2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

AnytimeFusion: Parameter-free RGB Camera-Radar Sensor Fusion Algorithm in Complex Maritime Situations

Yeongha Shin[1], Hanguen Kim[2], and Jinwhan Kim[3]

***Abstract*— Determining the position of obstacles is crucial for unmanned vehicles, and, to achieve this, cameras and radar sensors are widely utilized. However, establishing correlation between two or more sensors proves challenging in the dynamically changing maritime environment. To solve these issues, we propose the AnytimeFusion algorithm. The key innovation of AnytimeFusion lies in the utilization of a parameter-free method that does not require accurate sensor alignment and calibration. The algorithm consists of the following four stages. First, calibration targets are selected in the maritime environment based on segmentation images. Second, radar and camera data are pre-fused to model the correlation of azimuth information. After completing the auto-calibration stages, Inverse Perspective Mapping (IPM) is employed to integrate the coordinate systems of the two sensors. To determine the parameters for this integration, optimization based on the Particle Swarm Optimization (PSO) method is employed. Finally, an Error Polygon for the positions of the camera and radar is generated, and sensor fusion is carried out based on this information. We validated our method through experiments conducted on real ships in complex maritime environments, achieving an average accuracy of 95.7%.**

I. INTRODUCTION

The interest in autonomous surface vehicles has significantly increased in the last few years [1]. Detecting objects in the vicinity is crucial for autonomous vehicles to avoid obstacles and follow a path [2], [3]. In maritime environments, radar sensors are widely employed for object detection, as they enable the acquisition of precise distance and azimuth information [4]. However, radar sensors have the drawback of lacking information about height and facing challenges in acquiring semantic information. In contrast, although camera sensors are widely used in the maritime environment [5], they have limitations such as inadequate robustness in harsh weather and the inability to reliably obtain depth information [6]. Therefore, sensor fusion technology, which integrates the features of several sensors to generate accurate information, is currently being studied as an effort to address these challenges [7], [8].

There are challenges in integrating multi-sensor fusion due to the inherent differences between radar and RGB camera sensors. Radar lacks height information, while RGB cameras lack depth information. Additionally, cameras and radars have different levels of azimuth precision [9]. As cameras

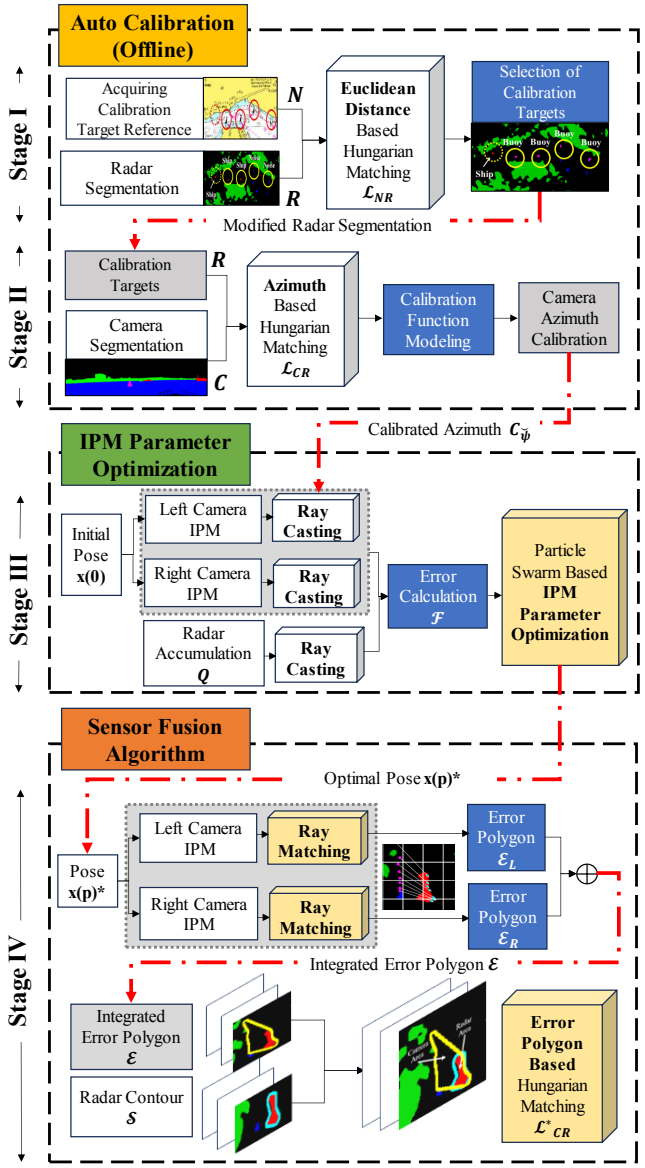


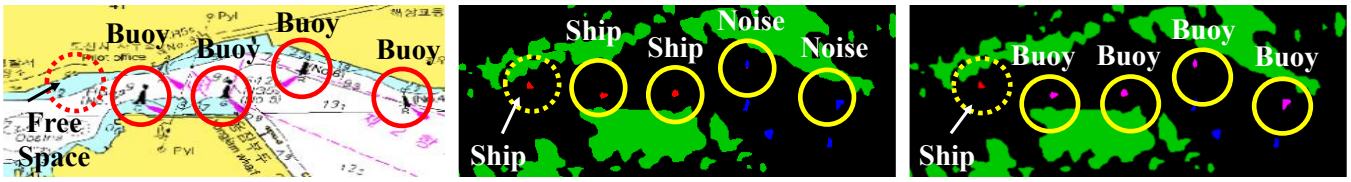
Fig. 1: Overview of AnytimeFusion: the algorithm comprises four stages. Auto-calibration for azimuth information is divided into Stage I and Stage II. In Stage III, the coordinate systems of the camera and radar are integrated through IPM parameter optimization. In Stage IV, camera-radar fusion is conducted based on the Error Polygon.

*This work was not supported by any organization.

[1]Yeongha Shin was with Seadronix Corp, South Korea, at the time of submission. She is currently with the Robotics Program, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea. yeongha.shin@kaist.ac.kr

[2]Hanguen Kim is the CTO of Seadronix Corp., Seoul 06235, South Korea. hank05@seadronix.com

[3]Jinwhan Kim is with the Department of Mechanical Engineering, Korea Advanced Institute of Science and Technology, Daejeon 34141, South Korea. jinwhan@kaist.ac.kr



(a) Calibration Target Reference (b) Radar Segmentation Result (c) Selection of Calibration Targets

Fig. 2: Stage I) Example of selecting calibration targets. After obtaining reference information in (a), matching is performed with radar objects via Hungarian matching in (b), resulting in the supplementation of class information for calibration targets, as shown in (c).

and radars differ in dimensions, various methods have been employed to combine their coordinate systems. The methods for sensor fusion can be classified into two categories: 1) conducting the transformation with prior knowledge of the correlation information between cameras and radars, or 2) estimating the transformation without prior knowledge of the correlation information.

In the approach where correlation information is known, RGB cameras are converted using the IPM technique [10] or directly transformed into world coordinates. However, these approaches have limitations, as they require precise installation and pose information. Obtaining accurate pose information for a ship using sensors such as Inertial Measurement Units (IMU) poses challenges due to accumulated errors over prolonged movements [11].

On the other hand, in the approach where the correlation information between the two sensors is unknown, research has focused on identifying optimal combinations of roll and pitch to ensure the best matching between the two sensors [4]. However, to determine such combinations, the approach requires the selection of objects detected by both sensors as targets. This approach faces limitations when applied to complex maritime environments with a large number of objects, potentially introducing errors. Furthermore, studies have been carried out to extract the horizon line from cameras to obtain depth information [12]. However, accurately acquiring the horizon line in the marine port environment, which is the focus of this study, poses challenges due to ships with irregular shorelines and unpredictable land-sea boundaries.

In this paper, we propose AnytimeFusion, a parameter-free RGB camera-radar sensor fusion algorithm designed for complex maritime environments. The core idea involves conducting Auto Calibration to integrate two sensors with differences in azimuth accuracy. Subsequently, optimization of IPM parameters is performed to fuse RGB camera and radar information, especially in situations where depth information is unavailable in the RGB camera. Finally, an Error Polygon is generated to compensate for potential errors introduced in the previous stages.

The contributions of the proposed algorithm include:

- The ability to perform fusion even without precise correlation between sensors or accurate knowledge of the ship's movement.
- The capability to proceed with fusion despite errors in segmentation and classification or even if detected by a single sensor.
- The capacity to achieve accurate matching in complex maritime scenarios with a large number of objects.

II. AnytimeFusion

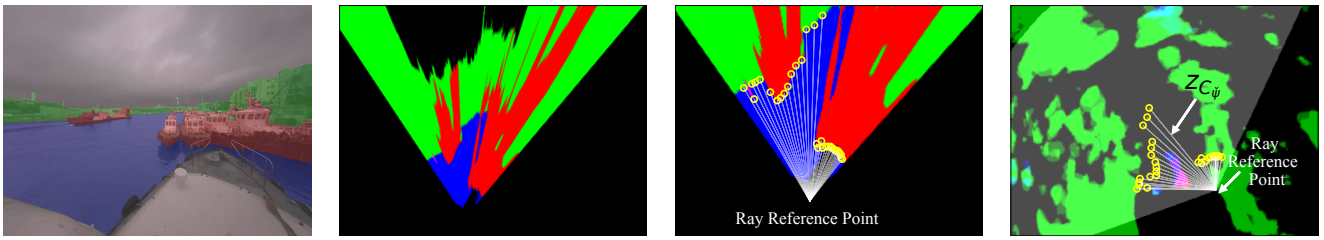
The overview of AnytimeFusion is illustrated in Fig 1. This proposed algorithm consists of four stages: Stage I) selection of calibration targets, Stage II) pre-fusion of camera and radar for azimuth information modeling, Stage III) IPM parameter optimization for the integration of camera and radar coordinates, and Stage IV) fusion of camera and radar based on an Error Polygon. In the following sections, we present the details of each stage in AnytimeFusion.

A. Auto Calibration (Stage I, II)

Calibrating azimuth information is essential to establish the correlation between the camera and radar. This study assumes the use of camera results obtained through intrinsic calibration. However, relying solely on intrinsic calibration may not accurately capture azimuth information, as it can change during the sensor installation process on real ships due to differences in installation locations or ship attitude. Hence, the Auto Calibration process becomes essential to address these factors.

The Auto Calibration procedure consists of two steps. In the first stage, the selection of calibration targets is conducted. The second stage involves the fusion of camera and radar calibration objects to model azimuth information.

1) **Camera & Radar Segmentation:** For object recognition, a segmentation model based on deep learning was applied to data acquired from both the camera and radar sources. These models are primarily designed to identify marine objects, even if they are very small. Specifically, the camera model is configured to provide outputs with five distinct classes: land, ship, buoy, sky, and sea surface [13]. Concurrent research has introduced a robust model capable of handling weak illumination and haze conditions that impair camera sensors. [14] AnytimeFusion allows for integrating such models into camera segmentation, enabling the system to address weak illumination and adverse weather conditions. Meanwhile, the radar sensor model is structured to determine results categorized into three classes: land, object (regardless of the object's type, such as ships, buoys, etc.), and noise [15].



(a) Camera Segmentation (b) Camera IPM Result (c) Ray Casting on IPM (d) Camera Ray on Radar Domain

(e) Radar Segmentation (f) Accumulated Radar Image (g) Ray Casting on Radar (h) Error Calculation

Fig. 3: Stage III) Optimization process of IPM parameters. For the camera, Ray Casting is performed after IPM to measure distances on obstacle surfaces in (b)-(c). The radar conducts Ray Casting after accumulating multiple frames in (f)-(g). In (h), the parameters minimizing the comparison between the Ray Casting results of the camera and radar are identified.

2) **Selection of Calibration Targets (Stage I):** To perform Auto Calibration, selecting calibration targets is essential. In this study, we chose buoys—small, narrow-width objects in the camera's field of view—as the calibration targets. The reason for selecting buoys as calibration targets is to achieve accurate azimuth information. Furthermore, as they are objects fixed underwater, buoys have been chosen as stable calibration targets that can be acquired in a maritime environment. However, depending on changes in radar resolution or a switch to an RGBD camera or if used in a land environment, a different approach to selecting calibration targets can be employed.

In this paper, the radar segmentation model does not include a class for buoys, and there may be errors in the classification of the existing class, as shown in Fig 2(b). Therefore, additional processing is required to select accurate calibration targets. To address this, we first obtained the reference information of calibration targets from the nautical chart, as shown in Fig 2(a). The Hungarian matching technique [16] was then used to match these nautical chart objects $N = n_1, \ldots, n_i \in \mathbb{R}^2$ with radar objects $R = r_1, \ldots, r_j \in \mathbb{R}^2$ using Euclidean distance $d(\mathbf{n_i}, \mathbf{r_j}) = \sqrt{(\mathbf{n_i}_x - \mathbf{r_j}_x)^2 + (\mathbf{n_i}_y - \mathbf{r_j}_y)^2}$. This allowed us to accurately supplement the class information, as illustrated in Fig 2(c). Meanwhile, the original ship class—not calibration targets—is still applicable in areas without buoys. The error cost function of the Hungarian matching, denoted as $\mathcal{L}_{NR}$, is represented by the equation:

$$\mathcal{L}_{NR} = argmin \sum_{i=1}^{N} \sum_{j=1}^{R} d(\mathbf{n}_i, \mathbf{r}_j) \tag{1}$$

3) **Calibration Function Modeling (Stage II):** In this stage, pre-fusion of camera objects in camera coordinates $\mathbb{C} = u, v$ and radar is conducted to model the correlation of azimuth information.

To achieve the initial alignment of azimuth between the camera and radar, the normalized camera's azimuth $c_{\hat{\psi}}$ is calculated based on the camera's image width w and horizontal field of view (FOV) $FOVh$. On the other hand, the radar object's azimuth information $r\psi_i$ is calculated from the radar center position (position of ownship) $r_o = (r_{o_x}, r_{o_y})$. Subsequently, the azimuth differences between radar objects $R = r_1, \ldots, r_j \in \mathbb{R}^2$ and camera objects $C = c_1, \ldots, c_k \in \mathbb{C}$ are utilized for the application of the Hungarian matching error cost function $\mathcal{L}_{CR}$.

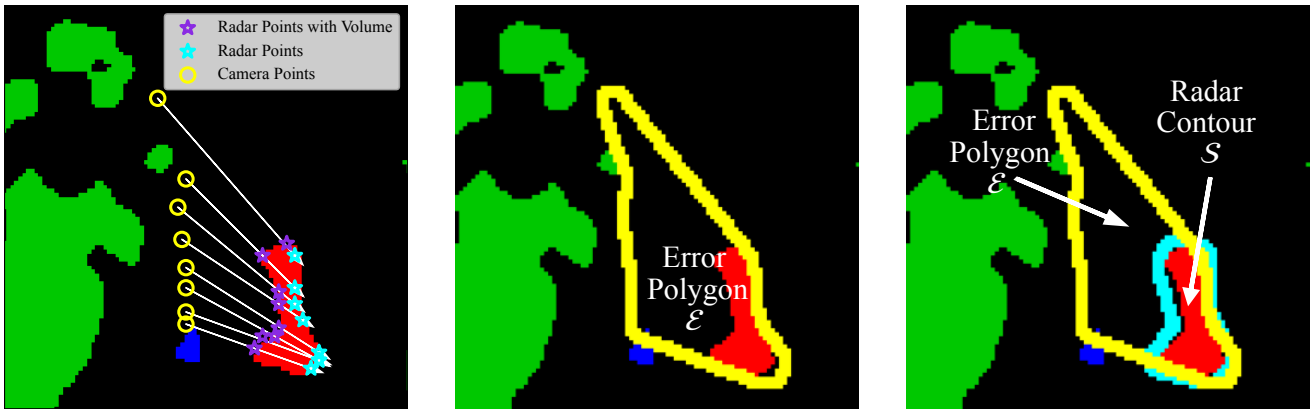
$$c_{k_{\hat{\psi}} = \frac{c_{k_u}}{w} \times FOV_h} \tag{2}$$

$$r_{j_\psi} = \arctan\left(\frac{r_{j_y} - r_{o_y}}{r_{j_x} - r_{o_x}}\right) \tag{3}$$

$$\mathcal{L}_{CR} = argmin \sum_{k=1}^{C} \sum_{j=1}^{R} |\mathbf{c}_{k_{\hat{\psi}}} - \mathbf{r}_{j_\psi}| \tag{4}$$

The reason for performing matching based on azimuth information is that the camera is not installed at a high position, making it difficult to extract meaningful height-direction information from the camera image.

During the entire scenario T, matched objects are stored in $M = m_1, \ldots, m_l$, where $m_l = (c_k(u, v), r_j(x, y))$, and a process is undertaken to identify the most suitable representative function for these objects. To address outliers, a method based on standard deviation is employed, filtering out objects for which the difference exceeds a certain standard deviation σ during line fitting with the Least Squares Method [17] between c_{m_u} and r_{m_ψ}. With this method, objects with unsuccessful matches are automatically separated out, and only



(a) Ray Matching on Radar Domain (b) Error Polygon Generation (c) Matching based on Error Polygon

Fig. 4: Stage IV) The principle of generating Error Polygons and performing Sensor Fusion. In (a), Ray Marching is carried out starting from the Camera IPM position in the radar domain, allowing the volume of radar obstacles to contribute to the creation of an Error Polygon. Consequently, the Error Polygon is generated, as shown in (b). In (c), Sensor Fusion is performed by assessing the overlap between the generated Error Polygon and the segmentation contour of the radar.

successfully matched objects are taken into consideration for line fitting, as shown in Eq (5). A 4th-degree polynomial is chosen for calibration modeling, providing adaptability to complex distortion patterns in real-world scenarios.

$$argmin \sum_{m=1}^{l} \left(\alpha {c_{m_u}}^4 + \beta {c_{m_u}}^3 + \gamma {c_{m_u}}^2 + \delta c_{m_u} + \epsilon - r_{m_\psi}\right)^2 \tag{5}$$

The camera width-direction is divided into intervals to identify the Auto Calibration termination point and evaluate the efficacy of the created model. The calibration procedure continues if there aren't enough objects within a given interval; if there are more objects above a predetermined threshold, the calibration process is terminated. This eliminates line-fitting errors caused by a biased object distribution and guarantees a balanced distribution of objects across intervals.

B. IPM Parameter Opimization (Stage III)

The sensor fusion of RGB cameras and radar requires the integration of coordinate systems, especially with RGB cameras when depth information is not accessible. The goal of this stage is to combine coordinate systems using the IPM approach.

However, finding the parameters used in IPM is a challenging task. This is because obtaining accurate installation information during the sensor installation process is difficult, and accurately determining pose information is challenging due to the movement of the ship and changes in the maritime environment. Therefore, in this stage, the optimization process is explained to obtain information regarding these parameters.

1) **Error function of Optimization:** The error function for the optimization algorithm relies on the Ray Casting technique. Ray Casting involves projecting virtual rays from a reference point to each pixel to ascertain the presence of obstacles [18].

For the camera, initialized at $\mathbf{x}(\mathbf{0}) = (\phi_0, \theta_0, \psi_0) = (0, 0, 0)$, where $\mathbf{x}$ represents roll, pitch, and yaw, IPM transformed the camera image into an integrated coordinate system with the radar, as shown in Fig 3(b). Subsequently, surfaces corresponding to obstacles in the camera image were detected through Ray Casting along the camera rays' range $z_{C_{\check{\psi}}}$ within the calibration camera's FOV $F\check{O}V_h$. As illustrated in Fig. 3(d), this procedure established the initial distance values for obstacles in the camera images. In the radar image, an Accumulation step preceded Ray Casting. If only a single frame of the radar is used, it cannot address classification errors of objects, as shown in Fig 3(e). As depicted in Fig. 3(f), an accumulation queue $Q = q_1, \ldots, q_m$ was utilized to create a cumulative image to reduce misclassification due to changes in pose, moving obstacles, or inaccuracies in the segmentation model through weighted average.

To find the distances to obstacles, the radar ray range $z_{R_{\check{\psi}}}$ was determined after the Accumulation step. The obstacle distance values for camera and radar objects were computed on a per-ray basis, with their sum serving as the error function for the optimization algorithm of a certain particle $\mathcal{F}(\mathbf{x}(\mathbf{p}))$ in Eq (6).

$$\mathcal{F}(\mathbf{x}(\mathbf{p})) = \sum_{\check{\psi}=1}^{F\check{O}V_h} |z_{C\check{\psi}} - z_{R\check{\psi}}| \tag{6}$$

2) **Particle Swarm based IPM Parameter Optimization:** We employed the particle swarm optimization (PSO) approach to determine the IPM parameters [19]. PSO is renowned for its proficiency in global optimization, utilizing a swarm-based methodology for effective exploration. Through this algorithm, particles $P = \mathbf{p}_1, \ldots, \mathbf{p}_n$ with random values for roll, pitch, and yaw $\mathbf{x}(\mathbf{p})$ are generated, and

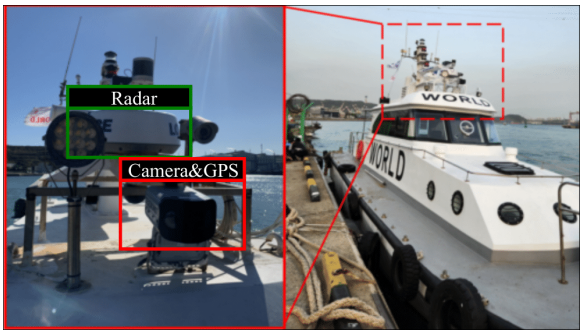


Fig. 5: Sensor Configuration.

TABLE I: A Detail of Data Set.

Contents	Set 1	Set 2	Set 3
Frame	223	88	127
Camera Objects (Left/ Right)	776/577	185/386	466/355
Radar Objects	592	**1664**	827
Matched Object Pairs	147	162	133
Actual Objects (Dynamic,Small)	11(2,5)	14(3,2)	6(2,**6**)

the optimization process is repeated until the optimal IPM parameters $\mathbf{x}(\mathbf{p})^*$ are found. A threshold value is set during the optimization process to prevent convergence to overly specific values.

$$\mathbf{x}(\mathbf{p})^* = \arg\min_{\mathbf{p}\in P} \mathcal{F}(\mathbf{x}(\mathbf{p})) \tag{7}$$

C. Sensor Fusion Algorithm (Stage IV)

1) **Error Polygon Generation:** The concept of an Error Polygon was introduced in this stage. Despite completing Auto Calibration to enhance the alignment of azimuth information between the camera and radar, followed by coordinate integration through IPM optimization, inaccuracies may still remain. In order to address this, for each camera object, an Error Polygon is generated by connecting camera ray range $z_{C_{\check{\psi}}}$ and radar ray range $z_{R_{\check{\psi}}}$ in Fig 4(a). However, in this stage for radar obstacles, a Ray Marching [20] method from the camera IPM point was applied rather than Ray Casting. As shown in Fig. 4(a), Ray Marching was chosen to precisely capture the volume information of radar v_R and ensure an accurate representation of obstacle positions. Thus, the final result includes the generation of an Error Polygon $\mathcal{E}$ for the camera object's bounding box, as depicted in Fig. 4(b) and Eq (8).

$$\mathcal{E} = \bigcup_{\check{\psi}=1}^{\text{bounding box width}} (z_{C_{\check{\psi}}}, z_{R_{\check{\psi}}}, v_R) \tag{8}$$

2) **Hungarian Matching based on Error Polygon:** The Error Polygon was utilized as the cost function for the Hungarian Matching algorithm. This stage is inspired by the Intersection over Union (IoU) method [21]. We employed a modified concept, as expressed in Eq (9). This modification deviates from the traditional IoU method by considering the union $\mathcal{E} \cup \mathcal{S}$ of the Error Polygon $\mathcal{E}$ and the segmentation contour area of the radar $\mathcal{S}$. We select the highest proportion among the IoU and differences of the Error Polygon and radar $\mathcal{E} \cap \mathcal{S}, \mathcal{E} - \mathcal{S}, \mathcal{S} - \mathcal{E}$. This adjustment accommodates the possibility of the radar object forming an inclusion relationship with the Error Polygon, as illustrated in Fig 4(c). Before integrating the camera and radar, if multiple cameras are installed and their FOVs overlap, pre-fusion is performed among the cameras to consolidate regions where the FOVs intersect into a single object. Subsequently, the final fusion is executed between the integrated camera objects and the radar based on Eq (9).

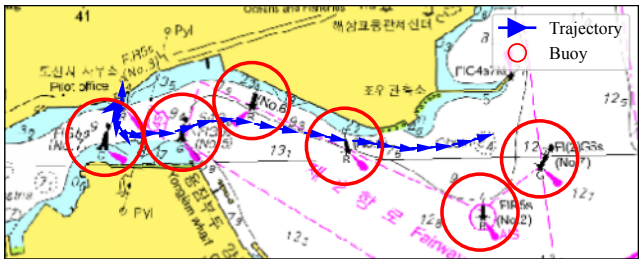


Fig. 6: The trajectory of the dataset used for Auto Calibration and the positions of calibration targets, which are buoys, are depicted. Translation and rotation were applied to ensure a uniform distribution along the camera width directions.

$$\mathcal{L}^*_{CR} = argmax \sum_{k=1}^{C} \sum_{j=1}^{R} max \left(\frac{\mathcal{E}_k \cap \mathcal{S}_j}{\mathcal{E}_k \cup \mathcal{S}_j}, \frac{\mathcal{E}_k - \mathcal{S}_j}{\mathcal{E}_k \cup \mathcal{S}_j}, \frac{\mathcal{S}_j - \mathcal{E}_k}{\mathcal{E}_k \cup \mathcal{S}_j} \right) \tag{9}$$

III. EXPERIMENTS

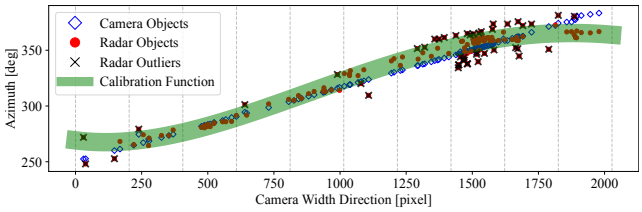
To evaluate the accuracy of our proposed algorithm, we conducted experiments in a complex maritime environment with real ships. The experiments were conducted in three stages: Auto Calibration, IPM parameter optimization, and the Sensor Fusion algorithm.

The ship used in the experiments is depicted in Fig. 5, equipped with cameras, radar, and a GPS sensor installed at the top of the mast to acquire environmental information. Two cameras were fitted, extending the camera's 135-degree field of view to 220 degrees. The radar's 500-meter detection radius was also adjusted. The cameras, radar, and GPS collected data at 1Hz intervals.

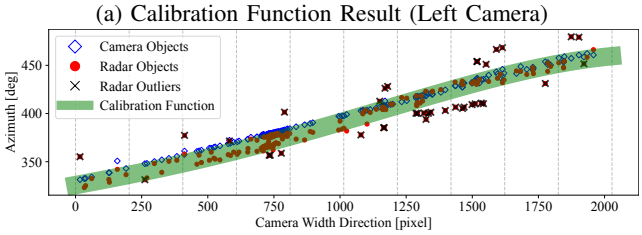
A. Dataset

To conduct the experiment, data collection was performed at Ulsan Port in South Korea. Table I provides a description of the dataset used in the experiments, including the total number of actual objects throughout the entire scenario. The analysis distinguishes between dynamic objects and small objects, such as buoys.

Analyzing each dataset separately, Set 1 is the most straightforward to use, with a minimal likelihood of inaccurate matches, as it possesses an almost equal number of camera and radar objects. In contrast, Set 2 has a higher

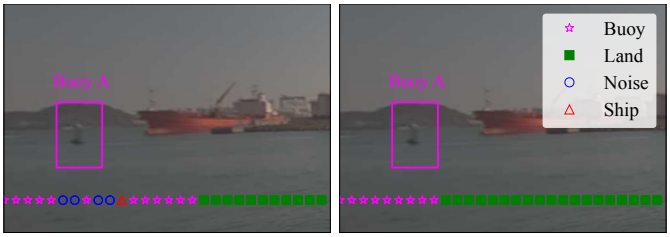


(a) Calibration Function Result (Left Camera)

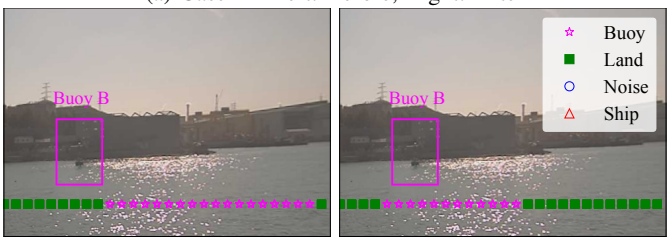


(b) Calibration Function Result (Right Camera)

Fig. 7: The results of the calibration function. By applying a pre-fusion of radar and camera, a function was developed to enhance the camera's azimuth accuracy. Outliers were excluded during the line fitting process to eliminate inaccuracies in the matching results.



(a) Case 1 - Left: Before, Right: After



(b) Case 2 - Left: Before, Right: After

Fig. 8: Before-and-after results show improved camera azimuth accuracy through Auto Calibration. Initially, there were discrepancies in class azimuth between the camera and radar segmentation. After correction, the results aligned with the radar's azimuth information.

proportion of radar objects than camera objects, making it more challenging due to the potential for object fusion. Set 3, characterized by the largest percentage of small objects, showcases the most noticeable impact of Ray.

B. Auto Calibration

Firstly, the Auto Calibration experiment was conducted to assess the improvement in the accuracy of the camera's azimuth information.

1) **Experiment Setups:** Set 3, which has the highest proportion of buoys set as calibration targets for Auto

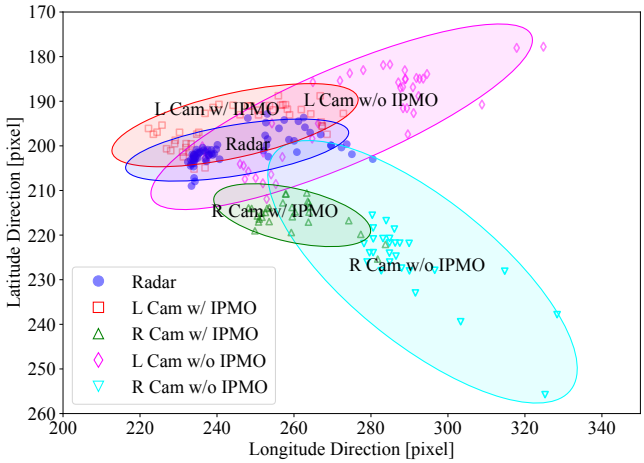


Fig. 9: Comparison Before and After IPM Optimization: By comparing the distribution of obstacle positions, it is evident that, after applying IPM Optimization, the results closely resemble the radar's distribution compared to before optimization.

Calibration, was selected as the experimental scenario. The trajectory and positions of buoys are illustrated in Fig 6. To collect data that is unbiased in the width direction of the coordinate system for the left and right cameras, both rotation and translation movements were uniformly conducted. Furthermore, a threshold of 30 degrees was set for azimuth-based Hungarian matching to prevent inaccurate matches.

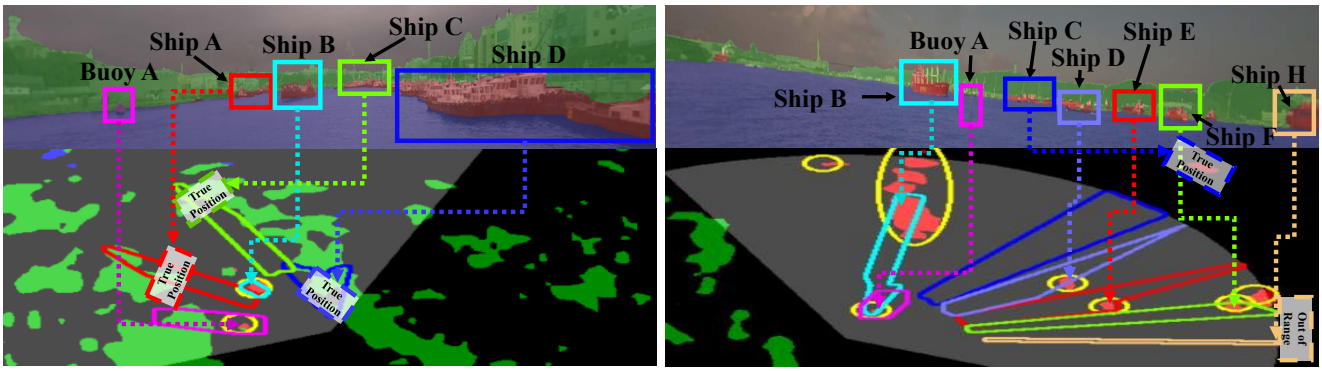
2) **Experiment Results:** The information about the calibration model obtained through this experiment is depicted in Fig 7. The left and right cameras were separated to establish azimuth calibration models. The camera width direction was divided into 10 sections, ensuring that five or more cumulative frequencies were satisfied. Once termination conditions were met, the Auto Calibration successfully concluded, and the model creation was completed.

The results show that the models for the left and right cameras are not exactly the same. This discrepancy is attributed to the ratio of inliers to outliers, with the left camera having a ratio (outlier/inlier) of 45/99, and the right camera of 37/145. This suggests that the right camera exhibited a more representative outcome for the modeling. However, compensations for such errors can be addressed in the IPM optimization and Error Polygon creation stages.

The results obtained by applying the azimuth calibration model are shown in Fig 8. Before the calibration, there was a discrepancy between the position where buoys were indicated and the class information of the radar image corresponding to this camera's azimuth. However, after performing the calibration, it was confirmed that the rays corresponding to buoy information and the radar information were in alignment.

C. IPM Parameter Optimization

In the second experiment, the accuracy of IPM optimization is assessed by contrasting the outcomes of initial settings of $\mathbf{x}(\mathbf{0})$ = (0, 0, 0) with optimal parameter of IPM method



(a) Example of Matching Results (Set 1)

(b) Example of Matching Results (Set 2)

Fig. 10: Qualitative Results of Sensor Fusion Algorithm. The algorithm successfully matches objects in complex maritime environments, complementing true obstacle positions using the Error Polygon, regardless of radar segmentation accuracy.

TABLE II: Results of Ablation Study

AC	IPMO	Set	Accuracy↑	FP↓	FN↓	ID Switch↓
X	X	1	48.29% (71/147)	18	39	37
		2	3.70% (6/162)	27	77	6
		3	10.52% (14/133)	21	64	54
X	O	1	48.19% (71/147)	18	35	41
		2	4.93% (8/162)	38	65	89
		3	7.51% (10/133)	24	68	53
O	X	1	91.83% (135/147)	1	7	5
		2	91.97% (149/162)	12	5	8
		3	79.69% (106/133)	12	10	15
O	O	1	**98.63**% (145/147)	3	2	0
		2	**93.82**% (152/162)	3	2	8
		3	**94.73**% (126/133)	7	0	6

*AC: Auto Calibration, IPMO: IPM Optimization

TABLE III: Results of Algorithm Comparison

Method	Set	Accuracy↑	FP↓	FN↓	ID Switch↓
d	1	69.38% (102/147)	18	34	11
	2	16.04% (26/162)	7	133	3
	3	55.63% (74/133)	17	31	27
ψ	1	87.75% (129/147)	44	1	17
	2	76.54% (124/162)	89	1	37
	3	73.68% (98/133)	68	0	34
$\mathcal{E}$(Ours)	1	**98.63**% (145/147)	3	2	0
	2	**93.82**% (152/162)	3	2	8
	3	**94.73**% (126/133)	7	0	6

*d: Euclidean distance based matching, ψ : azimuth difference based matching, $\mathcal{E}$: Error Polygon based matching

$\mathbf{x}(\mathbf{p})$. The improvement in object distance estimate is the main attention of the comparison.

The results are shown in Fig. 9, illustrating the accumulated position of a static object across 40 radar frames. As depicted in Fig. 9, the camera's distribution in both latitude and longitude directions appeared dispersed when compared to the distribution of radar positions before IPM optimization. However, after the IPM optimization, the result became more similar to the actual distribution of radar positions.

D. Sensor Fusion Experiment

1) **Experiment Setups:** In this experiment, two methods were employed to verify the algorithm's accuracy. Initially, an Ablation Study was conducted to carefully assess the effects of each algorithmic stage. The comparison focused on the impacts of Auto Calibration (AC) and IPM Optimization (IPMO) on accuracy improvement. Second, the Hungarian algorithm with Euclidean distance (d) and the Hungarian algorithm with azimuth difference (ψ) were used as cost functions to compare the method's results for quantitative performance evaluation.

The Eq. 8 was used to evaluate the algorithm's accuracy. The results were calculated for the entire frame T.

$$\text{Accuracy} = \frac{\sum_{t=0}^{T}(\text{Success Case})}{\sum_{t=0}^{T}(\text{Ground Truth})} \tag{10}$$

Additionally, 3 key factors - FN, FP, ID switches - were considered in addition to this accuracy formula, leading to further analysis. FN represents false negatives, FP denotes false positives, and ID switch refers to the phenomenon occurring when fusion with different objects is incorrect.

2) **Experiment Results:**

- **Ablation Study**

Table II presents the results of the experiment on the effects of Auto Calibration and IPM optimization.

Initially, excluding two specific factors, the accuracy is extremely low, to the extent that the success rate is not even half. Furthermore, a slight reduction in accuracy was observed when IPM Optimization was applied. This occurred because many objects were fused incorrectly based on azimuth, leading to higher ID switch values. On the other hand, accuracy showed a tendency to improve after using Auto Calibration. The algorithm's design, which primarily creates Error Polygons based on Ray Casting, Ray Marching, and measures similarity using the Error Polygon overlap, has been attributed to this increase in performance. As a result, azimuth accuracy is crucial for overall accuracy.

As a result, AnytimeFusion, the suggested algorithm that combines the two factors, demonstrated the highest algorith-

mic accuracy. The proposed algorithm exhibited accuracies of 98.63%, 93.82%, and 94.73% per datasets. When averaging across all datasets, the overall accuracy reached 95.7%.

- **Comparison with Other Algorithms**

The results of the second experiment, which compared different approaches, are shown in Table III. First, uncertainty from IPM Optimization cannot be addressed by the Euclidean cost function d since it only measures association using camera and radar positions with high FN. However, because the azimuth-based approach ψ is unable to reflect distance information, it generates excessive associations for all objects in the same azimuth, resulting in a high FP ratio and accuracy loss.

On the other hand, throughout all three datasets, the proposed algorithm showed the highest accuracy results. Particularly, out of the three datasets, Set 1, which was the easiest to utilize, had the greatest accuracy. The ID switch case ratio was high for Set 2, which had high azimuth similarity. The set with the most small objects—Set 3—also exhibited the second-lowest accuracy.

Additionally, the AnytimeFusion algorithm can use Ray-based coordinate integration to obtain distance data for objects that the camera detects—even if the radar classifies them as land or noise. In Fig. 10, the radar excluded objects with ship A, C, and D from fusion because they were categorized as land. Nevertheless, they can be correctly integrated into the coordinate system of the radar through ray projection.

IV. CONCLUSIONS

In this paper, we propose AnytimeFusion, a robust and adaptable algorithm for the fusion of RGB cameras and radars in complex maritime environments where depth information is unavailable to the camera sensor. This algorithm compensates for azimuth distortions in the camera by first performing an Auto Calibration step. It then employs the Inverse Perspective Mapping technique to coordinate the integration of the camera and radar systems. It utilizes a Particle Swarm-based optimization method to handle ship motions. Lastly, it generates an Error Polygon and performs sensor fusion to compensate for errors that might have occurred in the two previous steps and evaluates the accuracy of the matching between the camera and radar. Evaluations of the algorithm's accuracy were also carried out using data obtained from real ships in challenging marine environments. Even in complex marine conditions, the results showed a high level of robustness, with an average accuracy exceeding 95.7%. To further expand on this, we plan to include and evaluate LiDAR sensors in the algorithm to create a more comprehensive fusion algorithm with improved azimuth and distance accuracy. Furthermore, to validate the system's suitability in real marine environments, we aim to demonstrate its ability to detect and avoid obstacles through real-time experiments.

REFERENCES

[1] Øystein Kaarstad Helgesen, K. Vasstein, E. F. Brekke, and A. Stahl, "Heterogeneous multi-sensor tracking for an autonomous surface vehicle in a littoral environment," *Ocean Engineering*, vol. 252, p. 111168, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801822005753

[2] S. M. Sali, N. L. Manisha, G. King, and K. Vidya Mol, "A review on object detection algorithms for ship detection," in *2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS)*, vol. 1, 2021, pp. 1–5.

[3] T. Liu, B. Pang, S. Ai, and X. Sun, "Study on visual detection algorithm of sea surface targets based on improved yolov3," *Sensors*, vol. 20, no. 24, 2020. [Online]. Available: https://www.mdpi.com/1424-8220/20/24/7263

[4] K. Kim, J. Kim, and J. Kim, "Robust data association for multi-object detection in maritime environments using camera and radar measurements," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 5865–5872, 2021.

[5] Z. Zhou, S. Yu, and K. Liu, "A real-time algorithm for visual detection of high-speed unmanned surface vehicle based on deep learning," in *2019 IEEE International Conference on Signal, Information and Data Processing (ICSIDP)*, 2019, pp. 1–5.

[6] X. Zhang, H. Wang, and W. Cheng, "Vessel detection and classification fusing radar and vision data," in *2017 Seventh International Conference on Information Science and Technology (ICIST)*, 2017, pp. 474–479.

[7] D. Hall and J. Llinas, "An introduction to multisensor data fusion," *Proceedings of the IEEE*, vol. 85, no. 1, pp. 6–23, 1997.

[8] G. Mirzaei, M. M. Jamali, P. V. Gorsevski, J. Frizado, and V. P. Bingman, "Data fusion of ir and marine radar data," in *2013 Asilomar Conference on Signals, Systems and Computers*, 2013, pp. 565–568.

[9] S. Pang, D. Morris, and H. Radha, "Transcar: Transformer-based camera-and-radar fusion for 3d object detection," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 10 902–10 909.

[10] H. Mallot, H. Bülthoff, J. Little, and S. Bohrer, "Inverse perspective mapping simplifies optical flow computation and obstacle detection," *Biological cybernetics*, vol. 64, pp. 177–85, 02 1991.

[11] S. Singhal, Y. Ao, D. Maas, B. Arsenali, and S. Maranò, "Marine vessel attitude estimation from coastline and horizon," 10 2023, pp. 6149–6154.

[12] A. A. G. S. S. T. S. Del Pizzo, S. Gaglione, "Vessel attitude estimation by camera sensors."

[13] H. Kim, J. Koo, D. Kim, B. Park, Y. Jo, H. Myung, and D. Lee, "Vision-based real-time obstacle segmentation algorithm for autonomous surface vehicle," *IEEE Access*, vol. 7, pp. 179 420–179 428, 2019.

[14] Y. Kim, J. Park, S. Kang, and H. Kim, "Introducing vada: Novel image segmentation model for maritime object segmentation using new dataset," 2024. [Online]. Available: https://arxiv.org/abs/2407.09005

[15] H. Kim, D. Kim, and S.-M. Lee, "Marine object segmentation and tracking by learning marine radar images for autonomous surface vehicles," *IEEE Sensors Journal*, vol. 23, no. 9, pp. 10 062–10 070, 2023.

[16] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, no. 1-2, pp. 83–97, 1955. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/nav.3800020109

[17] L. N. Chernov, *Circular and Linear Regression: Fitting Circles and Lines by Least Squares*, ser. Monographs on Statistics and Applied Probability. Chapman & Hall/CRC, 2010, vol. 117.

[18] S. D. Roth, "Ray casting for modeling solids," *Computer Graphics and Image Processing*, vol. 18, no. 2, pp. 109–144, 1982. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0146664X82901691

[19] M. R. Bonyadi and Z. Michalewicz, "Particle swarm optimization for single objective continuous space problems: A review," *Evolutionary Computation*, vol. 25, no. 1, pp. 1–54, 2017.

[20] J. Hart, "Sphere tracing: A geometric method for the antialiased ray tracing of implicit surfaces," *The Visual Computer*, vol. 12, 06 1995.

[21] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, "Generalized intersection over union: A metric and a loss for bounding box regression," 2019.