An Automated Modular In Situ Machine Vision System for Real-Time Phytoplankton Monitoring Using the AFTI-scope

Andreas Vik Aasum[1], Isak Orlando Wagensteen[1], Matias Haugum[1][2222−−3333−4444−5555], Glaucia Moreira Fragoso[2][0000−0002−4497−2536], Christian Schellewald[2][0000−0002−2003−6612], Tor Arne Johansen[1][0000−0001−9440−5989], and Annette Stahl[1][0000−0002−8422−1091]

[1] Department of Engineering Cybernetics, Norwegian University of Science and Technology
[2] Sintef Ocean AS

Abstract. Phytoplankton are central to global biogeochemical cycles and aquatic food webs, yet their monitoring remains constrained by costly instrumentation and labor-intensive workflows. We present a modular, automated machine vision system for in situ, real-time detection, classification, and tracking of phytoplankton using the open-source Autonomous Flow-Through Imaging (AFTI) scope imaging platform. Our approach systematically benchmarks one-stage, two-stage, and transformer-based detectors, identifying YOLOv11 as the most effective balance of accuracy and efficiency for resource-constrained deployment. To improve species-level identification, we integrate an ensemble classifier (EfficientNet_B0 + Swin_T), boosting F1-scores to 0.9852, and couple this with BoT-SORT tracking to enable robust, non-redundant abundance estimation under dynamic flow conditions. Unlike prior systems, our pipeline is designed for embedded GPU hardware (Jetson AGX Orin) and validated on real-world deployments, demonstrating feasibility for scalable, autonomous plankton monitoring. This work presents an open-source flow-through imaging system with a real-time machine learning pipeline, along with integrated and standardized datasets combining public and AFTI-scope imagery. It further provides comprehensive benchmarking of detection, classification, and tracking architectures under in-situ conditions, and demonstrates in-field validation of real-time species-level monitoring with a design optimized for deployment on autonomous surface vehicles. This work provides the first embedded, low-cost platform capable of continuous, automated phytoplankton observation, supporting future large-scale marine ecosystem monitoring.

1 Introduction

Plankton, including unicellular photosynthetic phytoplankton, mixoplankton, and heterotrophic multicellular zooplankton, are fundamental to aquatic ecosystems and global biogeochemical cycles [34]. Phytoplankton contributes to nearly

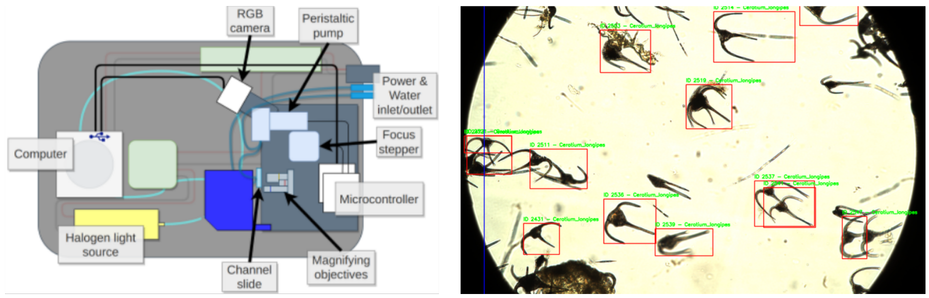

Fig. 1. AFTI-scope [16] imaging and analysis pipeline. **Left:** Schematic of the flow-through AFTI-scope hardware, including optical and control components. It contains the imaging backbone of the processing pipeline for automated detection, tracking, and ensemble-based classification of phytoplankton instances, presented in this paper (see flowchart in Figure 5 for more details). **Right:** Example output from a deployment sequence, where individual plankton are detected, classified, and tracked across frames.

half of the planet's oxygen and serves as the main primary producers of aquatic food webs. Through the biological carbon pump, they also play a vital role in long-term carbon sequestration. Due to their sensitivity to changes in abiotic factors (e.g. temperature, salinity, nutrient levels, pollution) and climate variability, plankton populations are widely used as bioindicators of ecosystem health [8].

Monitoring plankton communities is critical, but traditional methods such as net sampling and microscopy identification and quantification are labor-intensive, time-consuming, and offer limited spatial and temporal resolution [22]. To be able to achieve real-time, large-scale temporal and spatial coverage and rapidly process large volumes of samples, automated monitoring of phytoplankton biodiversity is needed.

Recent technological advances have introduced imaging systems that support automated plankton monitoring in real-time. Instruments such as Imaging FlowCytobot (IFCB, McLane, USA) and CytoSense (Cytobuoy, The Netherlands) combine flow cytometry with high-resolution imaging, allowing detailed taxonomic analysis [13, 22]. The AILARON AI-Driven Mobile Robotic Explorer, a mobile robotic platform for imaging, classifying, adaptive mapping, and revisiting plankton hotspots in situ, although with lower imaging magnification [30], also brings the state-of-the-art regarding rapid quantification of plankton distributions. However, cost and operational complexity remain a limitation for end users, both among plankton taxonomists and in autonomous robotic systems. Open-source, low-cost alternatives, such as the flow-through imaging microscope PlanktoScope (more recently Fairscope), address this gap by offering modular, accessible imaging solutions for smaller research initiatives and citizen science, increasing the scalability of plankton monitoring [27]. A strength of PlanktoScope is its integration with open online platforms, such as Ecotaxa, a web-based tool for the classification and annotation of plankton images. With Ecotaxa, users can easily upload images from the PlanktoScope/FairScope and manually or semi-automatically identify plankton taxa, building a growing communal database of labeled plankton imagery.

These systems, increasingly paired with machine learning models for plankton identification and quantification, allow for scalable, in-situ ecological monitoring. However, plankton identification still requires careful validation, which can be as time-consuming. Tools such as PyOPIA, an open source Python package for automated plankton image processing, is a pipeline for real-time plankton identification that includes machine learning-based image analysis [11]. Beyond PyOPIA, a broader set of recent research advances has focused on improving automated real-time plankton identification through deep learning methods. For example, single-stage detection architectures have been explored for fast classification of zooplankton in streaming applications [2], while segmentation-based approaches have been developed to allow in-situ instance and semantic labeling of plankton taxa [4, 29]. Supporting these models, background extraction and data augmentation techniques [6] help improve the segmentation of time-series images, and active learning strategies have been proposed to make expert labeling more efficient and representative [15, 14]. Together with advances in visual sensing frameworks, these efforts are part of a growing ecosystem of tools and methods that aim to accelerate high-throughput and accurate plankton monitoring.

In response to these developments, this paper presents a modular, vision-based pipeline for real-time *in-situ* detection, classification, and tracking of plankton, incorporating recent advances in deep learning. The system is designed for deployment on the AFTI-scope [16] a compact open-source imaging platform that features a flow-through chamber and a microscope for continuous image capture of waterborne microorganisms. This setup enables unattended, high-throughput monitoring in marine environments.

While traditional laboratory-based systems remain unsuitable for autonomous deployment, the AFTI-scope's compact flow-through design, low power consumption, and embedded GPU support make it adaptable to autonomous surface vehicles for continuous open-ocean monitoring. In contrast, instruments such as the IFCB or CytoSense provide high-resolution imaging but are intended for stationary or ship-based use and remain too large, costly, and power-demanding for autonomous vehicles. Open-source systems like the PlanktoScope emphasize accessibility but lack integrated real-time processing. By combining affordability, modularity, and embedded machine learning, the AFTI-scope enables autonomous in-situ monitoring not achieved by previous systems.

From a robotics perspective, the AFTI-scope is designed as a perception module for autonomous marine systems. Its compact form factor, low power consumption, and embedded GPU-based processing enable tight integration with unmanned surface vehicles (USVs) and other robotic platforms. The proposed vision pipeline operates fully on-board, enabling closed-loop perception without reliance on shore-based computation or post-processing. This allows autonomous robotic systems to perform continuous, adaptive, and long-duration plankton monitoring missions, aligning the system with core challenges in marine robotics such as autonomy, embedded perception, and real-time decision support.

The proposed pipeline integrates deep learning-based object detection, ensemble classification, and multi-object tracking to support species-level identification and abundance estimation under realistic operating conditions. In particular, our approach combines state-of-the-art object detection paradigms, including one-stage, two-stage, transformer-based, and algorithmic methods, and demonstrates that YOLOv11 [19] offers the best combination of accuracy and real-time performance for this application. We further enhance classification performance through ensemble strategies that improve F1-scores while maintaining efficiency. To address the critical challenge of robust, identity-preserving tracking in dynamic aquatic environments, we integrate the BoT-SORT [1] multi-object tracking algorithm, ensuring individual plankton are accurately counted even under irregular movement (Figure 1, right). This pipeline is evaluated in both live deployments and (harmful) algal bloom recordings, demonstrating its potential as a scalable solution for automated high-throughput plankton monitoring in marine ecosystems. While individual components build on existing detection, tracking, and classification methods, the novelty of this work lies in their systematic integration, benchmarking, and optimization under real-world, in-situ, and embedded constraints required for autonomous robotic deployment. The main contributions presented are:

- A modular, real-time pipeline that unifies detection, tracking, and ensemble-based classification, optimized for low-power embedded deployment (NVIDIA Jetson AGX Orin).
- Development of a domain-specific dataset combining public plankton repositories with AFTI-scope imagery to address class imbalance and morphological variability.
- Comprehensive model benchmarking, including one-stage, two-stage, and transformer-based detectors, with quantitative trade-offs between accuracy, speed, and resource usage.
- In-situ validation on real-world samples, demonstrating reliable species-level abundance estimation and highlighting limitations in dynamic aquatic environments.
- Demonstration of robotic deployment capability through tight integration with an USV, enabling fully autonomous, embedded, real-time perception for continuous open-ocean operation.
- Commitment to open science: upon acceptance, we will release code, trained models, and datasets to support reproducibility and future research.

In the following we review existing work in Section 1.1. Section 2 introduces the AFTI-scope imaging platform, datasets, and target plankton species underlying our pipeline. Section 3 details the proposed detection, classification, and tracking pipeline. Section 4 presents experimental results from initial deployments. Finally, Section 5 discusses limitations and future work, and concludes.

1.1 Related Work

The application of computer vision and deep learning to plankton image analysis has seen rapid growth over the last decade, driven by increased interest in automated environmental monitoring. A major bottleneck in this domain is the availability of high-quality, diverse training data for plankton species. Most existing datasets derived from the Imaging FlowCytobot (IFCB), such as WHOI22, WHOI40, and Kaggle-121, are limited in class diversity or exhibit significant class imbalance, making model generalization challenging [33, 26, 3]. Furthermore, plankton datasets often require expert-level annotation due to fine-grained morphological differences between species, while object-level annotations (e.g., bounding boxes) remain scarce.

Early approaches relied on traditional feature engineering and classical classifiers like SVMs, using hand-crafted descriptors tailored to specific datasets [33, 37]. While effective within narrow domains, these pipelines lack generalizability. Recent advances utilize convolutional neural networks (CNNs), often combined with transfer learning and ensemble methods, improving classification accuracy across benchmark datasets [24, 23, 10]. Techniques like test-time augmentation and deep ensembling have been shown to mitigate the effects of dataset shift [9].

Object detection, essential for in-situ applications where images contain multiple overlapping organisms, has increasingly been addressed through one-stage detectors like YOLO and two-stage detectors like Faster R-CNN. These models have demonstrated strong performance in multi-instance settings, especially when paired with high-resolution imaging systems [20, 2]. However, challenges remain in detecting small, transparent, or occluded objects, which are common in plankton imagery.

To address the temporal dimension of in-situ monitoring, object tracking methods like SORT have been proposed, enabling de-duplication and motion analysis across frames [5]. Although not yet widely adopted in marine vision, these methods offer promising avenues for real-time abundance estimation. This work builds on these advances by integrating object detection, classification, and tracking into a pipeline tailored for deployment on the AFTI-scope. Unlike many prior works focused on lab-acquired datasets, our approach emphasizes robustness under natural conditions, contributing towards the development of scalable and autonomous in-situ marine monitoring systems.

2 Background

2.1 The AFTI-scope Imaging System

The AFTI-scope (Autonomous Flow-Through Imaging Scope) [16] shown in Figure 2 is a compact, open-source imaging platform designed for real-time, in-situ observation of microscopic aquatic life. It enables continuous capture of high-resolution images as water flows through a transparent imaging channel beneath a microscope, eliminating the need for manual sampling.

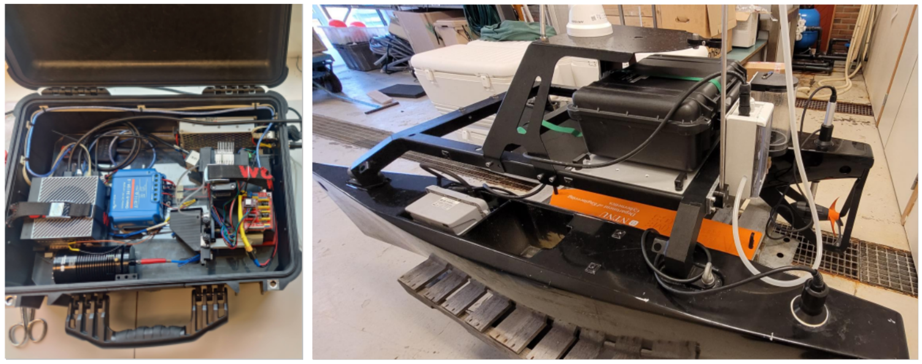

Fig. 2. Integration of the AFTI-scope imaging system for autonomous deployment. Left: Internal view of the AFTI-scope [16] electronics and flow-through imaging setup, including embedded GPU and control hardware housed in a waterproof case. Right: The complete system mounted on an Otter unmanned surface vehicle (USV), enabling continuous in-situ phytoplankton monitoring during open-ocean deployments.

Equipped with magnification levels ranging from 40x to 400x and a resolution of 0.41 µm/pixel[3], the AFTI-scope supports detailed imaging of both phytoplankton and zooplankton (size range 5 µm - 1000 µm). Its modular design allows for sensor customization and integration with RGB or other camera types. Stepper motors enable remote adjustment of focus and stage position, making it suitable for deployment on USVs.

The system is fully operable via open-source GRBL firmware [32] and Python-based control scripts, providing flexibility for field deployment and automation [16]. This setup forms the imaging backbone of the detection and classification pipeline presented in this paper.

2.2 Datasets for Detection, Classification and Tracking

The development of the pipeline's detection and classification modules relied on several distinct, custom-constructed datasets derived from both our own imagery and public sources. As summarized in Table 1, these include detection, classification, and tracking datasets constructed from AFTI-scope imagery and integrated with external repositories. Public datasets include the WHOI plankton datasets (WHOI22, WHOI40) [33], the PMID2019 phytoplankton dataset [20], and the Nalia-HAB dataset collected during a harmful algal bloom event using the AFTI-scope.

For the initial deployment task, five phytoplankton species were selected, sampled using the AFTI-scope [16], as shown in Figure 3. The selected target species were: Tripos longipes, Tripos lineatum, Tripos fusus, Chaetoceros decip-

[3] using a camera resolution of 2592x1944

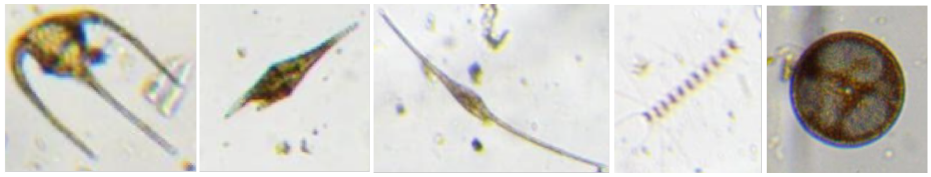

Fig. 3. Images captured using the AFTI-scope camera system [16] showing representative phytoplankton species: *Tripos longipes*, *Tripos lineatum*, *Tripos fusus*, *Chaetoceros decipiens*, and *Coscinodiscus sp.*, respectively.

iens, and Coscinodiscus sp. Representative samples from the WHOI dataset and the combined AFTI/PMID2019 imagery are illustrated in Figure 4.

Object Detection Dataset Our object detection pipeline was trained on a custom-built dataset designed to enhance class diversity. The foundation of this dataset consists of 229 images collected using an AFTI-scope, which were selected from over 2000 initial samples after a rigorous quality-filtering process. All images were manually annotated in MakeSense [31] using an inclusive protocol where all discernible instances were marked, a strategy chosen to reduce false negatives in detection. Annotations were verified by domain experts.

The initial AFTI-scope dataset exhibited significant class imbalance. To mitigate this, we augmented our training data with a filtered subset of the public PMID2019 dataset [20]. This added 633 relevant images, after we mapped five overlapping species and adjusted class labels accordingly (e.g., *Ceratium* spp. was mapped to *Tripos longipes*). Per-batch augmentations, including spatial (rotation, flipping) and pixel-level transformations, were applied during training via the Ultralytics framework [19] to improve model robustness.

Classification Datasets For the classification module, we prepared two key datasets to support transfer learning and evaluate performance on target imagery. The first is a large external pre-training dataset combining images from the WHOI-plankton repository with additional minority-class samples. Data standardization and cleaning were required, including manual re-classification (e.g., splitting a general *Ceratium* group into three species) and discarding irrelevant WHOI classes. To mitigate class imbalance, the dominant *Chaetoceros* class was down-sampled from over 10,000 images to create a more balanced training distribution.

The second dataset was designed for fine-tuning and evaluation. It consists of image patches cropped from AFTI-scope bounding box annotations, ensuring direct comparability between the detector and a separate classifier module. Following a context-preserving strategy, each crop was padded to at least 224×224 pixels to maintain detail and provide consistent classifier input. This final AFTI-scope classification set is more balanced and serves as the primary benchmark for evaluating performance on our own data.

Multi-Object Tracking Dataset For the qualitative evaluation of the multi-object tracking module, a dedicated video sequence from the AFTI-scope was used. This sequence, referred to as the Nalia-HAB dataset, captures a Harmful Algal Bloom (HAB) event of Tripos longipes over 398 frames. A 105-frame segment was selected for the assessment because beyond this point, increased pump speed compromised the smooth motion capture necessary for tracking. Within this segment, the sampling rate was sufficient to resolve continuous trajectories. It is important to note that due to resource limitations, ground-truth tracks were not created for this dataset; therefore, the resulting evaluation is qualitative only. Additionally, this footage was recorded using an older version of the AFTI-scope's camera system, which could potentially affect the detection module's performance.

To avoid data leakage, we ensured that images from the same video sequence never appeared in both training and testing sets. All annotations were performed using the MakeSense tool and cross-verified by domain experts. For the tracking dataset (Nalia-HAB), we excluded segments with pump speeds above 0.67 ml/min, where large frame-to-frame displacements prevented reliable manual annotation. We report this limitation and plan to extend future work with fully annotated, high-speed sequences to capture these challenging conditions.

Table 1. Overview of datasets used for detection, classification, and tracking.

Dataset (Usage)	Source	Classes	Samples	Notes
AFTI-Detection	AFTI + PMID2019 subset	5	862 images	Result on val. set
AFTI-Classification	Crop AFTI det. + WHOI	5	4,200 patches	Result on val. set
Nalia-HAB Tracking	AFTI video	1 (Tripos)	398 frames	105 frames (qual.)
WHOI40 subset (pre-tr.)	WHOI public repository	40	3,000 images	ext. pre-train only

3 Real-time in-situ Abundance Estimation

In this paper, a modular phytoplankton analysis pipeline as shown in Figure 5 is proposed, decomposing the over all task of abundance estimation into three main components (i.e. Object Detection, Object Tracking and Ensemble Classification). This modular design simplifies development and enables each stage to be optimized independently. The detection component can prioritize recall, deferring fine-grained classification to a more powerful ensemble-based classifier. Tracking operates on detection outputs and can be evaluated with or without re-identification using deep feature matching. Classification can be deferred or performed on high-resolution crops of tracked instances. This sequential breakdown facilitates targeted improvements, easier testing, and greater system flexibility. New algorithms or models can be integrated into individual modules without redesigning the entire system. A schematic overview of the pipeline is shown in Figure 5. In the following, we provide a detailed description of each module,

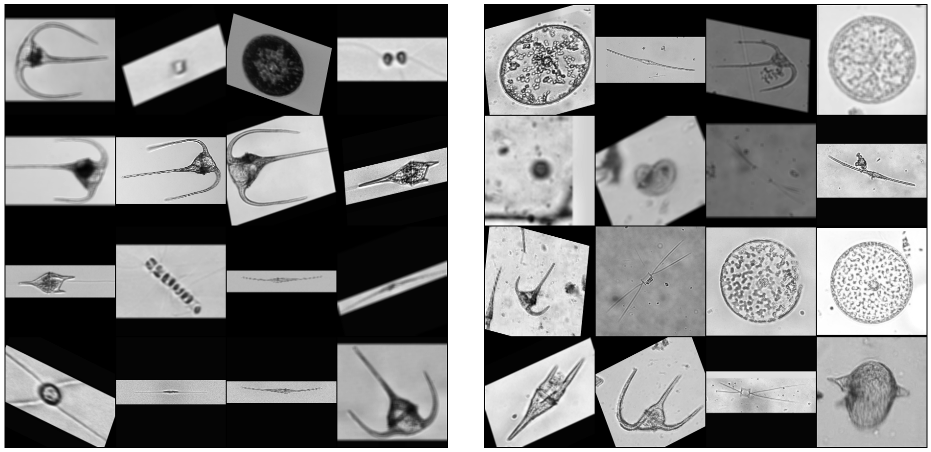

Fig. 4. Left: Examples from the WHOI dataset (cropped). Examples form the NaliaHAB Tracking dataset are shown in Fig. 6 and Fig. 7. Right: Examples images (cropped) from the AFTI and PMID2019 dataset.

along with the rationale for selecting specific models, and an analysis of their performance and suitability for the overall system.

3.1 Object Detection

In the following, we present a comparative evaluation of three object detection architectures applied to plankton detection: YOLOv11 [19], RT-DETR-L [36], and Faster R-CNN [28]. The analysis spans three dimensions: single-class detection accuracy, multi-class detection robustness, and computational efficiency. These aspects are critical for determining the most suitable model for deployment within the resource-constrained AFTI-scope platform.

To identify the best trade-off between detection accuracy and computational efficiency for single-class plankton detection, we evaluated five YOLOv11 model variants of varying sizes: n (nano), s (small), m (medium), l (large), and x (extra-large). These models balance speed and accuracy differently: models n and s are optimized for lightweight, real-time use with minimal computational demand, while model m serves as a balanced option, and l and x offer higher accuracy at the cost of increased computational complexity. YOLO11n is the lightest model in the family, with 2.6 million parameters, offering fast inference while maintaining competitive accuracy, whereas YOLO11m is a more complex variant with 20.1 million parameters designed to provide improved detection performance. All models were trained and validated on 640×640 px images with a batch size of 4 to assess detection accuracy, and computational benchmarks such as framerate and latency were conducted with a batch size of 16 to better approximate realistic throughput by amortizing fixed overheads (e.g., I/O and CUDA launch costs). Testing, training and validation were carried out on a worksta-

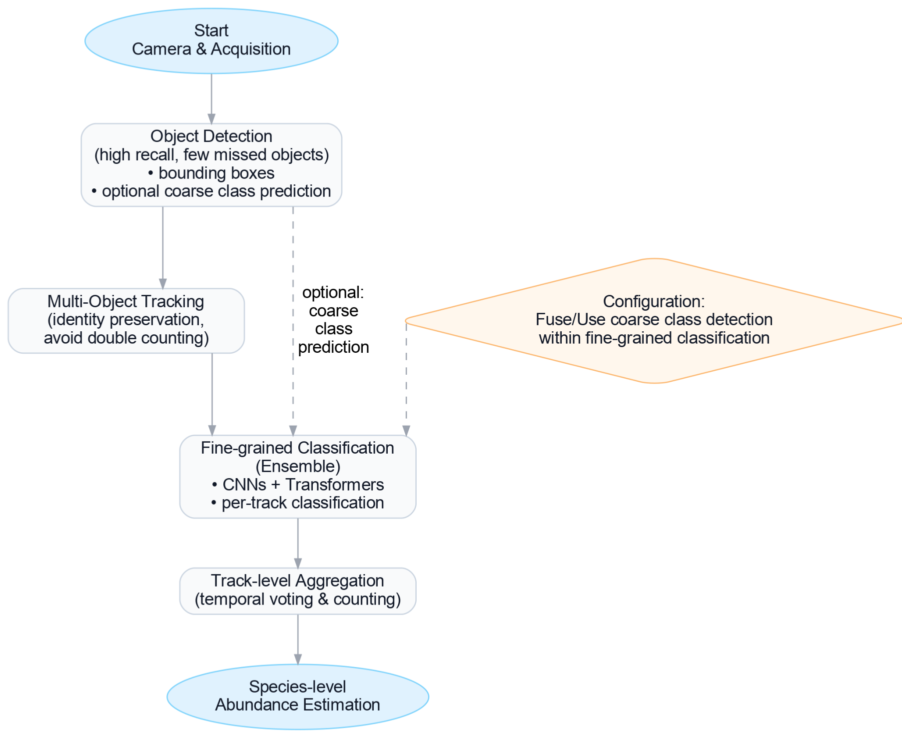

Fig. 5. Overview of the real-time phytoplankton analysis pipeline for species-level abundance estimation. Solid arrows indicate the execution order: object detection is tuned for high recall to minimize missed organisms, followed by multi-object tracking to preserve identity and avoid double counting. Fine-grained species classification is performed using an ensemble of classifiers operating on tracked instances, after which track-level aggregation applies temporal voting and counting to produce stable abundance estimates. Dashed arrows denote optional information flow and configuration influence: depending on the selected configuration, the detector's coarse class prediction may be fused into the fine-grained classification stage.

tion with Ubuntu 22.04, an AMD Ryzen 9 5900X CPU, 62 GB RAM, and an NVIDIA RTX 3070 GPU (8 GB VRAM). The results of the detection and computational performance evaluations are summarized in Table 2 and Table 3, respectively.

Model	Precision	Recall	F1	50	75	50:95
YOLO11n	**0.888**	0.864	0.876	0.908	0.793	0.684
YOLO11s	0.886	0.842	0.863	0.889	0.809	0.683
YOLO11m	0.885	**0.876**	**0.880**	**0.916**	**0.823**	**0.709**
YOLO11l	0.884	0.866	0.875	0.906	0.788	0.682
YOLO11x	0.852	0.819	0.835	0.874	0.781	0.666

Table 2. Detection metrics for YOLO variants (640×640 px, batch size 4). Columns 50, 75, and 50:95 denote mAP at IoU thresholds of 50%, 75%, and averaged from 50% to 95%, respectively.

Model	FPS	Lat. (ms)	Mem (MB)	GFLOPs	MParams
YOLO11n	**95.6**	**10.5**	**325**	**6**	**2.6**
YOLO11s	81.9	12.2	575	22	9.4
YOLO11m	65.7	15.2	1066	68	20.1
YOLO11l	59.6	16.8	1235	87	25.3
YOLO11x	44.1	22.7	1886	195	56.9

Table 3. Computational performance of YOLOv11 variants (batch size 16). “FPS” = frames per second; “Lat.” = latency per image (ms); “Mem” = GPU memory usage in MB; “GFLOPs” = giga floating-point operations; “MParams” = million parameters. Measurements exclude initialization overheads.

Table 2 shows that YOLO11m achieves the highest overall detection accuracy, with leading Recall, F1-Score, and all mAP metrics. Notably, scaling up further to YOLO11l and YOLO11x yields diminishing returns, likely due to suboptimal default training hyperparameters for larger models.

Table 3 highlights the computational efficiency of smaller models, with YOLO11n achieving the highest FPS (95.59) and minimal memory usage. YOLO11m, however, maintains robust computational performance with FPS reaching 65.69 while delivering superior detection accuracy.

Balancing these trade-offs, the YOLO11m model was selected for further development, with YOLO11n serving as a lightweight benchmark in subsequent analyses.

Following the selection of YOLO11m, the effect of varying training batch size was investigated while maintaining an image resolution of 640×640 px. Results are shown in Table 4.

BS	Precision	Recall	F1-Score	mAP50	mAP75	50:95
4	0.885	**0.876**	**0.880**	**0.916**	**0.823**	**0.709**
8	**0.891**	0.850	0.870	0.903	0.799	0.688
12	0.853	0.858	0.856	0.892	0.802	0.689

Table 4. Detection performance of YOLO11m with varying batch sizes (BS) at 640×640 px resolution.

Table 4 indicates that a batch size of 4 yielded the highest F1-Score (0.8803) and mAP50:95 (0.7091), while batch size 8 provided slightly better precision (0.8906). Performance declined slightly with batch size 12. The relatively small differences in performance observed across batch sizes (4, 8, 12) can be explained by the Ultralytics YOLO framework's use of gradient accumulation, which effectively simulates a larger nominal batch size of 64 by summing gradients over multiple smaller hardware batches. This approach stabilizes learning and minimizes sensitivity to hardware batch size. However, some residual variation remains because Batch Normalization layers compute their statistics independently within each hardware batch, introducing subtle differences in training dynamics and final model performance, so careful batch size tuning remains important.

In addition, different optimizer configurations, such as SGD and AdamW within the Ultralytics framework [19] were tested to understand their impact on training performance. All runs used consistent validation thresholds chosen to better reflect true model quality. Results showed that the SGD optimizer achieved the highest precision, F1-score, and advanced mAP metrics, while the Auto (Default) setting produced the best recall and mAP50. Analysis indicates that the 'Auto' mode internally switches between SGD and AdamW while applying its own learning rates and momentum settings, which can override user-specified values. AdamW performed less effectively overall, likely due to its sensitivity to learning rate choices; the default rate used was too high for optimal AdamW behavior. These findings suggest that while SGD offers robust performance with default hyperparameters, adaptive optimizers like AdamW require careful tuning of learning rate, betas, and weight decay to reach their full potential.

Overall Single-Class Detection Performance

All three models (YOLO11m, RT-DETR-L, and Faster R-CNN) were benchmarked on a single-class detection task using their optimal parameter settings, and evaluated with mean Average Precision (mAP) scores as well as traditional classification metrics. RT-DETR-L achieved the highest mAP across all thresholds (e.g., mAP@0.50:0.95 = 0.6861), closely followed by YOLO11m (0.6701), while Faster R-CNN lagged behind (0.6020). In terms of F1-score, RT-DETR-L (0.8743) and YOLO11m (0.8651) performed comparably, with Faster R-CNN slightly lower (0.8450) but exhibiting the highest recall (0.8770).

Model	Precision	Recall	F1	mAP50	mAP75	50:95
YOLO11m	0.875	0.856	0.865	0.930	0.773	0.670
RT-DETR-L	**0.888**	0.861	**0.874**	**0.934**	**0.795**	**0.686**
Faster R-CNN	0.814	**0.877**	0.845	0.896	0.677	0.602

Table 5. Comparison of Single-Class Precision, Recall, and F1-Scores for YOLO11m and RT-DETR-L derived from Ultralytics validation output. Faster R-CNN reports Precision, Recall, and F1-Scores at Score Thr: 0.4, IoU Thr: 0.6. Columns mAP50, mAP75, and 50:95 indicate mAP at IoU thresholds of 50%, 75%, and averaged from 50% to 95%, respectively.

An examination of the standardized mAP scores (Table 5) shows that the RT-DETR-L configuration achieves the highest performance, establishing it as the most accurate model in this single-class comparison, with YOLO11m a close second. Considering Precision, Recall, and F1-scores: RT-DETR-L also leads in F1-score, but Faster R-CNN, which was specifically optimized for this metric, achieves the highest recall (0.8770). However, these results should be contextualized: Faster R-CNN's lower mAP may reflect a less extensive tuning phase, while RT-DETR-L used parameters guided by its source paper. YOLO11m's strong performance was achieved with general-purpose 'Auto' settings, suggesting room for further optimization.

Multi-Class Detection Performance

Model	Precision	Recall	F1	mAP50	mAP75	50:95
YOLO11m	**0.8303**	**0.8671**	**0.8483**	**0.8751**	**0.7467**	**0.6237**
RT-DETR-L	0.802	0.857	0.825	0.867	0.702	0.608
Faster R-CNN	0.832	0.858	0.845	0.833	0.585	0.537

Table 6. Comparison of Multi-Class detection performance.

The overall multi-class performance metrics in Table 6 highlight YOLO11m as the top-performing model, achieving the highest scores across key categories. In addition to its balanced detection accuracy, YOLO11m maintained a robust 65.69 FPS with a peak memory usage of 1065.50 MB, approximately 39% faster than RT-DETR-L (47.15 FPS, more memory-intensive) and more than twice as fast as Faster R-CNN (27.97 FPS, with the highest memory consumption at 4873.81 MB). This combination of accuracy, robustness across classes, and superior computational efficiency makes YOLO11m the most suitable choice for real-time deployment on the resource-constrained AFTI-scope platform.

The tuned model YOLO11m showed consistent improvements across overall metrics, with higher precision, F1-score, and mAP50:95 values. Per-species analysis further confirmed these gains, as shown in Table 7, with notable increases for challenging classes such as *Chaetoceros* and *Tripos fusus*, demonstrating that the

Class	Precision	Recall	F1	mAP50	mAP75	50:95
Tripos longipes	**0.946**	0.886	**0.915**	**0.975**	**0.853**	**0.727**
Tripos fusus	0.878	**0.944**	0.910	0.923	0.923	0.737
Chaetoceros	0.742	0.695	0.718	0.709	0.335	0.343
Tripos lineatum	0.869	0.900	0.884	0.944	0.761	0.659
Coscinodiscus	0.940	0.935	0.937	0.973	0.850	0.707

Table 7. Per-species detection performance for YOLO11m (with AdamW hyperparameters). Columns 50, 75, and 50–95 indicate mAP at IoU thresholds of 50%, 75%, and averaged from 50% to 95%, respectively.

optimized hyperparameters improved the model's ability to accurately identify difficult cases and generalized well to multi-class detection.

While RT-DETR-L led in single-class accuracy, YOLO11m emerged as the most robust and efficient detector when considering all dimensions. Its superior performance in the multi-class scenario, combined with its high computational efficiency and out-of-the-box generalizability, makes it the most suitable candidate for deployment and further optimization.

3.2 Object Tracking

The tracking module is essential for enabling abundance estimation by facilitating the counting of individual plankton instances across video frames.

From the literature on visual tracking, it is evident that state-of-the-art online trackers primarily follow the tracking-by-detection paradigm. BoT-SORT [1] and ByteTrack [35], have demonstrated strong performance and were therefore selected for evaluation and implementation.

The tracking module was integrated with the Yolo11m plankton detection and classification modules. Since the Ultralytics framework does not support re-identification (Re-ID) features, the BoxMot library [7] was adopted to enable deep feature descriptor-based similarity matching.
The YOLO + BoT-SORT approach showed reliable performance in occlusion scenarios, effectively distinguishing adjacent instances without merging them into single detections. It also produced bounding boxes that more consistently encompassed the full extent of target objects, supporting accurate classification.

To produce abundance estimates, two counting strategies were considered: Counting objects only when they cross a predefined reference line (cf. blue line in Figures 6, 7); Counting each unique track ID throughout the video. The first method was selected to reduce double-counting due to identity switches. This approach ensures that only consistent and complete trajectories contribute to the final abundance estimate.

To ensure label stability during classification, a majority-voting mechanism was applied to the sequence of predicted class labels for each track. This reduces the impact of transient misclassifications that could otherwise affect abundance estimates when classification is active.

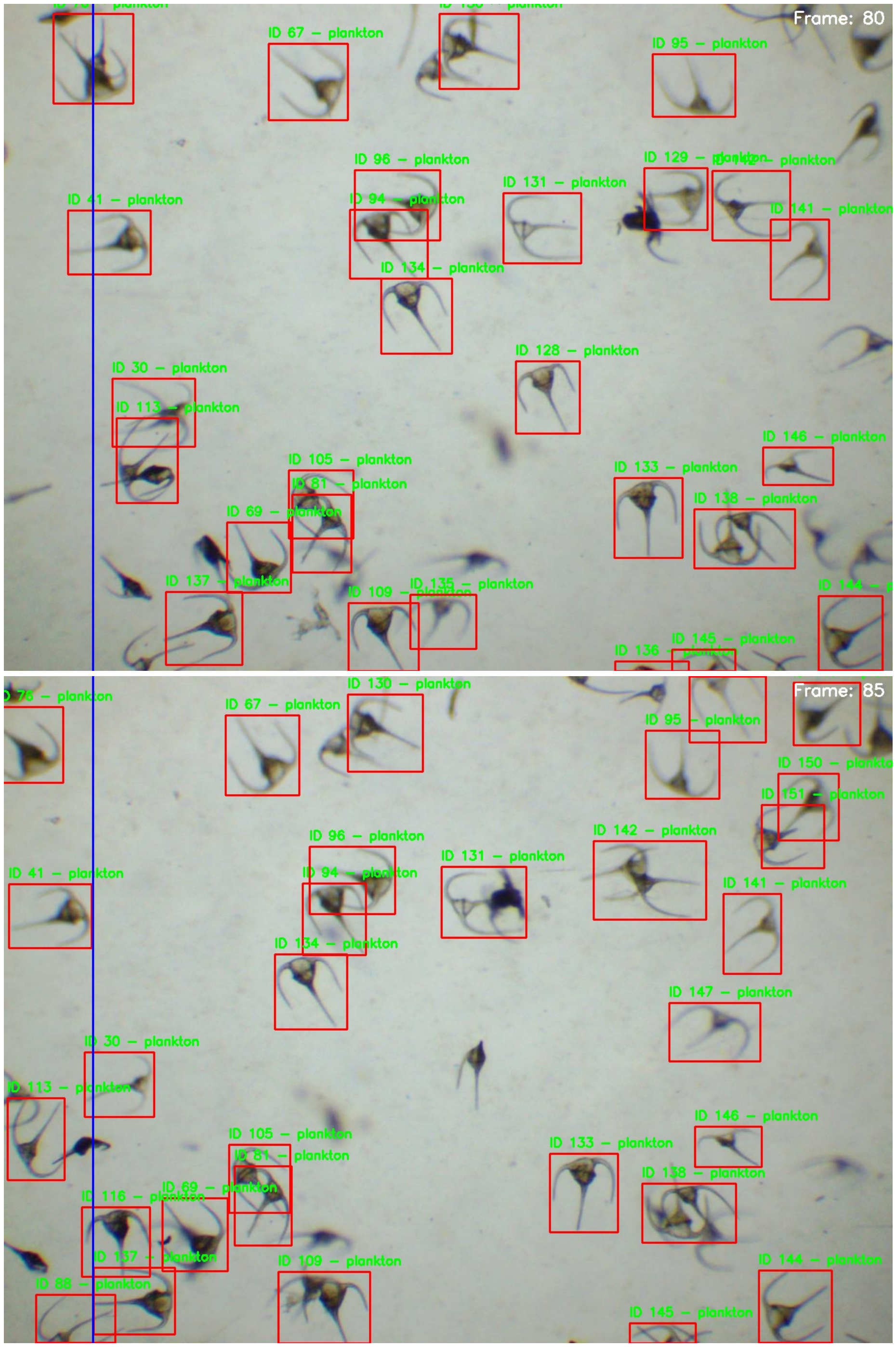

Fig. 6. Examples from the YOLO tracking test on the Nalia-HAB set. Frames shown are 80, and 85 separated by 4-frame intervals to illustrate the movement of tracked instances. A blue line was included as location and count reference.

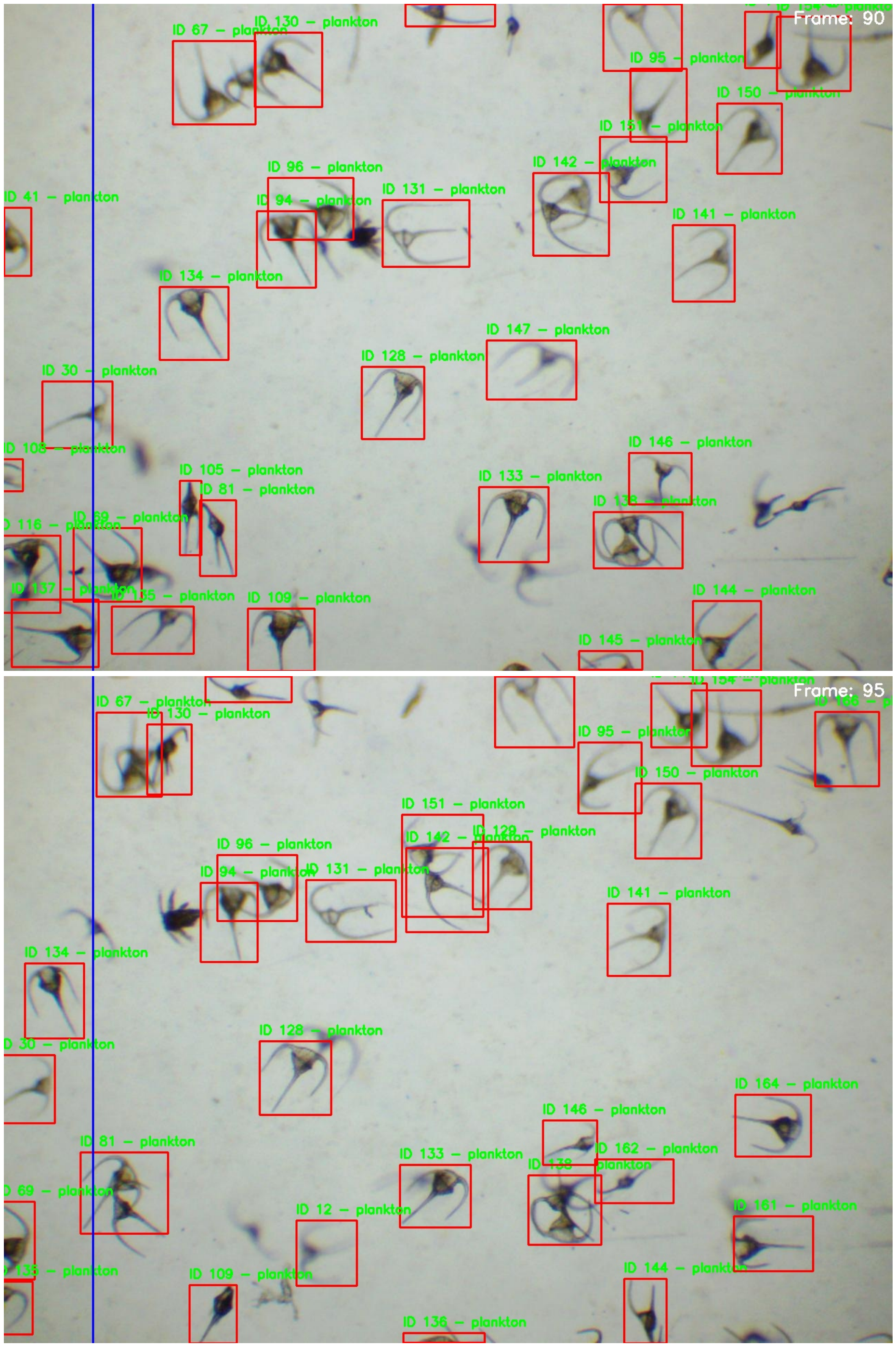

Fig. 7. Examples from the YOLO tracking test on the Nalia-HAB set. Frames shown are 90, and 95, separated by 4-frame intervals to illustrate the movement of tracked instances. A blue line was included as location and count reference.

3.3 Ensemble Classification

The ensemble classification module, the final stage of the plankton detection pipeline, is critical to achieving robust and accurate abundance estimates of selected target classes. To ensure optimal performance, a diverse set of neural network architectures was explored, including ResNet [17], DenseNet [18], EfficientNet [25], Vision Transformer (ViT) [12], and Swin Transformer [21]. These models were selected based on their prior success in visual classification tasks and their varied inductive biases, offering complementary strengths.

Each model was trained on both legacy plankton datasets and newly collected data with the AFTI-scope as listed in Section 2.2. A standardized training pipeline was implemented, featuring runtime data augmentation, weighted sampling to counter class imbalance, and fine-tuning for domain adaptation. Despite applying transfer learning strategies, significant domain shift between datasets limited pre-training benefits. However, the transformers, particularly ViT and Swin, demonstrated greater resilience to domain shift.

Post fine-tuning, all models exceeded an F1-score of 0.9, with EfficientNet_B0 achieving the highest score of 0.962 (cf. Table 8). A comparative analysis of computational efficiency also revealed EfficientNet_B0 to be the most resource-efficient model.

Method	F1 Score	GFLOPs
Individual Models		
EfficientNet_B0	**0.962**	**0.39**
Swin_T	0.942	4.49
ViT_b_16	0.940	17.56
ResNet50	0.912	4.09
DenseNet121	0.908	2.83
Ensembles (True Positives Only)		
YOLO11n + Swin_T	0.9777	–
YOLO11m + EfficientNet_B0 + Swin_T	**0.9852**	–

Table 8. Final classification performance (F1-score) and computational cost (GFLOPs) for individual models and top ensemble configurations after fine-tuning on the AFTI dataset.

Ensemble strategies were implemented to combine the best individual classifiers with YOLO-based detections, using both weighted and majority voting schemes. These methods consistently improved classification accuracy on true positive detections, with top-performing configurations achieving F1-scores of 0.9777 (YOLO11n + Swin_T) and 0.9852 (YOLO11m + EfficientNet_B0 + Swin_T), compared to baseline scores of 0.9640 and 0.9809, respectively.
Table 8 summarizes these results, showing both the fine-tuned F1-scores and computational costs for the evaluated architectures. EfficientNet_B0 stood out

among individual models with the highest F1-score (0.962) and lowest computational complexity (0.39 GFLOPs), making it particularly well-suited for deployment on resource-constrained systems. Transformer-based architectures such as Swin_T and ViT_b_16 offered strong robustness to domain shift but required greater computational resources. These findings highlight the value of combining complementary model types while emphasizing the need to balance accuracy, efficiency, and robustness in final system design.

4 Deployment and In-situ Measurements

Multiple combinations of detection, classification, and tracking modules were tested to assess real-time performance and robustness during deployment of the developed pipeline, which runs on an NVIDIA Jetson AGX Orin Developer Kit with 64 GB RAM integrated into the AFTI-scope. Specifically, the top-performing YOLO detectors (both single-class (YOLO11m) and multi-class (YOLO11n)) were evaluated in combination with leading CNN (EfficientNet) and transformer-based (Swin_T) classifiers. All setups employed BoT-SORT via the BoxMot library for tracking.

To facilitate integration and testing independent of AFTI-scope hardware, a Docker image identical to the on-device environment was deployed on an external GPU-enabled workstation. A streaming simulator was implemented to loop over the AFTI-scope Nalia-HAB dataset, enabling consistent and repeatable testing conditions. The acquisition and pipeline parameters are summarized in Table 9.

Parameter	Value
Acquisition Parameters	
Exposure time	1400-2000 μs
Pump speed	0.335 - 6.5 ml/min
Frame rate	30 fps
Magnification/resolution	1920x1200
Pipeline Parameters	
Confidence threshold	0.001
Min existence frames	5
Crop margin	10 pixels

Table 9. Test parameters overview.

Each pipeline successfully processed AFTI-scope video streams in real-time. The YOLO11m + Swin_T setup exhibited slightly higher latency due to its larger model size but remained suitable for real-time applications. A representative image is shown in Figure 1 to the right.

The deep learning pipelines displayed strong performance, effectively detecting and classifying minority classes and exhibiting robustness against occlusion and visual artifacts. Despite some instances of under-sized bounding boxes and occasional grouping of multiple objects into one detection, both classification and tracking modules performed well. Misclassification of non-target objects was observed when such objects were morphologically similar to the target classes.

The overall tracking performance was high, with consistent instance identity preservation despite occlusions. Some weaknesses were noted near frame edges and during rapid object rotation.

Runtime Speed for Pumps. Tracking reliability was tested across flow rates ranging from 0.335 ml/min to 6.5 ml/min. Above 0.667 ml/min, tracklets were not consistently initialized due to large object displacements between frames. Optimal tracking was achieved at lower speeds (~0.335 ml/min).

Optimal Camera Exposure Time for Sample Detection. Controlled tests determined 1700 µs as the optimal exposure time, balancing contrast and avoiding overexposure. This value worked well across both test samples. Higher exposures (e.g., 2000 µs) remained effective, particularly in Lugol-preserved samples.

5 Conclusion

This paper presented a modular pipeline for real-time detection, classification, and tracking of phytoplankton using the AFTI-scope, a compact open-source *in situ* imaging system designed for continuous, automated monitoring of marine microorganisms.

A comparative evaluation of state-of-the-art object detection paradigms, including the one-stage YOLOv11, two-stage Faster R-CNN, and transformer-based RT-DETR, demonstrated that, despite varying strengths across models, YOLOv11 offered the best overall balance between accuracy and real-time performance. To enhance its classification capabilities, ensemble strategies that incorporated additional classifiers were explored. These approaches improved the classification F1-score from 0.9640 to 0.9777 for the lightweight YOLO11n variant, and from 0.9809 to 0.9852 for the more complex YOLO11m, highlighting the effectiveness of ensembling in this domain.

A key challenge addressed was ensuring reliable tracking to prevent double-counting plankton. Due to the irregular and species-dependent nature of plankton movement, simple frame-based heuristics were insufficient. Instead, a state-of-the-art multi-object tracking algorithm, BoT-SORT, was integrated into the pipeline. Qualitative assessments during live deployments and on a recorded harmful algal bloom dataset confirmed its effectiveness, with future evaluations using tracking-annotated datasets planned to further validate performance. From a robotics standpoint, this work demonstrates how modern embedded vision pipelines can be co-designed with sensing hardware to enable autonomous ecological monitoring, positioning the AFTI-scope as a practical perception subsystem for marine robotic platforms.

Limitations and Future Work. While our pipeline demonstrates feasibility for real-time plankton monitoring, several limitations remain. First, our current evaluation covers only five phytoplankton species, restricting taxonomic diversity. Second, all samples were collected from a limited geographic region and season, which may reduce generalizability to broader oceanographic settings. Third, our tracking evaluation was qualitative only, due to the absence of fully annotated ground-truth trajectories in the Nalia-HAB dataset.

As a next step, we plan to expand the dataset to include additional ecologically relevant and toxic species (e.g., *Karenia brevis*), increase geographic and seasonal coverage, and develop fully annotated multi-object tracking benchmarks. This expansion will improve robustness and ecological relevance, while enabling more comprehensive quantitative evaluation of the tracking module.

Acknowledgments. This research was supported by the Research Council of Norway through the IKTPLUSS project AWAS (grant number 327670).

Disclosure of Interests. The research and resources used in this paper originates from work done at the Department of Engineering Cybernetics at the Norwegian University of Science and Technology. The authors have no competing interests that are relevant to the contents of this article.

References

1. Aharon, N., Orfaig, R., Bobrovsky, B.Z.: Bot-sort: Robust associations multi-pedestrian tracking. arXiv preprint arXiv:2206.14651 (2022)
2. Ansari, S., Desai, D.V., Saad, A., Stahl, A.: Implications of single-stage deep learning networks in real-time zooplankton identification. CURRENT SCIENCE **125**(11) (2023)
3. Aurelia, Luo, J., Allen, J.B., Sullivan, J., Mills, S., Cukierski, W.: National data science bowl. https://kaggle.com/competitions/datasciencebowl (2014), kaggle
4. Bergum, S., Saad, A., Stahl, A.: Automatic in-situ instance and semantic segmentation of planktonic organisms using mask r-cnn. In: Global Oceans 2020: Singapore-US Gulf Coast. pp. 1–8 (2020)
5. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: 2016 IEEE International Conference on Image Processing (ICIP). pp. 3464–3468 (2016)
6. Borgersen, J.N., Saad, A., Stahl, A.: Mog: A background extraction approach for data augmentation of time-series images in deep learning segmentation. In: Fourteenth International Conference on Machine Vision (ICMV 2021). vol. 12084, pp. 360–368 (2022)
7. Broström, M.: BoxMOT: pluggable SOTA tracking modules for object detection, segmentation and pose estimation models. https://doi.org/https://zenodo.org/record/7629840, https://github.com/mikel-brostrom/boxmot
8. Chandel, P., Mahajan, D., Thakur, K., Kumar, R., Kumar, S., Brar, B., Sharma, D., Sharma, A.K.: A review on plankton as a bioindicator: A promising tool for monitoring water quality. World Water Policy **10**(1), 213–232 (2024)
9. Chen, C., Kyathanahally, S.P., Reyes, M., Merkli, S., Merz, E., Francazi, E., Hoege, M., Pomati, F., Baity-Jesi, M.: Producing plankton classifiers that are robust to dataset shift. Limnology and Oceanography: Methods **23**(1), 39–66 (2025)

10. Dai, J., Wang, R., Zheng, H., Ji, G., Qiao, X.: Zooplanktonet: Deep convolutional network for zooplankton classification. In: OCEANS 2016 - Shanghai. pp. 1–6 (2016)
11. Davies, E.J., Nimmo-Smith, W.A.M., Nepstad, R., Nordam, T., Brönner, U., Steinvik, A., Barstein, K.H.: Pyopia: A python ocean particle image analysis toolbox. In: Particles in Europe (PiE) Conference. Crete, Greece (Oct 2023), https://par.nsf.gov/servlets/purl/10544971, academic lecture, SINTEF Ocean
12. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: International Conference on Learning Representations (2021), https://openreview.net/forum?id=YicbFdNTTy
13. Eerola, T., Batrakhanov, D., Barazandeh, N.V., Kraft, K., Haraguchi, L., Lensu, L., Suikkanen, S., Seppälä, J., Tamminen, T., Kälviäinen, H.: Survey of automatic plankton image recognition: challenges, existing solutions and future perspectives. Artificial Intelligence Review **57**(5), 114 (April 2024)
14. Haug, M.L., Saad, A., Stahl, A.: Ciral: A hybrid active learning framework for plankton taxa labeling. IFAC-PapersOnLine **54**(16), 450–457 (2021)
15. Haug, M.L., Saad, A., Stahl, A.: A combined informative and representative active learning approach for plankton taxa labeling. In: Thirteenth International Conference on Digital Image Processing (ICDIP 2021) (2021)
16. Haugum, M., Fragoso, G.M., Bøe Henriksen, M., Zolich, A.P., Arne Johansen, T.: Autonomous flow-through rgb and hyperspectral imaging for unmanned surface vehicles. In: OCEANS 2023 - Limerick. pp. 1–8 (2023)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
18. Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K.Q.: Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4700–4708 (2017)
19. Jocher, G., Qiu, J.: Ultralytics yolo11. https://github.com/ultralytics/ultralytics (2024)
20. Li, Q., Sun, X., Dong, J., Song, S., Zhang, T., Liu, D., Zhang, H., Han, S.: Developing a microscopic image dataset in support of intelligent phytoplankton detection using deep learning. ICES Journal of Marine Science **77**(4), 1427–1439 (2020)
21. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 10012–10022 (2021)
22. Lombard, F., Boss, E., Waite, A.M., Vogt, M., Uitz, J., Stemmann, L., Sosik, H.M., Schulz, J., Romagnan, J.B., Picheral, M., Pearlman, J., Ohman, M.D., Niehoff, B., Möller, K.O., Miloslavich, P., Lara-Lpez, A., Kudela, R., Lopes, R.M., Kiko, R., Karp-Boss, L., Jaffe, J.S., Iversen, M.H., Irisson, J.O., Fennel, K., Hauss, H., Guidi, L., Gorsky, G., Giering, S.L.C., Gaube, P., Gallager, S., Dubelaar, G., Cowen, R.K., Carlotti, F., Briseño-Avena, C., Berline, L., Benoit-Bird, K., Bax, N., Batten, S., Ayata, S.D., Artigas, L.F., Appeltans, W.: Globally consistent quantitative observations of planktonic ecosystems. Frontiers in Marine Science **6** (2019)
23. Lumini, A., Nanni, L.: Deep learning and transfer learning features for plankton classification. Ecological informatics **51**, 33–43 (2019)

24. Orenstein, E.C., Beijbom, O.: Transfer learning and deep feature extraction for planktonic image data sets. In: 2017 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 1082–1088. IEEE (2017)
25. Pastore, V.P., Ciranni, M., Bianco, S., Fung, J.C., Murino, V., Odone, F.: Efficient unsupervised learning of biological images with compressed deep features. Image and Vision Computing **137**, 104764 (2023)
26. Pastore, V., Zimmerman, T., Biswas, S., Bianco, S.: Annotation-free learning of plankton for classification and anomaly detection. sci. rep. 10 (1), 12142 (2020)
27. Pollina, T., Larson, A.G., Lombard, F., Li, H., Le Guen, D., Colin, S., de Vargas, C., Prakash, M.: Planktoscope: Affordable modular quantitative imaging platform for citizen oceanography. Frontiers in Marine Science **9** (2022)
28. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks (2016), https://arxiv.org/abs/1506.01497
29. Saad, A., Bergrum, S., Stahl, A.: An instance segmentation framework for in-situ plankton taxa assessment. In: Thirteenth International Conference on Machine Vision. vol. 11605, pp. 294–303 (2021)
30. Saad, A., Stahl, A., Våge, A., Davies, E., Nordam, T., Aberle, N., Ludvigsen, M., Johnsen, G., Sousa, J., Rajan, K.: Advancing ocean observation with an ai-driven mobile robotic explorer. Oceanography **33**(3), 50–59 (2020)
31. Skalski, P.: Make Sense. https://github.com/SkalskiP/make-sense/ (2019)
32. Sonmez, S.: Grbl - high performance, open-source, embedded cnc controller (2024)
33. Sosik, H.M., Olson, R.J.: Automated taxonomic classification of phytoplankton sampled with imaging-in-flow cytometry. Limnology and Oceanography: Methods **5**(6), 204–216 (2007)
34. Suthers, I.M., Rissik, D. (eds.): Plankton: a guide to their ecology and monitoring for water quality. CSIRO Publishing, Collingwood, Vic. (2008)
35. Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., Luo, P., Liu, W., Wang, X.: Bytetrack: Multi-object tracking by associating every detection box. In: Avidan, S., Brostow, G., Cissé, M., Farinella, G.M., Hassner, T. (eds.) Computer Vision – ECCV 2022. pp. 1–21. Springer Nature Switzerland, Cham (2022), https://github.com/ifzhang/ByteTrack
36. Zhao, Y., Lv, W., Xu, S., Wei, J., Wang, G., Dang, Q., Liu, Y., Chen, J.: Detrs beat yolos on real-time object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 16965–16974 (June 2024)
37. Zheng, H., Wang, R., Yu, Z., Wang, N., Gu, Z., Zheng, B.: Automatic plankton image classification combining multiple view features via multiple kernel learning. BMC bioinformatics **18**, 1–18 (2017)