indicate that the presence of reflective panels has a significant impact on detection performance, with all methods performing better in scenarios with reflective panels, Table 2.

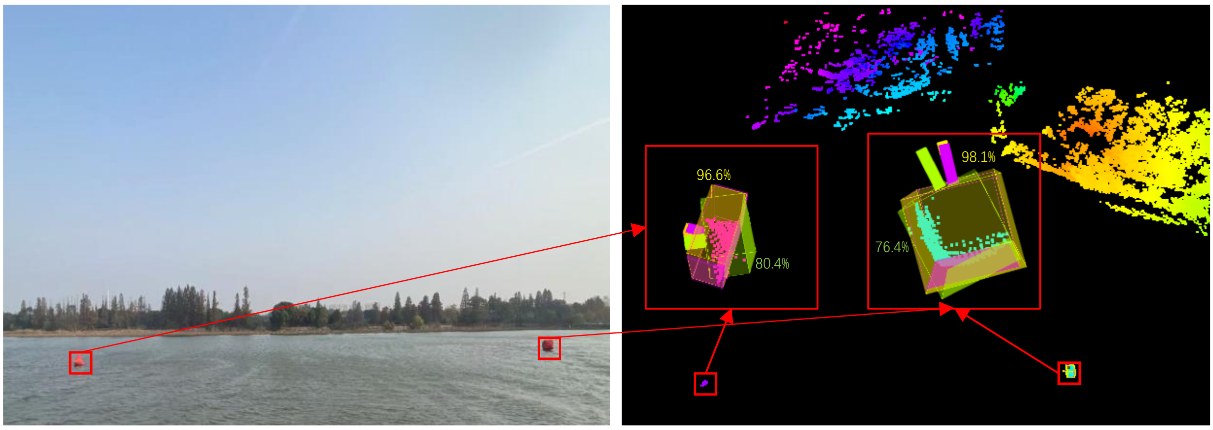
Table 2 Reflective panels augmentation method verification by 175 USV

Method	3D AP (%) (IoU = 0.7)		BEV AP (%) (IoU = 0.7)	
	No REF	With REF	No REF	With REF
VoxelRCNN	77.06	81.62	86.13	88.83
Focals Conv	77.59	82.29	86.33	89.00
DCDET	75.28	80.47	86.32	89.36
VoxelNext	74.36	77.35	84.51	87.11

5.2 PointPillars-VGLA implementation details

Given the custom dataset's range of [−30, 0, −3, 30, 60, 1], the scene is partitioned into squares with a voxel size of [0.5, 0.5] to optimize the window slicing operation, resulting in a planar window range of [600,600]. For the pillar version, the voxel length along the z-axis is set to 4. Prior to being fed into the backbone, the model enhances voxel features to 192 dimensions via the voxel feature encoder layer. In VGLA, the candidate region size for each GT box on the feature map is defined as 5. The model's classification and regression losses are implemented in the same manner as CenterPoint [31]. The model is deployed on OpenPCDet [36] for training and validation, utilizing a batch size of 8 and 80 epochs. For each category, the training and testing data are divided in a ratio of 4:1. Model evaluation criteria are based on those described in Section 3.2.

5.3 Main results



(a) Triangle and Cube Buoys in Camera view (b) Model Detection Results in Lidar view

Fig. 9 Point Cloud-Based Object Detection for Floating Targets: **Pink Box**: GT Box; **Yellow Box**: Predictions by PointPillars-VGLA; **Green Box**: Predictions by Standard PointPillars

From Table 3, although PV-RCNN is a more advanced detector, its complex structure makes it more difficult to deploy on embedded platforms for real-time applications. CenterPoint, on the other hand, does not outperform PointPillars on our dataset. Therefore, we chose PointPillars as the baseline for VGLA integration, considering both its competitive performance and practical deployment advantages in water-surface scenarios. It can be seen that the PointPillars-VGLA model performs the best on most metrics, achieving the highest scores in all scenarios for 3D AP and BEV AP. Specifically, in 3D AP, PointPillars-VGLA reaches 89.50% for the 175 USV targets, 83.70% for the Cube targets, and 75.20% for the Triangular targets. In BEV AP, it achieves 95.20%, 91.00%, and 86.70% in the 175 USV, Cube, and Triangular scenarios, respectively.

Fig. 9(a) shows the water-surface targets, while the point cloud detection results in Fig. 9(b) demonstrate that the PointPillars-VGLA model can accurately detect these targets with higher confidence scores (e.g., 98.1%, 96.6%). In summary, the advantages of the PointPillars-VGLA model are prominently reflected in its outstanding detection performance, which indicates that the PointPillars-VGLA model maintains excellent detection results across different scenarios, making it one of the top-performing methods currently available.

Table 3 Results on high-resolution water surface point cloud dataset.

Method	AP 3D (%)			AP BEV (%)		
	IoU = 0.7	IoU = 0.5		IoU = 0.7	IoU = 0.5	
	175 USV	Cube	Triangular	175 USV	Cube	Triangular
PV-RCNN	88.75	80.12	74.98	93.45	89.20	85.30
CenterPoint	82.50	73.60	71.80	89.00	84.50	82.90
PointPillars	85.80	77.50	74.20	91.00	87.50	85.60
PointPillars-VGLA	**89.50**	**83.70**	**75.20**	**95.20**	**91.00**	**86.70**

5.4 Validation of VGLA

In ablation experiments focusing solely on the dense head component, the widely recognized VoxelBackBone8x serves as the backbone to better emphasize performance enhancements. The batch size adjusts to 16, and the epoch count sets to 40, with training conducted on a single NVIDIA 4090 GPU while keeping all other parameters constant.

Table 4 Results on high-resolution water surface point cloud dataset for VGLA under BEV AP.

Strategy	175 USV	Cube	Triangular	Region size
	IoU = 0.7	IoU = 0.5	IoU = 0.5	
CenterPoint	94.88	93.33	90.86	1
DCDet	96.77	95.49	91.43	5
VGLA	95.03	92.86	90.92	3
VGLA	**96.92**	**95.60**	**91.52**	5
VGLA	95.33	94.15	90.55	7

The adoption of VGLA significantly boosts model performance by increasing the number of positive samples through diverse candidate region sizes for VGLA, as demonstrated by comparisons with other label assignment strategies.

VGLA compares with CenterPoint, which offers only one positive sample per GT box, and another label assignment method, DCDet, which provides multiple positive samples for the same GT box. As detailed in Table 4, with a candidate region size of 5, VGLA achieves the best performance, showing improvements of 0.15, 0.11, and a reduction of -0.11 compared to DCDet at the same region size. Notably, when the region size increases to 7, model performance declines. This occurs because the introduction of voxels outside the GT box results in lower-quality positive samples. Consequently, a candidate region size of 5 remains optimal.

5.5 Model deployment and field tests

To achieve real-time target detection for LiDAR, we train the PointPillars-VGLA model on the custom dataset and plan to deploy it on the Jetson AGX Orin 64 GB platform. 3D point cloud detection models typically involve a large number of parameters, and the complexity of processing point cloud data significantly exceeds that of traditional data formats such as images. As a result, the original model trained under the PyTorch framework struggles to meet the performance requirements for real-time detection (typically requiring 10 FPS). To address this challenge, we utilize the TensorRT deep learning inference library to convert the trained model into an efficient inference engine, thereby significantly improving the model's inference speed and computational efficiency and ensuring real-time target detection on resource-constrained embedded platforms.

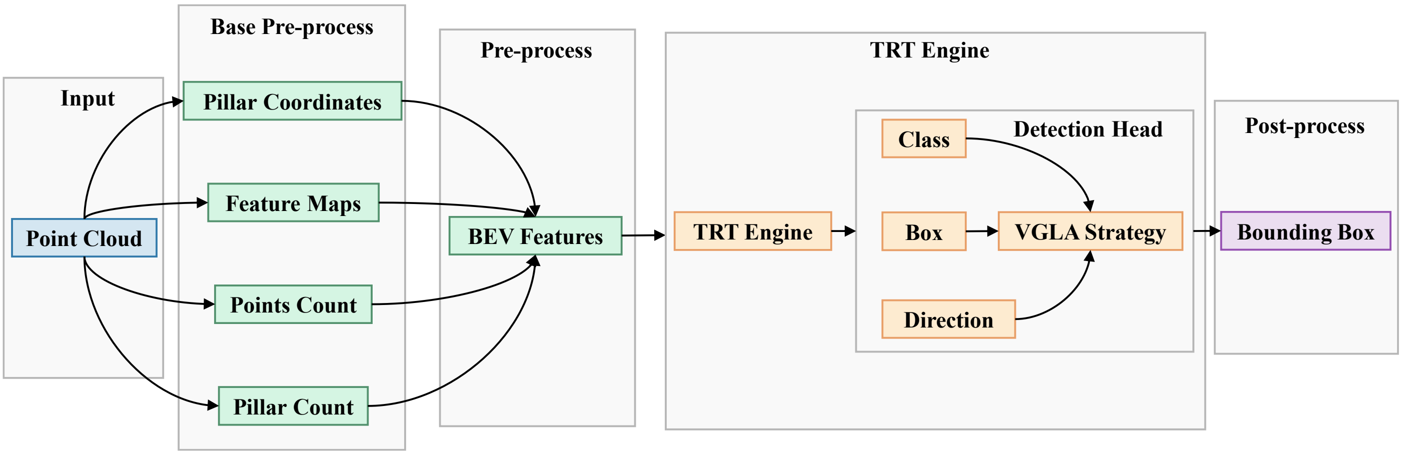
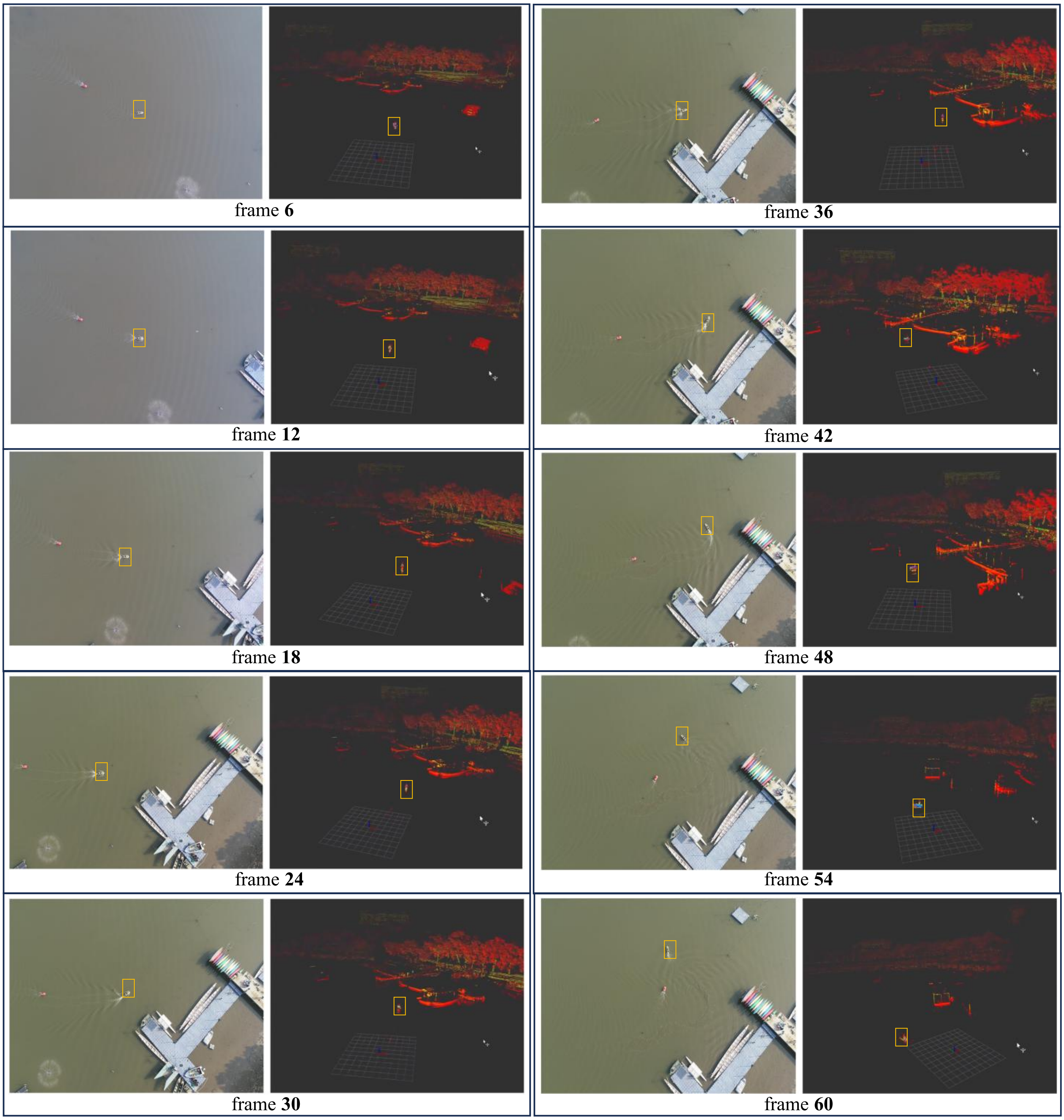


Fig. 10 Model Deployment Process

The deployment of this model is structured into four key stages, as illustrated in Fig. 10. Initially, basic preprocessing transforms point cloud data into pillar coordinates, base feature maps, point counts per pillar, and the total number of pillars, laying the groundwork for subsequent steps. Following this, the preprocessing stage produces bird's-eye view feature maps to further capture the spatial characteristics of the point cloud. These feature maps are then fed into the TensorRT engine for optimized inference. TensorRT significantly boosts efficiency by loading the ONNX model and supporting quantization methods such as FP16 and INT8. In our experiments, the model achieves real-time inference speeds of up to 30 FPS under both FP16 and INT8 precision on the Jetson AGX Orin platform, while the impact on detection accuracy is negligible. FP16 quantization simplifies computations by reducing floating-point precision with minimal accuracy loss, and INT8 quantization further minimizes computation and storage needs while maintaining inference precision through calibration. The Jetson AGX Orin operates within a configurable power envelope of 15W to 60W, making the model suitable for deployment on embedded platforms with similar power budgets. Lastly, post-processing generates the final bounding boxes for object detection and localization based on the inference results. Enhanced by TensorRT, this process not only accelerates inference but also reduces computational resource demands, ensuring the model's effectiveness in real-time applications on resource-constrained devices.

After converting the PointPillars-VGLA model to TensorRT, we utilize a USV equipped with LiDAR, as shown in Fig. 2, to detect and track the target labeled as the 175 USV in the custom dataset. Once the target is detected by PointPillars-VGLA, the information is sent to the downstream control system, which employs the LOS-PID control strategy for target tracking in the local coordinate system. To verify the robustness of PointPillars-VGLA in complex shoreline environments, we specifically design a route from far to near approaching the shoreline, as depicted in Fig. 11.



frame **6**

frame **12**

frame **18**

frame **24**

frame **30**

frame **36**

frame **42**

frame **48**

frame **54**

frame **60**

Fig. 11 Object Detection and Tracking Experiments in water surface Scenarios at 2m/s. The figure presents a 60-seconds video clip, showing corresponding bird's eye view and LiDAR view, with frames at 6, 12, 18, 24, 30, 36, 42, 48, 54, and 60

To transform the point cloud data into the vessel's coordinate system, it is necessary to determine the vertical height (H) and longitudinal distance (L) from the LiDAR coordinate center to the vessel center, which are set to 40 cm and 10 cm, respectively. During the experiments, the wind speed during the trials is approximately 2–3 m/s, and the current speed is about 1 m/s.

From this experiment, it can be observed that the model can accurately and continuously provide the relative position of the target USV in real time. The target is successfully detected and marked in most frames, with a detection success rate of over 90%. Even when the target USV is close to the shoreline, where there is interference from the shore, the detection model still demonstrates strong anti-interference capability.

Experimental results demonstrate that the designed detector effectively meets the requirements for dynamic target detection and tracking based on LiDAR, maintaining stable performance in complex shoreline environments with interference from the shoreline.

6. Conclusion

In this study, we present a comprehensive framework for water surface 3D point cloud object detection, integrating high-resolution datasets, innovative data augmentation strategies, and an improved PointPillars model enhanced with VGLA. The proposed data augmentation approach, utilizing reflective panels, has proven effective in enhancing both the quality and diversity of the training data. The incorporation of VGLA further refines the label assignment process, significantly improving the detection accuracy and robustness of the PointPillars architecture.

It should be noted that the custom dataset is primarily collected on a local lake, and the diversity of target types is limited in representativeness. As part of our future work, we plan to deploy the system in offshore and open-sea environments to enrich the dataset, including more vessel types and richer conditions like different waves and lightning. Additionally, we intend to investigate the integration of estimation-tracking algorithms to enhance the applicability and robustness of our framework in more challenging and dynamic aquatic environments.

ACKNOWLEDGEMENT

This work is supported by National Key R&D Program of China - Intergovernmental International Scientific and Technological Innovation and Cooperation Program under Grant 2023YFE0115400, National Natural Science Foundation of China under Grant 52131101, and in part by International Science and Technology Cooperation Project of Hubei Province (2023EHA024).

Dataset link: fx110127/The-high-resolution-128-beam-LiDAR-dataset

REFERENCES

[1] Li, J., Xiang, X., Zhang, Q., Yang, S., 2024. Robust practical prescribed time trajectory tracking of USV with guaranteed performance. *Ocean Engineering*, 302, 117622. https://doi.org/10.1016/j.oceaneng.2024.117622

[2] Liu, Z., Zhang, Y., Yu, X., Yuan, C. 2016. Unmanned surface vehicles: An overview of developments and challenges. *Annual Reviews in Control*, 41, 71-93. https://doi.org/10.1016/j.arcontrol.2016.04.018

[3] Xiang, X., Yu, C., Zhang, Q., 2017. Robust fuzzy 3D path following for autonomous underwater vehicle subject to uncertainties. *Computers & Operations Research*, 84, 165-177. https://doi.org/10.1016/j.cor.2016.09.017

[4] Guan, W., Xi, Z., Cui, Z., Zhang, X., 2025. Adaptive trajectory controller design for unmanned surface vehicles based on SAC-PID. *Brodogradnja*, 76(2). https://doi.org/10.21278/brod76206

[5] Wang, Z., Wei, Z., Yu, C., Cao, J., Yao, B., Lian, L., 2023. Dynamic modeling and optimal control of a positive buoyancy diving autonomous vehicle. *Brodogradnja*, 74(1), 19-40. https://doi.org/10.21278/brod74102

[6] Ahmed, F., Xiang, X., Jiang, C., Xiang, G., Yang, S., 2023. Survey on traditional and AI based estimation techniques for hydrodynamic coefficients of autonomous underwater vehicle. *Ocean Engineering*, 268, 113300. https://doi.org/10.1016/j.oceaneng.2022.113300

[7] Jiang, C., Tang, Y., Wang, J., Zhang, W., Zhou, M., Niu, J., Cheng, X., 2024. An optimized method for AUV trajectory model in benthonic hydrothermal area based on improved slime mold algorithm. *Brodogradnja*, 75(4), 1–25. https://doi.org/10.21278/brod75401

[8] Zhang, Y., Wang, H., Li, P., Chen, M., 2024. Small Modular AUV Based on 3D Printing Technology: Design, Implementation and Experimental Validation. *Brodogradnja*, 75(1), 1-16. https://doi.org/10.21278/brod75104

[9] Liu, C., Xiang, X., Huang, J., Yang, S., Zhang, S., Su, X., Zhang, Y., 2022. Development of USV autonomy: Architecture, implementation and sea trials. *Brodogradnja*, 73(1), 89-107. https://doi.org/10.21278/brod73105

[10] Wang, X., Xiang, X., Xiong, S., Yang, S., 2025. Position-based acoustic visual servo control for docking of autonomous underwater vehicle using deep reinforcement learning. *Robotics and Autonomous Systems*, 186, 104914. https://doi.org/10.1016/j.robot.2024.104914

[11] Liu, Z., Sun, J., Hu, Y., Zhao, T., 2024. Ship collision avoidance decision-making research in coastal waters considering uncertainty of target ships. *Brodogradnja*, 75(2), 1-16. https://doi.org/10.21278/brod75203

[12] Zong, C., Wan, Z., 2022. Container ship cell guide accuracy check technology based on improved 3D point cloud instance segmentation. *Brodogradnja*, 73(1), 23-35. https://doi.org/10.21278/brod73102

[13] Lang, A. H., Vora, S., Caesar, H., Zhou, L., Yang, J., Beijbom, O., 2019. Pointpillars: Fast encoders for object detection from point clouds. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA. https://doi.org/10.1109/CVPR.2019.01298

[14] Dai, D., Chen, Z., Bao, P., Wang, J., 2021. A review of 3D object detection for autonomous driving of electric vehicles. *World Electric Vehicle Journal*, 12(3), 139. https://doi.org/10.3390/wevj12030139

[15] Zhou, Z., Sun, J., Yu, J., Liu, K., Duan, J., Chen, L., Chen, C. P., 2021. An image-based benchmark dataset and a novel object detector for water surface object detection. *Frontiers in Neurorobotics*, 15, 723336. https://doi.org/10.3389/fnbot.2021.723336

[16] Cheng, Y., Jiang, M., Zhu, J., Liu, Y., 2021. Are we ready for unmanned surface vehicles in inland waterways? The USV Inland multisensor dataset and benchmark. *IEEE Robotics and Automation Letters*, 6(2), 3964-3970. https://doi.org/10.1109/LRA.2021.3067271

[17] Cheng, Y., Zhu, J., Jiang, M., Fu, J., Pang, C., Wang, P., Bengio, Y., 2021. Flow: A dataset and benchmark for floating waste detection in inland waters. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada. https://doi.org/10.1109/ICCV48922.2021.01077

[18] Vinodkumar, P. K., Karabulut, D., Avots, E., Ozcinar, C., Anbarjafari, G., 2024. Deep learning for 3D reconstruction, augmentation, and registration: a review paper. *Entropy*, 26(3), 235. https://doi.org/10.3390/e26030235

[19] Charles, R., Su, H., Kaichun, M., Guibas, L. 2027. PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation. *2017 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA. https://doi.org/10.1109/CVPR.2017.16

[20] Yan, Y., Mao, Y., Li, B., 2018. SECOND: Sparsely embedded convolutional detection. *Sensors*, 18(10), 3337. https://doi.org/10.3390/s18103337

[21] Zhou, Y., Chen, H., Gao, L., Li, G., Chen, Y., 2025. An automatic method of siltation depth detection and 3D modelling in water-filled sewer pipelines based on sonar point clouds. *Measurement*, 242, 115954. https://doi.org/10.1016/j.measurement.2024.115954

[22] Mao, J., Xue, Y., Niu, M., Bai, H., Feng, J., Liang, X., Xu, C., 2021. Voxel transformer for 3D object detection. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada. https://doi.org/10.1109/ICCV48922.2021.00315

[23] Yang, W., Yu, H., Luo, X., Xie, S., 2024. Geometric relation-based feature aggregation for 3D small object detection. *Applied Intelligence*, 54(19), 8924-8938. https://doi.org/10.1007/s10489-024-05342-z

[24] Guo, Y., Yu, H., Ma, L., Zeng, L., Luo, X., 2023. THFE: A Triple-hierarchy Feature Enhancement method for tiny boat detection. *Engineering Applications of Artificial Intelligence*, 123, 106271. https://doi.org/10.1016/j.engappai.2023.106271

[25] Ma, R., Chen, C., Yang, B., Li, D., Cong, Y., Hu, Z., 2022. CG-SSD: Corner guided single stage 3D object detection from LiDAR point cloud. *ISPRS Journal of Photogrammetry and Remote Sensing*, 191, 33-48. https://doi.org/10.1016/j.isprsjprs.2022.07.006

[26] Shi, S., Guo, C., Jiang, L., Wang, Z., Shi, J., Wang, X., Li, H., 2019. PV-RCNN: Point-voxel feature set abstraction for 3D object detection. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA. https://doi.org/10.1109/CVPR42600.2020.01054

[27] Zhang, Z., Sun, B., Yang, H., Huang, Q., 2029. H3DNet: 3D object detection using hybrid geometric primitives. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA. https://doi.org/10.1007/978-3-030-58610-2_19

[28] Geiger, A., Lenz, P., Urtasun, R., 2012. Are we ready for autonomous driving? The KITTI vision benchmark suite. *2012 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR),* Providence, RI, USA. https://doi.org/10.1109/CVPR.2012.6248074

[29] Li, Z., Lan, S., Alvarez, J. M., Wu, Z., 2024. BEVNext: Reviving dense BEV frameworks for 3D object detection. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA. https://doi.org/10.1109/CVPR52733.2024.01901

[30] Li, Y., Guan, K., Hu, Z., Chen, Y., 2016. An optical fiber lateral displacement measurement method and experiments based on reflective grating panel. *Sensors*, 16(6), 808. https://doi.org/10.3390/s16060808

[31] Yin, T., Zhou, X., Krahenbuhl, P., 2021. Center-based 3D object detection and tracking. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA. https://doi.org/10.1109/CVPR46437.2021.01161

[32] Chen, Y., Liu, J., Zhang, X., Qi, X., Jia, J., 2023. VoxelNext: Fully sparse VoxelNet for 3D object detection and tracking. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Vancouver, Canada. https://doi.org/10.1109/CVPR52729.2023.02076

[33] Chen, Y., Li, Y., Zhang, X., Sun, J., Jia, J., 2022. Focal sparse convolutional networks for 3D object detection. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, New Orleans, LA, USA. https://doi.org/10.1109/CVPR52688.2022.00535

[34] Liu, S., Li, B., Fang, Z., Huang, K., 2024. DCDet: Dynamic cross-based 3D object detector. *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, 1119-1127. https://doi.org/10.24963/ijcai.2024/124

[35] Deng, J., Shi, S., Li, P., Zhou, W., Zhang, Y., Li, H., 2021. Voxel R-CNN: Towards high-performance voxel-based 3D object detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(2), 1201-1209. https://doi.org/10.1609/aaai.v35i2.16207

[36] OpenPCDet Development Team, 2020. OpenPCDet: An Open-source Toolbox for 3D Object Detection from Point Clouds. https://github.com/open-mmlab/OpenPCDet. accessed 25th February 2025.