

Aerial Draft Surveyor (ADS)

Ship Drafting via YOLO11n and YOLO11n-Seg

John Matthew H. Escarro, Fharjan M. Taguinopon, Gyrielle Kysha M. Demegillo, Dan Kevin T. Amper,
Rosanna C. Ucat, Mark John S. Pag-Alaman

College of Engineering, University of Southeastern Philippines, Davao City, Philippines

Abstract—Draft surveying is an essential procedure in determining the displacement and loaded cargo weight of bulk carriers. Currently, the most acceptable method is through manual visual observation by trained draft surveyors. However, this process is subjective, error-prone, and unsafe under poor visibility or during rough sea conditions. This study presents an automated computer vision-powered UAV draft surveying system integrating TensorRT Optimized YOLO11n object detection and YOLO11n-seg image segmentation models deployed on an NVIDIA Jetson Orin Nano. The system performs real-time draft estimation by detecting draft marks, segmenting the waterline, and computing draft values using convergence and line-fitting algorithms. Comparative evaluation with licensed human surveyors on 40 paired readings yielded an MAE of 0.1068 m, RMSE of 0.2740 m, and an R^2 of 0.948, demonstrating human-comparable accuracy. Agreement analysis indicates high reliability (two-way random effects ICC(2,1) = 0.974) and a small mean bias (system – manual = +0.0628 m, 95% limits of agreement: –0.467 m to +0.592 m). Moreover, a paired t-test ($t = 1.469$, $df = 39$) found no statistically significant difference between methods ($p \approx 0.150$). The results validate that the proposed UAV-driven computer vision system can perform reliable, real-time draft surveying with accuracy comparable to human experts.

Keywords—Draft survey; UAV; machine learning; computer vision

I. INTRODUCTION

Ships play an important role in facilitating the global and international exchange of goods. In 2023 alone, the global trade volume of dry bulk shipping was around 5.508 billion metric tons [1].

To determine the weight of bulk cargo, ship draft surveys are conducted [2]. A draft is defined to be the vertical distance between the waterline and the bottom of the ship hull [3]. The method is based on Archimedes' principle and is conducted by observers who visually observe a ship's six draft marks while aiming to maintain an observation angle parallel to the water's surface to determine the ship's displacement tonnage, from which the weight can be calculated. Precision is important in measuring the draft as a singular centimeter error may correspond to an 80-ton error valued at around 40,000 USD or 500 USD per ton [1].

The ship draft represents the vertical distance from the waterline to the keel. Manual observation is traditionally prone to human error, but it is now evolving through automation [1, 2]. Draft marks are standardized numerical indicators positioned vertically along the hull, functioning as

calibrated measurement points, while the draft line or waterline forms the critical reference boundary where water meets the hull, from which all measurements originate [3, 4]. Draft points represent strategically positioned locations, typically at the forward and starboard, port, midship, and aft sections along the vessel hull, while draft scales refer to the vertical row of draft marks that together form a measurement system used to read the draft at a specific point on the vessel [5].

Manual observation over a time period by draft surveyors is the current accepted and widely practiced method, yet this is highly subjective and prone to human errors [1, 3, 5]. Additionally, manual observation is also greatly affected by other factors including surveyor experience, large waves, visibility, and tilted or rusted draft characters, among others [4, 5]. It is necessary to consider appropriate weather conditions during draft surveying to ensure that the vessel is stable for an accurate reading [6]. Moreover, having multiple people complete the draft surveying is ideal because it allows for crosschecking measurements for errors between surveys, ensuring that all final calculations are ideally accurate [7, 8].

In response to the need for precision in draft surveys and upholding the occupational safety of surveyors, recent studies have been conducted to aid in the automation of the draft reading process. A review from [2] highlighted recent advancements in automating ship draft surveying, including image recognition technologies, radar, optical fiber, IR, ship hull climbing robot, ship draft measuring ruler, sonar, and pressure sensor detection. Of these advancements, image recognition performed mostly on par with visual observation, while sensor-based solutions typically struggle under dynamic or harsh maritime conditions [2, 9, 10].

Several computer vision studies have presented similar solutions, utilizing a combination of YOLO object detection models for draft mark detection and U-Net or Mask R-CNN for waterline segmentation [11, 12, 13]. Chernyi and Ivanovskii [11] utilize YOLOv5 due to its low compute power requirement and speed in conjunction with U-Net to divide the image between "water" and "not water". Their method allowed them to achieve a mean Average Precision of 93.9% and processes video at 30 frames per second (FPS). Additionally, they state that 1800 sample values from a one minute 30 FPS video is sufficient. The primary caveat of their method is the requirement that a video of the measured draft point must be recorded before processed by their proposed system.

A study conducted in China [12] proposed a multitask learning framework (MTL-VDR) for automated vessel draft reading, integrating object detection and image segmentation within a single unified model. In their proposed system, the first stage utilizes YOLOv8 to detect and extract the region of the hull with draft marks. Then, their multi-task framework processes the extracted region, performing draft scale recognition and vessel-water segmentation before calculating the final draft. Their results show an error of only $\pm 0.074\text{m}$ while achieving a process time of 0.016s per video frame on a laptop with AMD Ryzen 7 5800H CPU accelerated by a NVIDIA GTX 3060 GPU. Though their study shows promise for draft reading, the approach is limited to single-frame image analysis and cannot ensure accurate draft estimation when the water surface fluctuates due to waves or motion. They address this limitation and recommend temporal stabilization or multi-frame averaging techniques to improve reliability in real-world, dynamic maritime environments.

Wang et al. [13] had a similar approach but utilized three distinct models for their system, namely Mask R-CNN for segmenting the targeted region with draft marks and the water regions, U-Net to refine the waterline boundary, and ResNet for recognition of draft characters. They achieved 93% accuracy in waterline segmentation, 98.5% accuracy in digit recognition, and an accuracy within three standard deviations of manual readings. They have achieved a processing speed of 0.36s per frame using a 32GB RAM PC with NVIDIA GTX 1080Ti GPU by analyzing recorded video footage. Despite the high accuracy, their system remains constrained by non-real-time performance and sensitivity to environmental disturbances. However, they do introduce a perspective distortion correction formula similar to [1] which greatly improves upon the quality of their draft estimate.

Based on the discussion of related literature above, the use of Convolutional Neural Networks (CNN) in the form of either U-Net, YOLO, or Mask R-CNN is relatively successful in detecting draft marks and segmenting water from an image. Despite the promising accuracy achieved by CNN-based systems, these methods remain confined to post-processed or static analysis. None, to date, have demonstrated an integrated real-time approach suitable in guaranteeing stable measurements for dynamic maritime environments where waves, reflections, and vessel motion introduce substantial visual fluctuations [12]. Further, an interesting problem that has yet to be brought up by previous studies is determining a way to know when or how to finalize measurement for real-time analysis. Hence a gap exists in achieving both real-time draft estimation and automated stability evaluation.

This study addresses these prior limitations by implementing real-time detection and segmentation through a deployable UAV system with TensorRT optimized YOLO11n and YOLO11n-seg on a Jetson Orin Nano for real-time analysis in the field. YOLO11n has improved small object detection capability and is more computationally efficient compared to its predecessors [14, 15, 16]. Although Mask R-CNN and U-Net excel in segmentation accuracy, the need for real-time analysis means that a balanced approach in speed and accuracy is required, making YOLO11n-seg suited for the task [14, 15, 16, 17]. Model optimization with TensorRT was

implemented as it is crucial in maximizing inference speed, where various studies have shown significant performance gains with TensorRT optimization [18, 19, 20]. The Jetson Orin Nano was chosen to eliminate dependence on cloud computing as it is powerful enough to handle both inference and mathematical tasks on its own which is crucial when operating out in the sea where a stable internet connection may not be present [21]. Where this study delineates from previous studies is the introduction of temporal subsampling and a cumulative averaging mechanism for continuous evaluation of sequential draft readings.

Temporal subsampling is a technique where every $1/N$ frame is taken for processing [22, 23, 24]. This technique is useful in conserving computational resources, which is essential in edge computing, provided that data skipped between frames is negligible. Recent data in the Philippines shows that the local wave period can be roughly estimated to be around 5-6 seconds [25, 26, 27]. Performing a simple calculation, we can infer that the dominant frequency is $1/5$ or 0.2 Hz, and consequently the Nyquist sampling rate is 0.4 Hz. Subsampling the input stream of 30 FPS at a rate of every third frame yields a sampling rate of 10 Hz, far higher than the minimum Nyquist requirement indicating that data resolution is sufficient. For this study, the minimum time for data collection of draft estimates will be 1 minute similar to [11], yielding at least 600 datapoints.

Each subsampled frame then contributes to a cumulative moving average (CMA). The CMA utilizes all data from the start of time to the present time point and has a characteristic of eventually converging into a stable value as more data points accumulate [28, 29, 30, 31, 32]. This stable value is the potential draft estimate. To validate stability, convergence check based on zero-slope fitting and normalized root mean square error (NRMSE) criteria automatically determines when a sufficiently stable reading has been achieved [33, 34, 35, 36, 37, 38, 39]. Normalization of the RMSE translates it into a meaningful percentage value of how much the data fits the model, where $\text{NRMSE} < 10\%$ is excellent, NRMSE between 10% and 20% is good, NRMSE between 20% and 30% is fair, and $\text{NRMSE} > 30\%$ is poor [35, 36, 37]. In the Philippine context, draft estimation is precise only up to the thousandth decimal, hence we set the minimum tolerance for near-zero slope to be at least 0.0001, and the NRMSE must be 0.1% to indicate true stability.

In brief, this work seeks to determine whether a field-deployable UAV-based computer vision system can achieve draft measurement accuracy comparable to human surveyors while maintaining operational reliability under real maritime conditions. It further aims to deliver real-time draft readings despite the limited computational resources of edge devices by employing TensorRT-optimized YOLO11n and YOLO11n-seg models, combined with lightweight algorithms for temporal subsampling, CMA, and draft reading stability checks based on line fitting and NRMSE.

The remainder of the study is organized as follows: Section II presents the system design, algorithm, and methodology. Section III discusses results and performance

evaluation. Finally, Section IV provides a conclusion iterating key findings and recommendations for future work.

II. METHODOLOGY

In designing an effective real-time draft surveying system, it needs to achieve real-time performance on edge computing hardware while also being a field-deployable measurement platform capable of achieving accuracy matching those of experienced surveyors even when faced by dynamic maritime conditions. To fulfill these requirements, the Aerial Draft Surveyor (ADS) system integrates a UAV-mounted camera for live video acquisition, an NVIDIA Jetson Orin Nano for on-site model inference, and a computer vision pipeline that detects draft marks, segments the waterline, and computes draft estimates in real time. The following subsections detail the system architecture, model training and development, draft computation algorithms, data gathering procedure, and statistical tests and validation for system evaluation.

A. System Architecture

The system architecture consists of four primary parts: the drone, the remote control (RC), the Jetson Orin Nano, and the User Interface (see Fig. 1).

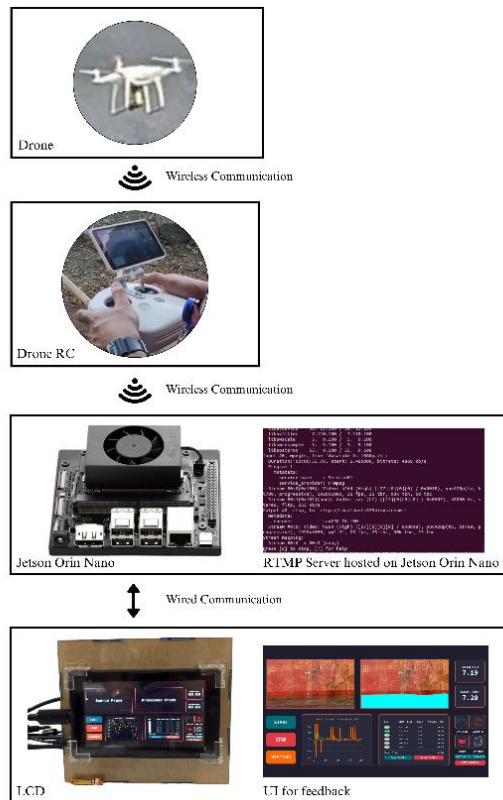


Fig. 1. System architecture.

The manually operated drone, a DJI Phantom 4, is capable of long-range flight in windy seaside conditions and is equipped with an onboard camera with wireless video transmission at 1080p, 30 FPS. It is positioned directly in front of the targeted draft marks at a distance of 3 to 5 meters while maintaining perpendicular line of sight. The drone transmits its captured video to the operator's remote controller. The

remote controller then streams this live video feed it receives from the drone via an RTMP (Real-Time Messaging Protocol) stream over a LAN (Local Area Network) generated by a simple mobile hotspot, which is received by an RTMP server running on the NVIDIA Jetson Orin Nano. This video stream is then fed to a program running on the Jetson Orin Nano responsible for running inference on the video feed, performing both object detection and image segmentation to detect ship draft marks and isolating the waterline. It then calculates the draft value and outputs the result to a user interface for review and logging. For hardware, a portable power supply was used to power the device, an LCD was attached to display the program's user interface, and peripherals were utilized for ease of use (see Fig. 2).

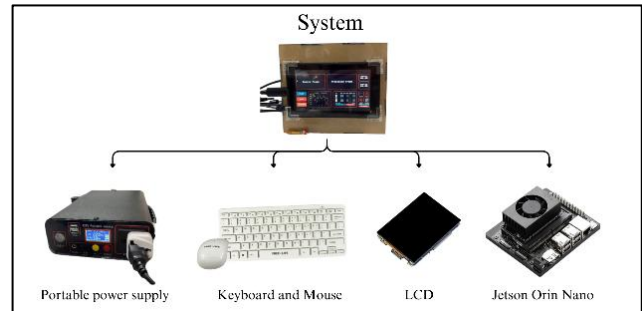


Fig. 2. Basic system hardware components.

B. System Development

Frames from the video stream were subsampled by the program then processed by two deep learning models: YOLO11n and YOLO11n-seg. A lightweight nano model was chosen to facilitate real-time inference on the Jetson Orin Nano. These models were trained using images in labeled datasets from Roboflow using a 70:20:10 train-validation-testing split (see Fig. 3 and Fig. 4).

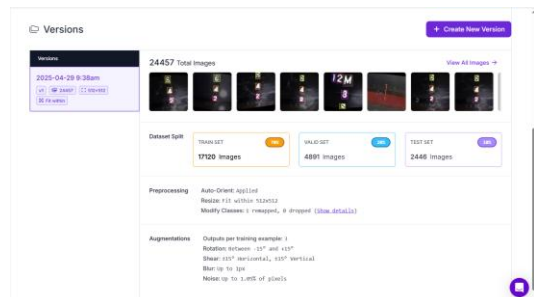


Fig. 3. Detection dataset.

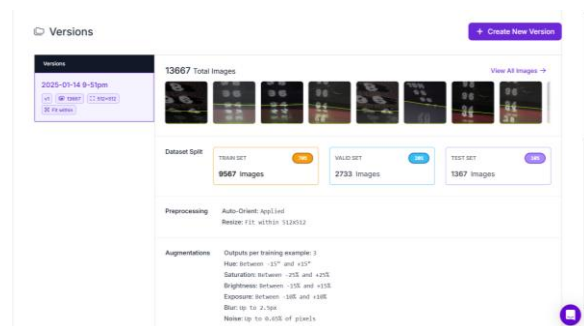


Fig. 4. Segmentation dataset.

Augmentations were applied to enhance model performance in varying conditions. Below is a summary of the model training parameters applied (Table I).

TABLE I. SUMMARY OF MODEL TRAINING PARAMETERS

Model	Images	Augmentations Applied	Epochs	Size	Classes
YOLO11n	24457	Rotation ($\pm 15^\circ$) Shear ($\pm 15^\circ$ H, $\pm 15^\circ$ V) Blur (up to 1px) Noise (up to 1.05% px)	250	512x512	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, M
YOLO11n-seg	13667	Hue ($\pm 15^\circ$) Saturation ($\pm 25\%$) Brightness ($\pm 15\%$) Exposure ($\pm 10\%$) Blur (up to 2.5px) Noise (up to 0.65% px)	50 ^a	512x512	Water, background

^a Model set to train for 250 epochs but stopped at 50 due to no improvement.

Model training utilized labeled datasets obtained from Roboflow Universe, which contained publicly available ship hull images with annotated draft marks and waterlines. For training draft mark detection, dataset from user shuichil which featured pictures of draft marks from various ships in different perspectives was used. To train the segmentation model, dataset from user shuixian1 which featured annotated segmented waterline was used.

These datasets were forked and augmentations were added to enhance model robustness in real-world conditions, and consequently, increase generalization. Augmentations for rotation ($\pm 15^\circ$) and shear ($\pm 15^\circ$ H, $\pm 15^\circ$ V) were applied to enhance model resistance against varying camera angles. Image noise was added to allow the model to correctly identify regions despite water splashes. Lastly, blur was augmented into both datasets to accommodate for the drone's motion which may induce blur. Saturation, brightness, and exposure were specifically added to the segmentation model to help it correctly segment the waterline against small variations in water color and brightness distortion from lighting.

Both models were trained on Google Colab's A100 High-RAM GPU using the PyTorch implementation of YOLO11 under the default training hyperparameters provided by the Ultralytics framework for 250 epochs. The YOLO11n model was trained for numerical detection of the ship's draft marks while the YOLO11n-seg model was trained for waterline segmentation. Models were then converted into TensorRT engine files locally on the Jetson Orin Nano to optimize performance.

C. Draft Calculation Algorithm

Algorithm 1 is a pseudocode representing the live-stream draft measurement pipeline, showing a simplified and high-level overview of the functions for instantaneous draft analysis as well as determining the final draft. The 30 FPS stream is sent as an input to the primary loop function MeasureDraft (line 10), where it subsamples the video stream by sending every third detected frame as an input to the InstantaneousDraft function (line 1). The subsampled frame then passes through both object detection and segmentation (line 2), from which the bottom-center points of the bounding boxes that form the draft line are obtained (lines 3-4) to fit a line and predict the position of the next point (lines 5-6). The draft for this frame is then determined (line 7). Fig. 5 shows a visual representation of the InstantaneousDraft function.

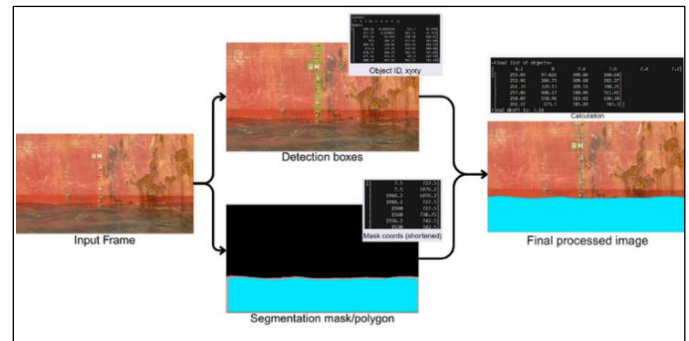


Fig. 5. Per frame instantaneous draft reading process.

In calculating the final draft, a cumulative average is applied continuously (line 18) for each draft. When the number of samples have reached the threshold, a check for convergence is conducted, outputting the mean as the final draft if the slope and NRMSE fall below specified thresholds (lines 19-20). Fig. 6 provides an overview of the algorithm's processes through a visual flowchart.

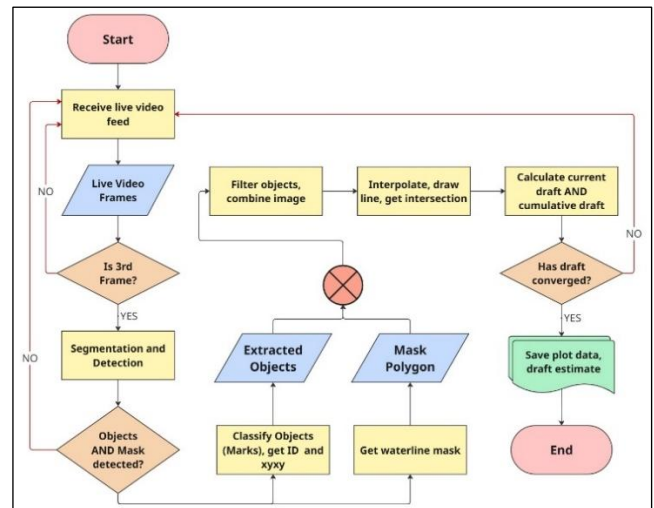


Fig. 6. Algorithm flowchart.

Algorithm 1: Draft Calculation

INPUT: live video frames S
OUTPUT: streaming currentDraft, mean when converged

```

1  FUNCTION InstantaneousDraft(frame):
:
2      draft_marks, water_mask ← RunInference(frame)
:
3      bottom_centers ← ExtractBboxBotCenters(draft_marks)
:
4      draft_line ← FitLine (bottom_centers)
:
5      predicted_point ← InterpolateNext(draft_line)
:
6      intersect ← GetIntersection(draft_line, water_mask)
:
7      draft ← CalculateDraft(draft_line, predicted_point,
:      intersect)
:
8      RETURN draft
:
9  END FUNCTION

0: 1  FUNCTION MeasureDraft(S)
1: 1      frame_cnt ← 0
2: 1      count ← 0, mean ← 0, window ← []
3: 1      WHILE streaming IS available:
4: 1          frame_cnt ← frame_cnt + 1
5: 1          IF frame_cnt MOD 3 == 0 THEN CONTINUE
6: 1          current_draft ← InstantaneousDraft(S)
7: 1          count ← count + 1
8: 1          mean ← CumulativeAvg(mean, count,
9: 1          current_draft)
10: 1          APPEND mean to window
11: 2          IF count > 600:
12: 2              POP mean from window
13: 2              IF Converged(window) THEN RETURN
14: 2              mean
15: 2              END IF
16: 2          END WHILE
17: 2      END FUNCTION

```

The following are the key routines in this algorithm:

- RunInference(frame): returns draft mark bounding boxes and waterline mask.

- InterpolateNext(draft_line): predicts the point position of the next draft mark located below the waterline then connects all points.
- GetIntersection(draft_line, water_mask): finds the coordinates where the draft line intersects with segmented water mask.
- CalculateDraft(draft_line, predicted_point, intersect): calculates the draft in meters by translating pixel distances (1).
- CumulativeAvg(mean, count, current_draft): uses recursive averaging to calculate a running mean of all draft measurements (4).
- Converged(window): checks all cumulative means to determine if values have stabilized (6, 7).

To calculate the draft value from the images, the Eq. (1) below from [1] is used:

$$v_0 = v_1 - s \left(\frac{d_0}{r \cdot d_1} \right) \quad (1)$$

where, v_0 is the estimated draft at p_0 (m); v_1 is the known draft at p_1 (m); d_0 and d_1 are pixel distances (px); r is the perspective-correction ratio (unitless); and s is the true spacing between adjacent draft marks (m) which is 0.2m. Values for d_0 and d_1 are obtained through simply calculating the distance between points using Eq. (2):

$$d_0 = \|p_0 - p_1\| ; d_1 = \|p_1 - p_2\| ; d_2 = \|p_2 - p_3\| \quad (2)$$

- To account for instances, where the drone is not perfectly aligned with the ship's hull, the distances obtained from the two draft marks d_2 and d_1 above the draft mark nearest the waterline are used as a reference (see Fig. 7). From here the perspective-correction ratio is obtained through Eq. (3).

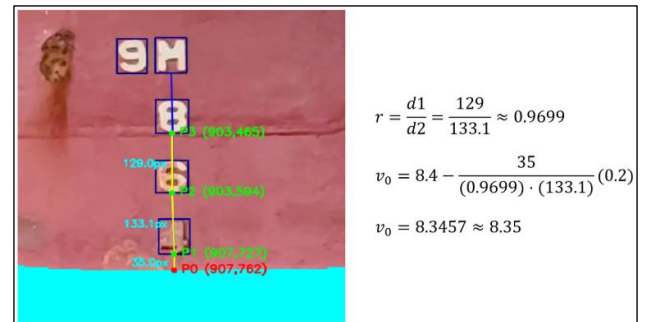


Fig. 7. System calculation to estimate draft mark value.

$$r = \frac{d_1}{d_2} \quad (3)$$

Welford's algorithm, Eq. (4), is used in calculating for the CMA of draft readings as it is computationally inexpensive, expressing the running average in a recursive form requiring only the previous average and the new value to obtain the new average [32].

$$\bar{x}_n = \left(\frac{n-1}{n} \right) \bar{x}_{n-1} + \left(\frac{1}{n} \right) x_n \quad (4)$$

In Eq. (4), n corresponds to the current point in time. Then, $\bar{x}_n$ is the current cumulative average, $\bar{x}_{n-1}$ is the previous cumulative average, and x_n is the current value. In this case, x_n represents the instantaneous draft reading at the current point in time, $\bar{x}_{n-1}$ is the old cumulative draft estimate before the current reading, and $\bar{x}_n$ is the new cumulative draft estimate.

Linear regression, Eq. (5), is performed on all datapoints to estimate the slope and generate a fitted line. The goal is to determine whether the cumulative draft estimate is stable and may be finalized.

$$\hat{y} = \beta_1 x + \beta_0 \quad (5)$$

Here, $\hat{y}$ is the predicted value from the fitted line, β_1 is the slope, x is the time step or index, and β_0 is the y-intercept. The slope is then identified by calculating β_1 as shown in Eq. (6), where N is the total number of points in the window, x_i is the frame index, $\bar{x}$ is the average frame index, y_i is the CMA at x_i , and $\bar{y}$ is the average CMA within the window N .

$$\beta_1 = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^N (x_i - \bar{x})^2} \quad (6)$$

Though a near-zero slope hints at stability, NRMSE is also obtained through Eq. (7) to determine if the actual values around this fitted line are also stable, as simply identifying the slope does not reveal information on how much the data points are still varying.

$$NRMSE = \frac{\sqrt{(1/N) \sum_{i=1}^N (y_i - \hat{y}_i)^2}}{\bar{y}} \quad (7)$$

Therefore, convergence of the draft estimate obtained from CMA is determined to be stable by both checking if slope and NRMSE fall under the required criterion, specified to be $\beta_1 \leq 0.0001$ and $NRMSE \leq 0.1\%$.

D. Data Gathering

Data was gathered by flying the drone towards a bulk cargo vessel. The operator will control the drone to record video from each of the ship's six draft locations, waiting for the system to output its final draft reading before moving to the next draft location. When all of the six draft locations are finished, the drone is retrieved. Whenever the ship is loaded with new cargo, the procedure is repeated. This is to ensure that the ship's draft would vary every time a measurement is taken. An overview of the process is shown in Fig. 8.

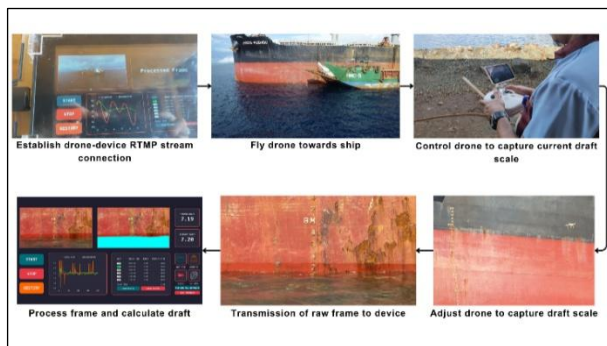


Fig. 8. Data gathering procedure.

E. Testing and Evaluation

To evaluate the system's performance, a team of licensed surveyors were also sent to the bulk cargo vessel to perform manual reading of each of the six draft marks. The system's results are then compared against the surveyor's readings to quantify the system's performance.

F. Statistical Analysis

To rigorously evaluate the agreement and accuracy of the automated draft survey system against traditional human measurements, four complementary statistical methods were applied. Firstly the standard error metrics, which include mean squared error (MSE), root mean squared error (RMSE), mean absolute error (MAE), and coefficient of determination (R2), are obtained. These determine how the algorithm's predictions track the ground truth observation of human surveyors alongside how much of the variability in ground truth is explained by the model. A paired t-test is also utilized to identify the existence of systematic biases and determine whether the mean difference between the two depart significantly from zero.

Next, Bland-Altman analysis is used to visualize and quantify agreement across the full measurement range. This reveals whether the algorithm's errors remain acceptably small and consistent at both low and high drafts, and identified any outliers [40]. In addition, the two-way random-effects, absolute-agreement intraclass correlation coefficient (ICC(2,1)) was also calculated. This partitions total variance into between-subject and error components, revealing whether the two are effectively interchangeable or not [41].

III. RESULTS AND DISCUSSION

This section features an evaluation of the live-stream draft measurement system, beginning with the accuracy and statistical reliability of the final draft estimates before proceeding to the performance of the underlying deep-learning models. First, the automated draft readings will be compared against manual surveyor measurements using the methods outlined under the statistical analysis section. Next, object-detection and segmentation metrics achieved by the YOLO11n and YOLO11n-seg networks, including precision, recall, mAP, IoU, and real-time inference and computation speed will be presented. Finally, these results are interpreted in the context of operational requirements for at-sea draft monitoring, discussing both the system's real-time capability and its potential to augment traditional surveying methods.

A. System and Manual Draft Reading

To evaluate the accuracy and reliability of the Aerial Draft Surveyor (ADS) system in comparison with traditional manual measurements, a total of forty (40) readings were obtained (see Table II). Each reading was simultaneously recorded by a licensed surveyor through manual observation and by the ADS through automated image processing. Both measurements were performed under similar environmental conditions to ensure fairness of comparison.

The quantitative comparison between manual and system-obtained readings is summarized in Table III. Statistical metrics including Mean Square Error (MSE), Root Mean

Square Error (RMSE), Mean Absolute Error (MAE), and Coefficient of Determination (R^2) were computed to evaluate accuracy, while a paired t-test (at $\alpha = 0.05$, $df = 39$) assessed whether systematic bias existed between the two measurement methods.

TABLE II. SYSTEM AND MANUAL DRAFT READING

Index	Measured Draft		Index	Measured Draft	
	System(m)	Manual(m)		System(m)	Manual(m)
1	4.4	4.6	21	8.37	8.3
2	6.61	6.65	22	8.38	8.31
3	6.27	6.38	23	5.6	5.66
4	8.17	8.05	24	7.17	7.18
5	8.1	8.05	25	6.55	6.48
6	5.29	5.29	26	8.37	8.3
7	6.07	5.05 ^a	27	8.36	8.31
8	7.17	7.13	28	5.58	5.66
9	6.3	6.26	29	7.16	7.18
10	8.37	8.37	30	8.38	8.3
11	8.32	8.37	31	8.37	8.31
12	5.3	5.29	32	5.56	5.66
13	6.39	5.05 ^b	33	7.19	7.18
14	7.14	7.13	34	8.37	8.3
15	6.25	6.26	35	8.41	8.31
16	8.38	8.37	36	5.63	5.62
17	8.36	8.37	37	7.17	7.17
18	5.57	5.66	38	6.37	6.47
19	7.2	7.18	39	8.32	8.31
20	6.54	6.48	40	8.3	8.3

^{a, b} Rust had turned mark into a color similar to the ship's hull resulting in faulty reading.

TABLE III. SUMMARY OF EVALUATION METRICS

MSE	RMSE	MAE	R^2	t-value
0.0751	0.2740	0.1068	0.9506	1.47

The linear relationship between the system's readings and those obtained manually is illustrated in Fig. 9. Each point corresponds to a single measurement pair, with the dashed line representing perfect 1:1 agreement and the solid line showing the regression fit. As shown, the system's predictions cluster tightly along the 1:1 line, indicating excellent correspondence with manual observations.

The obtained R^2 value of 0.9506 indicates a strong linear agreement between automated and manual readings, meaning that 95% of the variance in human-observed measurements can be explained by the system's estimates. The t-test result ($t = 1.47 < 2.02$, $p > 0.05$) further supports that no statistically significant difference exists between system and manual observations, thereby validating the reliability for operational

use. However, the average deviation is $\pm 0.1068\text{m}$, which is slightly larger than those from previous studies [12, 13].

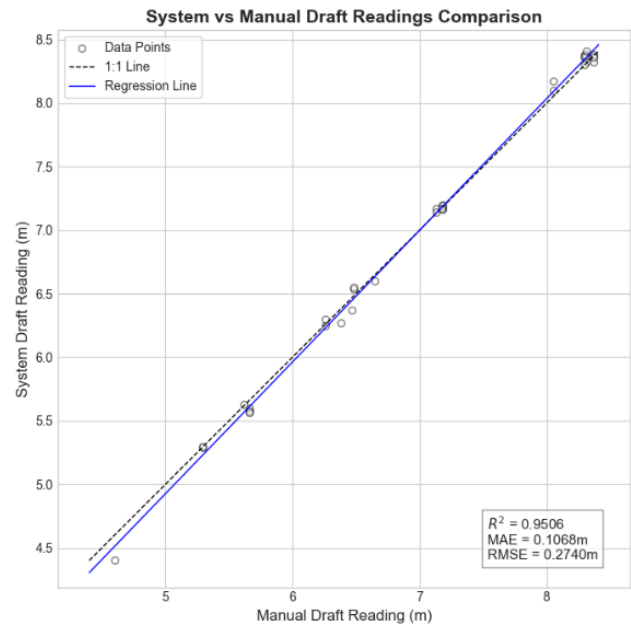


Fig. 9. Scatter plot comparing manual and system readings.

The major discrepancy observed is during cases where corrosion from rust led the system to misclassify the target draft mark resulting in a faulty reading. Nonetheless, the overall error remained small, and the algorithm proved capable of converging towards a reasonable stable draft estimate through the cumulative averaging and convergence criteria described in Section IIC.

To examine the potential effect of such outliers, an additional analysis was conducted by excluding measurements affected by severe hull corrosion. Under these idealized conditions, the mean absolute error (MAE) improved to $\pm 0.0505\text{ m}$, which falls within the same accuracy range reported by [12] and [13]. This indicates that under optimal imaging conditions, the system's performance approaches that of state-of-the-art laboratory-based methods.

B. Bland-Altman Analysis

TABLE IV. BLAND-ALTMAN SUMMARY

Parameter	Value
Mean Difference (Bias)	+0.063
Standard Deviation	0.269
Lower Limit of Agreement (LoA)	-0.467
Upper Limit of Agreement (LoA)	+0.592
% of Points Within LoA	95%

The mean difference (bias) between the manual and machine readings was +0.063, suggesting a slight tendency for the machine to read marginally higher than the manual method. The 95% limits of agreement ranged from -0.592 to +0.467. Approximately 95% of the paired measurements fell

within these limits, with no clear pattern of increasing disagreement for higher or lower draft marks (see Fig. 10 and Table IV).

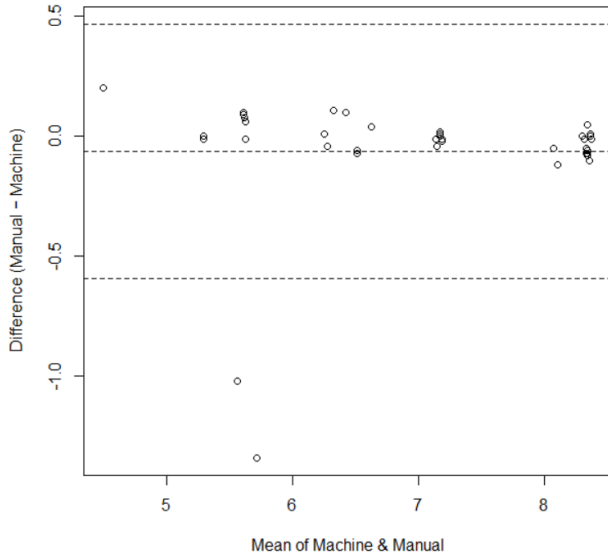


Fig. 10. Bland-Altman Plot of machine vs. Human surveyor readings.

The two major outliers were attributed to the presence of rust which heavily impaired the detection model's capabilities in discerning the correct draft mark.

C. Intraclass Correlation Coefficient [ICC (2,1)]

The agreement between the automated (machine) and manual draft readings was assessed using a two-way random-effects absolute-agreement single-measure ICC (ICC(2,1)). As shown in Table V, the ICC was 0.974 (95 % CI: 0.951–0.986; $F(39, 38.9) = 78.0$; $p < 0.001$), indicating reliability and consistency between the two methods.

TABLE V. INTRACLASS CORRELATION COEFFICIENT RESULTS

Statistic	Value
Number of subjects	40
Number of raters/methods	2
ICC value	0.974
95% Confidence Interval (CI)	0.951-0.986
F Statistic (df1, df2)	78.0 (39, 38.9)
p-value	< 0.001

According to the benchmarks proposed by [41] ICC values above 0.90 indicate excellent reliability. This suggests that nearly all the observed variance in draft measurements is attributable to genuine differences in the ship's draft rather than measurement error between the two methods. In practical terms, the system's readings are interchangeable with human observations under typical survey conditions.

The narrow confidence interval (0.951–0.986) reinforces the precision and stability of the reliability estimate, implying that even with sampling uncertainty, the true level of

agreement remains very high. Furthermore, the high F-ratio ($F = 78.0$, $p < 0.001$) confirms that between-measurement variability far exceeds residual error variance, suggesting the proposed system captures the same underlying measurement signal as human surveyors.

D. Model Performance

TABLE VI. MODEL TRAINING VALIDATION RESULTS

Model	Metric				
	Precision	Recall	mAP	IoU	Inference time
YOLO11n	99.82%	99.80%	99.49%	—	30ms
YOLO11n-seg	98.12%	97.74%	99.06%	97.28%	20ms

As seen in Table VI above, both the detection and segmentation networks achieved near real-time speeds with exceptionally high accuracy. The YOLO11n detector reached 99.82% precision, 99.81% recall and a mAP@0.5 of 99.49%, processing each frame in about ~30 ms. Its companion model, YOLO11n-seg, scored a mAP@0.5 of 99.06%, an IoU of 97.28%, and maintained over 97% recall (98.12 % precision) on full-scene masks, with an inference time of around ~20 ms per frame. Both models exhibit reliable isolation of their targets in varying hull conditions, provided excessive rusting does not distort the number for the detection model.

Speed and timing performance were evaluated on the Jetson Orin Nano by processing a 10-second video clip containing visible draft marks. The average inference time for each model and computation times were recorded (see Fig. 11). On average the complete pipeline yielded a latency of approximately 10ms per frame, corresponding to an effective throughput of 10 FPS.

```
0: 288x512 1 water, 30.9ms
Speed: 2.0ms preprocess, 30.9ms inference, 3.0ms postprocess per image at shape (1, 3, 288, 512)
Processed frame 110 || Draft is: nan || Running draft avg is: 8.607647058823527

0: 288x512 1 0, 1 1, 1 2, 2 4s, 2 6s, 2 8s, 1 9, 2 Ms, 23.9ms
Speed: 2.0ms preprocess, 23.9ms inference, 2.0ms postprocess per image at shape (1, 3, 288, 512)

0: 288x512 1 water, 29.9ms
Speed: 2.0ms preprocess, 29.9ms inference, 2.0ms postprocess per image at shape (1, 3, 288, 512)
Processed frame 111 || Draft is: nan || Running draft avg is: 8.607647058823527

0: 288x512 1 0, 1 1, 1 2, 2 4s, 2 6s, 2 8s, 1 9, 2 Ms, 24.9ms
Speed: 2.0ms preprocess, 24.9ms inference, 2.0ms postprocess per image at shape (1, 3, 288, 512)

0: 288x512 1 water, 31.0ms
Speed: 2.0ms preprocess, 31.0ms inference, 3.0ms postprocess per image at shape (1, 3, 288, 512)
Processed frame 112 || Draft is: nan || Running draft avg is: 8.607647058823527
Final Draft Reading: 8.607647058823527
Done!
Avg detection time: 0.039s
Avg segmentation time: 0.032s
Avg processing time: 0.039s
Avg program full loop time: 0.101s
```

Fig. 11. Average system time for detection, segmentation, and calculation.

Compared to the systems proposed by [11], [12], and [13], the process times of the ADS utilizing an edge computing device lies in the middle, as seen in Table VII.

TABLE VII. COMPARISON OF COMPUTE TIMES BETWEEN STUDIES

Study	Device	Total time
Chemyi & Ivanovskii [11]	N/A	33 ms ^a
Zhang et al. [12]	NVIDIA GTX 3060	16 ms
Wang et al. [13]	NVIDIA GTX 1080Ti	360 ms
ADS	NVIDIA Jetson Orin Nano	101 ms

^a Value inferred from stated performance at 30 FPS, but no compute time was stated in the study.

Due to hardware constraints, the ADS system does not show the fastest processing times. However, previous studies had only focused on processing recorded videos in calm and stable conditions, whereas the ADS system was able to demonstrate modest performance in real-time draft reading in a real-world environment.

E. Discussion

The error-metric analysis revealed excellent point-wise performance between the automated and manual measurements (MSE = 0.0751, RMSE = 0.2740 m, MAE = 0.1068 m, $R^2 = 0.9506$). In practical terms, the algorithm's draft estimates deviate from human readings by about 0.11 m on average, while capturing more than 95 % of the variation observed by human surveyors. The MAE being larger than those of [12] and [13] are attributed to the two faulty readings caused by hull corrosion, obscuring the target draft mark. The effect of these outliers were analyzed, and when excluded to simulate idealized conditions, dropped the average deviation or MAE to ± 0.0505 , now within the accuracy ranges from [12] and [13]. The paired t-test ($t = 1.47$, $p > 0.05$) confirmed that the difference between the two methods is statistically insignificant, indicating the absence of systematic bias and validating the reliability of the automated approach for operational use.

Bland-Altman analysis provided further evidence of strong agreement, with a small positive bias of +0.063 m and 95 % limits of agreement between -0.467 m and +0.592 m. Ninety-five per cent of all paired measurements lay within these limits, demonstrating that deviations are both limited and evenly distributed across the measurement range. Again, the few outliers were traced to localized corrosion which temporarily disrupted the detection model.

The intraclass correlation coefficient ICC(2,1) of 0.974 (95 % CI: 0.951–0.986; $F(39, 38.9) = 78.0$; $p < 0.001$) confirms excellent absolute agreement between manual and automated readings. According to [41], ICC values > 0.90 reflect outstanding reliability. This finding implies that nearly all variance arises from true physical changes in draft rather than measurement noise, meaning that the automated readings are interchangeable with those of human surveyors. The narrow confidence interval further confirms the stability of this reliability estimate. Taken together, the ICC, Bland-Altman, and regression analyses establish a consistent pattern of agreement, affirming the robustness of the system.

Model-level evaluation corroborates the system-level findings. The YOLO11n detector and YOLO11n-seg segmentation network achieved precision and recall exceeding 97 %, with mean average precision (mAP@0.5) values above 99 %. Despite running on an edge device, inference times averaged 30 ms and 20 ms per frame, respectively, yielding a full-pipeline latency of ~ 101 ms (~ 10 FPS). While this is slower than the 16 ms reported by [12] on an RTX 3060 GPU, it surpasses the 360 ms runtime of [13] and remains within real-time operational limits for live draft monitoring. The performance difference primarily reflects the hardware class, and the ADS achieves comparable accuracy in the field while maintaining full onboard processing capability without cloud dependence.

IV. CONCLUSION

This study presented the design, development, implementation, and validation of the Aerial Draft Surveyor (ADS), a UAV-based computer vision system that automates ship draft measurement through YOLO11n object detection and YOLO11n-seg waterline segmentation, optimized for real-time edge inference on an NVIDIA Jetson Orin Nano through TensorRT. The system addressed key limitations present in previous studies, particularly by implementing and validating a real-time approach to draft-reading in the field. It introduces temporal subsampling to reduce computational load in real-time processing, CMA to ascertain a draft estimate, and a convergence check to determine when to finalize the draft estimate through stability checks with zero-slope fitting and NRMSE.

Results demonstrated excellent agreement with manual surveyor readings, achieving an MAE of 0.1068 m, RMSE of 0.2740 m, and $R^2 = 0.9506$, with no statistically significant bias ($t = 1.47$, $p > 0.05$). Bland-Altman analysis showed 95 % of paired readings within ± 0.6 m limits of agreement, while the ICC(2,1) = 0.974 (95 % CI: 0.951–0.986) confirmed reliability and consistency with human observations. Moreover, TensorRT optimization allowed for efficient inference on low-power edge hardware. Alongside lightweight algorithms, it allowed for a throughput of ~ 10 FPS representing a significant step toward practical, autonomous draft surveying.

Yet, similar to its predecessors, the system remains sensitive to corrosion and occluded draft marks, which degrade detection accuracy or completely mislead the system to an erroneous reading. Additionally, there is still much room for improvement to further decrease latency from processing to prevent frame skipping and obtain as much information as possible from the live video feed. Another issue is the dependence on human readings as ground truth as there is no means to determine the correctness of manual observation. Lastly, addressing adverse weather conditions in future studies may significantly improve the dependability of similar systems.

This study contributes to the field by: 1) introducing a real-time, UAV-integrated measurement framework; 2) combining temporal subsampling with convergence-based stability evaluation for automated draft finalization; 3) demonstrating TensorRT-accelerated YOLO11n and YOLO11n-seg performance on embedded systems; and 4) validating the system through comparative field trials with licensed surveyors. Collectively, these contributions bridge the gap between laboratory-based automation studies and operational, deployable maritime systems, addressing both research and practical needs.

In summary, this work confirms the feasibility of real-time, edge-deployed computer vision for ship draft surveying. The system shows promise as being a supplementary tool for real-time aid to support and validate manual draft surveying.

ACKNOWLEDGMENT

The authors wish to express their appreciation to Hinatuan Mining Corporation (HMC) for their cooperation and for

granting access to their facilities and equipment during the conduct of system testing.

REFERENCES

- [1] B. Zhang, J. Li, H. Tang, and X. Liu, "Smart Ship Draft Reading by Dual-Flow Deep Learning Architecture and Multispectral Information," *Sensors*, vol. 24, no. 17, p. 5580, Aug. 2024, doi: 10.3390/s24175580.
- [2] Y. Wei, "Research Review of Ship Draft Observation Methods," *American Journal of Traffic and Transportation Engineering*, Mar. 2023, doi: 10.11648/j.ajtte.20230802.11.
- [3] B. Wang, Z. Liu, and H. Wang, "Computer vision with deep learning for ship draft reading," *Optical Engineering*, vol. 60, no. 02, Feb. 2021, doi: 10.1117/1.OE.60.2.024105.
- [4] Z. Wang, P. Shi, and C. Wu, "A Ship Draft Line Detection Method Based on Image Processing and Deep Learning," *J Phys Conf Ser*, vol. 1575, no. 1, p. 012230, Jun. 2020, doi: 10.1088/1742-6596/1575/1/012230.
- [5] W. Li *et al.*, "Research and Application of U²-NetP Network Incorporating Coordinate Attention for Ship Draft Reading in Complex Situations," *J Signal Process Syst*, vol. 95, no. 2–3, pp. 177–195, Mar. 2023, doi: 10.1007/s11265-022-01816-w.
- [6] M. Forsberg, "Conducting a Draft Survey on a RoPax Vessel – Finding the Ship's Lightweight," Bachelor's Thesis, Satakunta University of Applied Sciences, 2023. [Online]. Available: https://www.theseus.fi/bitstream/handle/10024/853298/forsberg_mira%20.pdf?sequence=2
- [7] X. Zhang *et al.*, "Self-Powered distributed water level sensors based on Liquid–Solid triboelectric nanogenerators for ship draft detecting," *Advanced Functional Materials*, vol. 29, no. 41, Mar. 2019, doi: 10.1002/adfm.201900327.
- [8] R. Ivčec, I. Jurdana, and R. Mohović, "Determining weight of cargo onboard ship by means of optical fibre technology draft reading," *PROMET - Traffic&Transportation*, vol. 23, no. 6, pp. 421–429, Feb. 2012, doi: 10.7307/ptt.v23i6.177.
- [9] Y. Wei, H. Du, Q. Hu, and H. Wang, "Optimizing Ship Draft Observation with Wave Energy Attenuation and PaddlePaddle-OCR in an Anti-Fluctuation Device," *Journal of Marine Science and Engineering*, vol. 12, no. 10, p. 1865, Oct. 2024, doi: 10.3390/jmse12101865.
- [10] S. Dhar and H. Khawaja, "Real-Time ship draft measurement and optimal estimation using Kalman Filter," *The International Journal of Multiphysics*, vol. 17, no. 4, Dec. 2023, doi: 10.21152/1750-9548.17.4.407.
- [11] "AUTOMATED SYSTEM FOR SHIP DRAUGHT MEASUREMENT WITH COMPONENT OF INTELLIGENT SYSTEMS," *Информационные Технологии И Вычислительные Системы*, Jun. 2022, doi: 10.14357/20718632220207.
- [12] J. Qu *et al.*, "Multi-Task Learning-Enabled automatic vessel draft reading for intelligent maritime surveillance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 5, pp. 4552–4564, Nov. 2023, doi: 10.1109/tits.2023.3327824.
- [13] B. Wang, Z. Liu, and H. Wang, "Computer vision with deep learning for ship draft reading," *Optical Engineering*, vol. 60, no. 02, Feb. 2021, doi: 10.1117/1.OE.60.2.024105.
- [14] N. Jegham, C. Y. Koh, M. Abdelatti, and A. Hendawi, "Evaluating the evolution of YOLO (You Only Look Once) models: A comprehensive benchmark study of YOLO11 and its predecessors," *arXiv (Cornell University)*, Oct. 2024, doi: 10.48550/arxiv.2411.00201.
- [15] J. Huang, K. Wang, Y. Hou, and J. Wang, "LW-YOLO11: A lightweight Arbitrary-Oriented ship detection method based on improved YOLO11," *Sensors*, vol. 25, no. 1, p. 65, Dec. 2024, doi: 10.3390/s25010065.
- [16] X. Qiu, Y. Chen, W. Cai, M. Niu, and J. Li, "LD-YOLOV10: A lightweight target detection algorithm for drone scenarios based on YOLOV10," *Electronics*, vol. 13, no. 16, p. 3269, Aug. 2024, doi: 10.3390/electronics13163269.
- [17] L. He, Y. Zhou, L. Liu, and J. Ma, "Research and application of YOLO11-Based object segmentation in intelligent recognition at construction sites," *Buildings*, vol. 14, no. 12, p. 3777, Nov. 2024, doi: 10.3390/buildings14123777.
- [18] I. Shamta and F. Demir, "Optimizing visual instruction detection in autonomous mobile robots using YOLOv8 and TensorRt acceleration," *Black Sea Journal of Engineering and Science*, vol. 8, no. 2, pp. 418–427, Mar. 2025, doi: 10.34248/bsengineering.1594542.
- [19] M. Vandersteegen, K. Van Beeck, and T. Goedeme, "Super accurate low latency object detection on a surveillance UAV," 2019 16th International Conference on Machine Vision Applications (MVA), May 2019, doi: 10.23919/mva.2019.8758060.
- [20] L. Steinmann, L. Sommer, A. Schumann, and J. Beyerer, "Fast and Lightweight Person Detector for Unmanned Aerial Vehicles." [Online]. Available: https://new.eurasip.org/Proceedings/Eusipco/eusipco2019/wp-content/uploads/2019/08/Fast_and_Lightweight_Person_Detector_for_Unmanned_Aerial_Vehicles.pdf
- [21] F. P. Scalcon *et al.*, "AI-Powered Video Monitoring: Assessing the NVIDIA Jetson Orin Devices for Edge Computing Applications," 2024 IEEE Transportation Electrification Conference and Expo (ITEC), pp. 1–6, Jun. 2024, doi: 10.1109/itec60657.2024.10598994.
- [22] X. Wang, Y. H. Hu, R. G. Radwin, and J. D. Lee, "Temporal Frame Sub-Sampling for video object tracking," *Journal of Signal Processing Systems*, vol. 92, no. 6, pp. 569–581, Nov. 2019, doi: 10.1007/s11265-019-01488-z.
- [23] X. Wang, Y. Hu, R. G. Radwin, and J. D. Lee, "Frame-Sub sampled, Drift-Resilient Long-Term video object tracking," 2022 IEEE International Conference on Multimedia and Expo (ICME), pp. 1–6, Jul. 2018, doi: 10.1109/icme.2018.8486499.
- [24] H. Araghi, J. Van Gemert, and N. Tomen, "Pushing the boundaries of event subsampling in Event-Based video classification using CNNs," in *Lecture notes in computer science*, 2025, pp. 276–292. doi: 10.1007/978-3-031-92460-6_17.
- [25] Z. Wang, D. Jiang, S. Dong, and Y. Gong, "Wave Energy Resource Availability Assessment in the Philippines based on 30-Year Hindcast data," *Journal of Ocean University of China*, vol. 22, no. 2, pp. 349–364, Mar. 2023, doi: 10.1007/s11802-023-5044-4.
- [26] J. C. Pacaldo, P. H. T. Bilger, and M. L. S. Abundo, "Nearshore Wave Energy Resource Assessment for Off-Grid Islands: A case study in Cuyo Island, Palawan, Philippines," *Energies*, vol. 15, no. 22, p. 8637, Nov. 2022, doi: 10.3390/en15228637.
- [27] T. Uda and Y. Noshi, "Recent Shoreline Changes Due to High-Angle Wave Instability along the East Coast of Lingayen Gulf in the Philippines," *Geosciences*, vol. 11, no. 3, p. 144, Mar. 2021, doi: 10.3390/geosciences11030144.
- [28] S. Ahmadi-Asl, M. G. Asante-Mensah, A. Cichocki, A.-H. Phan, I. Oseledets, and J. Wang, "Cross tensor approximation for image and video completion," *arXiv (Cornell University)*, Jan. 2022, doi: 10.48550/arxiv.2207.06072.
- [29] S. Abualroug, A. Alzubi, and K. Iyiola, "Inventory prediction using a modified Multi-Dimensional Collaborative wrapped Bi-Directional Long Short-Term Memory model," *Applied Sciences*, vol. 14, no. 13, p. 5817, Jul. 2024, doi: 10.3390/app14135817.
- [30] J. Y.-L. Chan *et al.*, "A Correlation-Embedded Attention Module to Mitigate Multicollinearity: an algorithmic trading application," *Mathematics*, vol. 10, no. 8, p. 1231, Apr. 2022, doi: 10.3390/math10081231.
- [31] D. Pileri, M. Cantono, A. Ferrari, A. Carena, and V. Curri, "Observing the effect of polarization mode dispersion on nonlinear interference generation in wide-band optical links," *OSA Continuum*, vol. 2, no. 10, p. 2856, Sep. 2019, doi: 10.1364/osac.2.002856.
- [32] A. A. Efanov, S. A. Ivliev, and A. G. Shagraev, "Welford's algorithm for weighted statistics," 2021 3rd International Youth Conference on Radio Electronics, Electrical and Power Engineering (REEPE), pp. 1–5, Mar. 2021, doi: 10.1109/reepe51337.2021.9387973.
- [33] A. A. Abdelmoneim, C. M. A. Kalaany, R. Khadra, B. Derardja, and G. Dragonetti, "Calibration of Low-Cost capacitive soil moisture sensors for irrigation management applications," *Sensors*, vol. 25, no. 2, p. 343, Jan. 2025, doi: 10.3390/s25020343.

- [34] S. Millán, C. Montesinos, and C. Campillo, "Evaluation of different commercial sensors for the development of their automatic irrigation system," *Sensors*, vol. 24, no. 23, p. 7468, Nov. 2024, doi: 10.3390/s24237468.
- [35] N. U. Amin et al., "Evaluation of crop phenology using remote sensing and decision support system for agrotechnology transfer," *Scientific Reports*, vol. 15, no. 1, Apr. 2025, doi: 10.1038/s41598-025-95109-4.
- [36] W. Addison-Atkinson, A. S. Chen, M. Rubinato, F. A. Memon, and J. D. Shucksmith, "Quantifying flood model accuracy under varying surface complexities," *Journal of Hydrology*, vol. 620, p. 129511, Apr. 2023, doi: 10.1016/j.jhydrol.2023.129511.
- [37] G. Wang et al., "Exploring the Water–Soil–Crop dynamic process and water use efficiency of typical irrigation units in the Agro-Pastoral Ecotone of northern China," *Plants*, vol. 13, no. 14, p. 1916, Jul. 2024, doi: 10.3390/plants13141916.
- [38] A. Singh, U. Kumar, and F. Seitz, "Remote sensing of storage fluctuations of poorly gauged reservoirs and State Space Model (SSM)-Based estimation," *Remote Sensing*, vol. 7, no. 12, pp. 17113–17134, Dec. 2015, doi: 10.3390/rs71215872.
- [39] S. Zauli-Sajani, S. Marchesi, C. Pironi, C. Barbieri, V. Poluzzi, and A. Colacci, "Assessment of air quality sensor system performance after relocation," *Atmospheric Pollution Research*, vol. 12, no. 2, pp. 282–291, Nov. 2020, doi: 10.1016/j.apr.2020.11.010.
- [40] M. Agostinho, A. Shani, R. Canajpa, and R. Treister, "Test–retest and interrater reliability of experimental within-subject variability of pain reports as assessed by the focused analgesia selection test," *PAIN Reports*, vol. 9, no. 5, p. e1175, Aug. 2024, doi: 10.1097/pr9.0000000000001175.
- [41] T. K. Koo and M. Y. Li, "A Guideline of selecting and Reporting Intraclass correlation coefficients for Reliability research," *Journal of Chiropractic Medicine*, vol. 15, no. 2, pp. 155–163, Apr. 2016, doi: 10.1016/j.jcm.2016.02.012.