A Closed-Loop Agentic Framework for Marine Litter Monitoring: Data Synthesis, Augmentation, and Performance-Guided Optimization

Alessio Pierluigi Placitelli[1,2], Vincenzo Mariano Scarrica[2][0009−0008−4640−2693], and Antonino Staiano[2][0000−0002−4708−5860]

[1] Department of Biology, University of Naples Federico II, Naples 80126, Italy
[2] Department of Sciences and Technology, University of Naples Parthenope, Naples 80143, Italy

Abstract. Data scarcity remains a major obstacle to robust computer vision for marine litter monitoring, especially for emerged litter detection and temporal tracking in UAV-based imagery. This paper introduces a closed-loop agentic framework that autonomously generates, annotates, and refines synthetic training data to improve downstream detection and tracking performance. The system is supervised by either a remote or local reasoning agent, which coordinates data generation, annotation, training, and evaluation within an iterative optimization loop.
The framework combines SDXL Inpainting for foreground object synthesis and LTX-2.3 for high-fidelity temporal sequence generation. Automated annotations are produced using FastSAM and ByteTrack, providing mask and tracking ground-truth labels suitable for supervised training. The synthesized data are used to train and evaluate YOLOv8 Nano models for marine litter detection and multi-object tracking. Performance metrics, including mAP and MOTA, are fed back to the agent to refine prompts, generation settings, and training configurations. Experimental results show that the proposed workflow improves performance, highlighting the potential of agentic data synthesis and augmentation for automated marine litter monitoring in data-scarce scenarios.

Keywords: Agentic workflows · Marine litter detection · Closed-loop optimization · Marine litter tracking · Synthetic data generation

1 Introduction

The continuous increase in plastic production and disposal has led to widespread litter accumulation in marine environments [19], with serious impacts on ecosystems, biodiversity, and human health [21,33]. Marine litter monitoring is therefore essential for understanding pollution dynamics, supporting mitigation strategies, and enabling clean-up operations [6,33]. Since traditional monitoring is labor-intensive, costly, and difficult to scale, autonomous and semi-autonomous platforms such as *Unoccupied Aerial Vehicles (UAVs)*, *Unoccupied Surface Vehicles (USVs)*, and *Autonomous Underwater Vehicles (AUVs)* are increasingly adopted [7,19]. However, robust vision algorithms remain a major bottleneck [19]: although supervised deep learning (DL) has shown promise for floating litter monitoring in rivers and canals [12], high-precision detection in unstructured maritime habitats is still challenged by environmental variability and domain shifts [37].

A key limitation of DL-based monitoring is the need for large volumes of accurately annotated data, especially for semantic segmentation and tracking tasks requiring pixel-level masks and temporally consistent object identities. Since manual annotation is difficult to scale, recent generative approaches such as *SeeDiff* [22] exploit text-to-image diffusion attention mechanisms to generate image–mask pairs without additional training or prompt tuning, using cross-attention seeding and multi-scale self-attention mask expansion. These capabilities are relevant for marine litter monitoring, where limited spatially and temporally annotated data constrain the quantification and tracking of individual macroplastic objects in coastal regions [7,29]. Still, data generation alone is insufficient: synthetic samples must be scenario-relevant, automatically annotated, and evaluated through their impact on downstream detection and tracking performance.

Current resources remain fragmented, with a lack of comprehensive annotated datasets for marine litter detection and tracking [20]. Datasets for submerged debris, such as *SeaClear* [37] and *AquaTrash* [21], and surface-level USV imagery, such as *FML* [17], provide valuable benchmarks. Nevertheless, *nadir-view (top-down)* UAV datasets remain scarce, and existing open-access datasets are rarely annotated for temporal tracking tasks.

To address these limitations, this paper presents a *Closed-Loop Agentic Framework* for automated marine litter monitoring through data synthesis and augmentation. The framework targets UAV-based nadir-view scenarios and integrates synthetic data generation, automated annotation, and downstream evaluation in an iterative workflow. Detection and tracking metrics are fed back to an agentic orchestrator, which adapts generation prompts, synthesis parameters, and training configurations according to their impact on model performance.

1.1 Contributions

Building on this framework, our main contributions are:

- *Closed-Loop Agentic Orchestration for Monitoring:* We implement a dual-paradigm agentic system [1] in which either a remote controller or a local reasoning agent supervises the full workflow, including data generation, annotation, training, and evaluation. Following the principles of *DAAO* [25] and *AgentForge* [15], the agents use execution feedback from the downstream model to refine generation prompts, synthesis parameters, and training configurations.
- *Autonomous Synthesis and Augmentation of Nadir-View Data:* Addressing the lack of UAV-specific datasets for emerged marine litter monitoring [7], our framework utilizes *Stable Diffusion XL* for foreground inpainting and

LTX-2.3 for latent video generation. This enables the creation of high-fidelity, nadir-view synthetic images and video sequences that simulate diverse maritime conditions.
- *Integrated Detection and Tracking Annotations:* Unlike many existing datasets that focus mainly on static detection scenarios, such as *FML* [17] and *SeaClear* [37], our framework automates the generation of both *instance segmentation masks* via FastSAM and *temporal tracking IDs* via ByteTrack. This enables the construction of training data suitable for both object detection and dynamic marine litter tracking.
- *Performance-Guided Optimization via Feedback:* We demonstrate that the proposed closed-loop approach can improve the performance of a lightweight downstream model, namely *YOLOv8 Nano*, across both detection and tracking tasks. By treating model performance metrics, including mAP and MOTA, as feedback signals [25], the agents iteratively synthesize and retain targeted examples that contribute to better generalization.

2 Related Works

2.1 Marine Litter Detection and Dataset Limitations

Deep learning techniques for marine debris have developed rapidly, with the *YOLO family* consistently outperforming other object detectors in terms of the speed-accuracy trade-off [20]. Specialized models like *MLDet* [19] and compressed versions of YOLOv5 for embedded devices [33] have pushed the boundaries of real-time detection. Other approaches have explored *Transformers* for capturing global context [6] or *Multi-spectral sensors* on drones to classify specific litter materials such as fiber, fragments, and foam [11].

Despite these advances, the data-hungry nature of supervised DL remains a hurdle [12]. Existing resources like the *SeaClear Marine Debris Dataset* [37], *AquaTrash* [21], and the *Floating Marine Litter (FML)* dataset [17] provide vital benchmarks but often struggle with *domain shift* where a model trained on one site fails to generalize to unseen cameras, different altitudes, or varying water conditions [37]. Simple data augmentation (DA) like image flipping can boost accuracy [12] [27], but it is often insufficient for the complexity of less-defined targets in sea-surface imagery, where self-supervised strategies sometimes offer a more robust solution than standard supervised detection [4].

2.2 Generative Augmentation and UAV Optimization

To enhance datasets, studies have employed background removal and basic augmentation to optimize detection for specific items like bottles (BMD) [27]. However, these methods remain static and lack environmental diversity. Recent aerial surveys suggest that a resolution of approximately 0.5 cm/pixel is ideal for detection [27], yet maintaining this across diverse altitudes and atmospheric conditions requires sophisticated, synthetic data generation methods that can simulate these variations realistically.

2.3 The Rise of Agentic AI

A transformative shift is occurring with the move from single-model prompting to *Agentic AI* [1] [3]. Unlike traditional systems, agentic workflows use *Large Language Models (LLMs)* to reason, plan, and execute multi-step tasks autonomously [34] [3]. Frameworks like *AFlow* [34] and *DAAO* (Difficulty-Aware Agentic Orchestration) [25] demonstrate that agentic systems can optimize their own workflows based on execution feedback and query difficulty. In software engineering, systems like *AgentForge* [15] use execution-grounded verification in sandboxed environments to ensure code correctness, while systems like *Helium* [30] optimize the efficiency of these multi-step LLM calls through proactive caching.

Currently, these agentic workflows are predominantly used in healthcare, finance, or general software development [1] [15]. Their application in *automated data engineering* for environmental monitoring remains an untapped frontier. Our work bridges this gap by applying agentic orchestration to the Curate-Train-Refine cycle of marine litter detection. Notably, the work most closely aligned with our objective is Meta-Harness [18], an outer-loop system designed for automated harness engineering. Unlike static optimizers, Meta-Harness utilizes an agentic proposer that analyzes execution traces and source code to iteratively refine the application's logic. This approach demonstrates that providing agents with rich access to prior experience—rather than just compressed feedback—significantly improves performance across coding and reasoning tasks. In our framework, we extend this philosophy to the environmental domain, leveraging execution feedback from the downstream YOLOv8 model to autonomously evolve the data synthesis harness.

3 Methodology

The proposed framework, implemented with LangGraph [16], operates as a closed-loop system entirely supervised by an agentic structure. Given an initial dataset and an initial configuration (e.g., generation prompt, metric to optimize), the pipeline iterates over four phases (see Fig. 1): *dataset augmentation* (optional), *downstream task* actions (training and evaluation for detection or tracking), *reason*, and *act*. The *train* split seeds and trains the initial model, while the *validation* split is never augmented and serves as a stable benchmark across iterations. In the augmentation phase, stable diffusion inpainting and latent video generation produce high-fidelity synthetic sequences, which an automated annotation module converts into consistent masks and temporal tracklets via foundational vision models. During *reason*, an agent inspects task performance, analyzes the current configuration, and decides whether to trigger new data generation or hyper-parameter tuning; during *act*, the agent's output is materialized into a tweaked configuration that feeds the next iteration.

The loop is preceded by a warm-up stage: the downstream task is first executed on the original dataset to establish a baseline, then re-executed on the

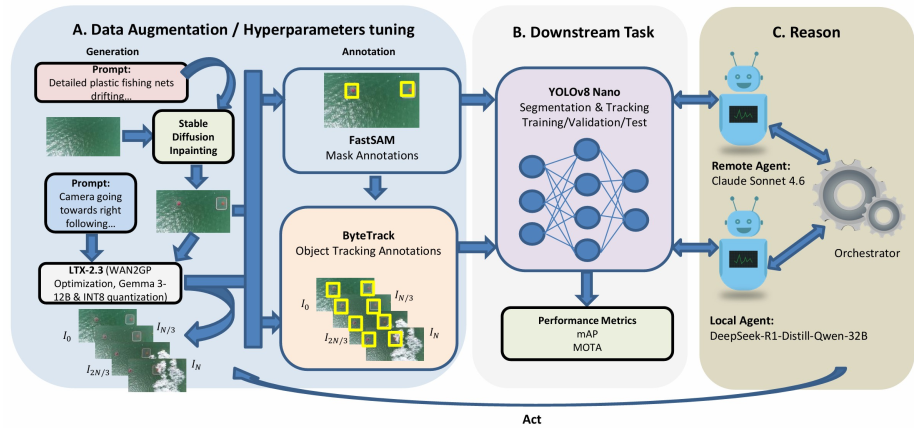

Fig. 1: Closed-loop Agentic Workflow for Marine Litter Dataset Augmentation. The processing stages: (A) (*optional*) Dataset augmentation / Hyper-parameters tuning: 1) Generation: SDXL Inpainting used for foreground synthesis and LTX-2.3 for video generation based on agent-optimized prompts; 2) Annotation: FastSAM and ByteTrack provide automated mask and tracking ground-truth labels; (B) Downstream Task, where a YOLOv8 Nano model is trained on the synthetic data. Performance metrics (mAP, MOTA) are fed back to the orchestrator as an optimization signal to iteratively refine the pipeline; (C) Reason, where a remote or local agent coordinates pipeline execution. The **Act** (Action) arrow links to the (A) stage in the loop.

union of original and synthesized data under the initial configuration, producing the starting point for agentic refinement.

3.1 Dataset augmentation

In this optional phase, new synthetic video or image data is generated, depending on the downstream task, based on the currently active configuration. Data annotations are produced alongside the new data. If the performances of the downstream task on the merged synthetic and original datasets improve for the designated metric, the generated data and its configuration are retained for the next iterations.

Data generation Our pipeline generates new data starting from user-configured static backgrounds and natural language prompts. To generate a marine litter dataset, a static image of the sea is used for all downstream tasks, taken in daylight with no clouds, from a nadir-viewing camera (i.e., UAV acquisitions).

To improve dataset diversity and task relevance, prompts are refined through an agentic pipeline that adapts textual descriptions according to scene context

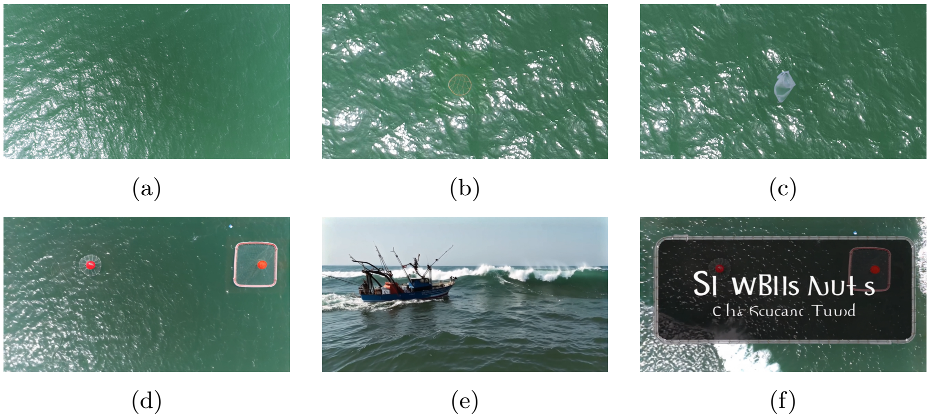

Fig. 2: (a) Background to condition models; (b),(c) cropped generated litter; (d) example good generated mot frame; (e),(f) generated visual artefacts.

(e.g., object type, scale, and water turbidity). This process is used exclusively for dataset construction and is not involved in downstream inference.

Image Generation and Inpainting: Stable Diffusion XL Inpainting 0.1 To generate high-quality synthetic training data for emerged marine litter scenarios, we employ *Stable Diffusion XL Inpainting 0.1* [24], a latent diffusion model specifically designed for conditional image completion and region-aware synthesis. The model operates in the latent space $\mathbf{z} \in \mathbb{R}^{h \times w \times c}$ of a pretrained Variational Autoencoder (VAE), where corrupted regions defined by a binary mask $\mathbf{m} \in \{0,1\}^{h \times w}$ are iteratively denoised conditioned on both the visible context and a textual prompt P.

In our framework, SDXL inpainting is used to synthesize *foreground marine litter instances* conditioned on real sea background imagery (Fig. 2), enabling controlled composition of realistic training samples. The model ensures spatial consistency between generated objects and the surrounding aquatic environment by leveraging both semantic conditioning and structural guidance from the unmasked regions.

Video Generation: LTX-2.3 22B Distilled To generate high-fidelity visual sequences from our agentically-refined prompts, we employ *LTX-2.3 22B Distilled* [10], a latent diffusion transformer (DiT) optimized for high-throughput generation. Our framework leverages the high-capacity 14B-parameter video stream of the 22B ensemble, operating on spatio-temporal latent tokens $\mathbf{z} \in \mathbb{R}^{H \times W \times T \times C}$ compressed via a causal VAE.

The model follows a Rectified Flow formulation, where the training objective minimizes the Mean Squared Error (MSE) of the predicted velocity field v_θ:

$$\mathcal{L}_{flow} = \mathbb{E}_{z_0, z_1, t}\left[\|v_\theta(z_t, t, c) - (z_1 - z_0)\|^2\right] \tag{1}$$

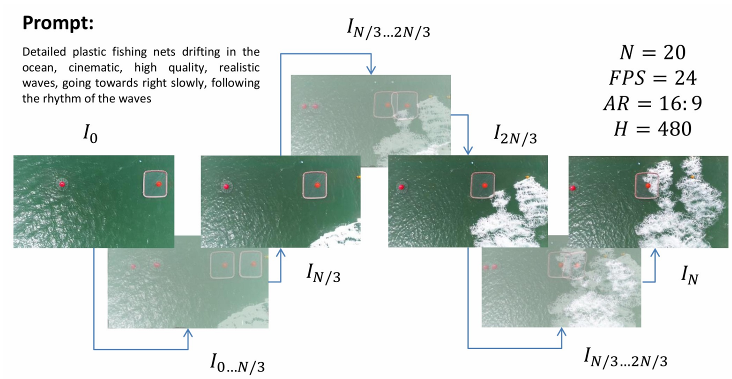

Fig. 3: An example of emerged marine litter video sequence generated by our workflow: I_0 is the frame generated by the stable diffuser generator, $I_{0\ldots N/3}$ represents the set of transition frames generated in between the first and the frame $I_{N/3}$, and similarly for $I_{N/3\ldots 2N/3}$ for $I_{N/3}$ and $I_{2N/3}$ and finally for $I_{2N/3\ldots N}$ for $I_{2N/3}$ and I_N. All of these frames have been generated by the LTX-2.3 model. The video generation prompt is shown in the top left: *"Detailed plastic fishing nets drifting in the ocean, cinematic, high quality, realistic waves, going towards right slowly, following the rhythm of the waves"*. Top right the parameters setting: N (Number of frames), FPS (Frames per second), AR (Aspect ratio), and H (Height in pixels).

where z_0 is the data latent, $z_1 \sim \mathcal{N}(0, I)$ is Gaussian noise, $t \in [0, 1]$, and c represents the conditioning context (text, initial image, and temporal parameters).

To fit within a 15GB VRAM envelope, the model is quantized to *INT8* and executed via the *Wan2GP* engine [31]. We adopt a two-stage refinement strategy: an initial base generation at half-resolution ($N = 8$ steps) followed by a latent upscaling and refinement stage ($N = 3$ steps). The final video V is recovered via:

$$V = \mathrm{VAE}_{dec}(\Phi_{refine}(\Phi_{base}(P, I_{start}, \tau, f), N_{refine})) \tag{2}$$

where P is the agent-generated prompt, I_{start} is the conditioning start image, τ is the duration in seconds, and f is the frame rate. This conditioning is further enhanced by *thinking tokens* within the Gemma 3 12B-based text encoder, ensuring high adherence to complex semantic instructions [26]. Since we do not need any extra information from the generated audio signal, this one is suppressed in a post-processing phase. The text prompt can affect the motion of litter over the sea surface and the dynamic of the nadir-viewing camera, therefore, it needs to be optimized by the agent before the generative step. A visual example is shown in Fig. 3.

Data annotation In the second part, we provide an automatic annotation module involving state-of-the-art models for generating coherent ground-truth masks and bounding boxes. This stage is crucial for ensuring a high-quality training/validation step at the end of the workflow. In general, this module provides annotations to be used even externally from our workflow, therefore proposing new potential datasets in the marine litter domain.

Object Segmentation: Fast Segment Anything Model (FastSAM) To ensure precise spatial grounding, we integrate the Fast Segment Anything Model (FastSAM), a foundation model for promptable image segmentation [36,14]. FastSAM decouples a heavy image encoder from a lightweight mask decoder, enabling real-time mask generation from sparse prompts (e.g., points or bounding boxes). To address prompt ambiguity, the model predicts multiple hierarchical masks simultaneously, ensuring a valid segmentation regardless of the prompt's precision, and inherits robustness from large-scale pre-training on SA-1B with a focal-plus-dice mask objective. In our pipeline, FastSAM isolates semantic entities from the conditioning image I_{start}, providing the spatial references used by the agentic workflow to guide LTX-2.3 with high semantic fidelity. Applying a confidence threshold to FastSAM predictions on each frame of the generated video yields the per-frame ground-truth annotations for the floating litter.

Object Tracking: ByteTrack To maintain temporal consistency and object identity across the generated sequences, we integrate ByteTrack, a robust multi-object tracking (MOT) framework [35]. Unlike standard association methods that filter out low-confidence detections, ByteTrack employs the *BYTE* data association strategy, which utilizes both high-score and low-score detection boxes to recover objects under partial occlusion or motion blur.

The tracking process utilizes a Kalman Filter to predict object states and performs a dual-stage association based on Intersection over Union (IoU) similarity. Let $\mathcal{T}$ denote the set of active tracklets and $\mathcal{D}$ the set of detections derived from SAM-generated masks. The association is formulated as:

$$\text{Match}(\mathcal{T}, \mathcal{D}_{high}) \cup \text{Match}(\mathcal{T}_{remain}, \mathcal{D}_{low}) \tag{3}$$

By matching unmatched tracklets $\mathcal{T}_{remain}$ with the low-score set $\mathcal{D}_{low}$, the system maintains trajectory continuity even when visual quality degrades. In our workflow, this mechanism is utilized to automatically annotate tracking ground truth, linking the semantic descriptions from our agentic workflow to consistent object IDs throughout the synthetic video duration.

Dataset merge In the last part, the final generated samples are subsequently integrated into a unified dataset for training detection and segmentation models on emerged marine litter scenarios. The unified dataset includes the original dataset, any previously generated dataset that was beneficial to the training, and the newly generated dataset.

3.2 Downstream task

This stage of our workflow involves training and validating a downstream model using the unified dataset to ensure robust performance on real-world distributions. The tasks involved are litter detection and tracking over frame sequences. The model is always validated on the original validation set, which is never augmented, for a fairer comparison with the original baseline. The implementation of the models is provided by the Ultralytics library [13] and Huggingface Transformer[32] and Diffusers libraries [23].

YOLOv8 Nano All segmentation, detection, and tracking tasks are realised by training a lightweight YOLOv8 Nano detector for real-time applications [28]. We leverage the spatio-temporal labels generated by the FastSAM-ByteTrack annotation loop to construct a specialized training set. For each synthetic frame, object masks are converted into normalized bounding box coordinates $\mathbf{b} = [x_c, y_c, w, h]$.

The $YOLOv8n$ model is optimized using a composite loss function:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{box} + \lambda_2 \mathcal{L}_{cls} + \lambda_3 \mathcal{L}_{dfl} \tag{4}$$

integrating Complete-IoU (CIoU) for localization and Binary Cross-Entropy for classification. Post-training, we evaluate the model's tracking capabilities using the **BYTE** association strategy. The performance is quantified via the Multi-Object Tracking Accuracy (MOTA):

$$\text{MOTA} = 1 - \frac{\sum_t (FN_t + FP_t + IDSW_t)}{\sum_t GT_t} \tag{5}$$

where FN, FP, and $IDSW$ represent false negatives, false positives, and identity switches at frame t, respectively. This evaluation validates the quality of the synthetic data distribution, proving that the distilled knowledge from high-parameter foundation models can effectively train efficient edge-ready architectures.

3.3 Reason: agentic workflow supervision

The proposed framework is driven by a Large Language Model (LLM) based agent that orchestrates the full pipeline, including data generation, annotation, and the training and evaluation of downstream tasks. This agent operates over structured representations of the system state, enabling coordinated decision-making across heterogeneous modules. Given a user-defined objective (e.g., target performance metrics), the agent iteratively analyzes intermediate results and triggers appropriate actions, effectively implementing a closed-loop optimization process over the pipeline configuration. The framework supports both remote agents and local agents, enabling fully local optimization loops. In order to mitigate local hardware memory constraints, local LLMs are loaded in memory exclusively for this phase, then unloaded.

Remote Agent We employ *Claude Sonnet 4.6* [2] as a remote agent responsible for high-level orchestration and strategic decision-making. The model operates autoregressively over token sequences, modelling the conditional distribution $p_\theta(x_t \mid x_{<t}, c)$, where c encodes the global system context, including dataset characteristics, model configurations, metric, and decision histories and evaluation metrics. Acting as a centralized controller, Claude Sonnet 4.6 coordinates the execution of the full workflow by initiating generation, annotation, and training phases, and by adapting the pipeline based on observed performance signals. Its remote deployment enables scalable interaction with the system through an abstract interface, decoupling high-level reasoning from local computational constraints and allowing flexible integration across different environments.

Local Agent As an alternative to the remote controller, we employ *DeepSeek-R1-Distill-Qwen-32B* [9] as a local agent for fully local reasoning and fine-grained control. The LLM is hosted locally and served via *llama.cpp HTTP Server* [8]. Built on the Qwen2.5-32B architecture, this model is distilled from a teacher optimized via Group Relative Policy Optimization (GRPO), a reinforcement learning scheme that replaces the actor-critic value estimator with a group-normalized advantage computed over multiple sampled outputs per query, combined with a KL penalty against a reference policy to preserve foundational capabilities. The resulting model exhibits strong chain-of-thought behaviour: reasoning trajectories are encapsulated within dedicated tags and shaped by a reward signal combining deterministic accuracy checks and structural format constraints. We exploit these emergent reasoning capabilities to perform rapid iteration over prompt adjustments and intermediate validation checks, trading the broader context of a remote model for the privacy and latency benefits of fully local operation.

3.4 Act: record metrics and execute

The final phase involves updating the active configuration by merging the agent-derived changes, checking if the target metric achieved the desired objective, and either terminating the loop or executing the planned decision.

4 Results

We evaluate the proposed framework on two downstream tasks: litter detection and multi-object tracking (MOT). The evaluation was performed on a machine running Ubuntu 24.04, equipped with an AMD Ryzen 9 3950X processor, 128GB of DDR4 RAM and an NVIDIA GeForce RTX 4090 GPU with 24 GB of VRAM. The system supported accelerated computation via CUDA version 12.2.

The *marine_litter12* dataset [5] was used as the original dataset for the detection task (685 images for the train set, 85 for validation). Images in this dataset have a variable resolution ranging from 512×341 up to 5280×3956 pixels. An internal video dataset was used for the MOT task (6 videos for the train set, 3 for the validation set). All videos have a resolution of 832×480 pixels at 24 frames-per-second.

The evaluation ran for 20 and 10 closed-loop iterations for the detection and MOT tasks, respectively (see Fig. 4), with a lower maximum number of iterations for MOT due to the higher time required to synthesize data at each iteration. As discussed in Section 3, the initial epoch in each experiment serves as a reference baseline, trained exclusively on real data with no synthetic augmentation.

Detection. The baseline model achieves a mAP50-95 of 0.469. Over 20 closed-loop iterations, the framework iteratively refines both the synthetic data generation parameters and the YOLOv8 loss weights, reaching a peak mAP50-95 of 0.488 in run 016, a gain of +2.0 percentage points (+4.3% relative). The proportionally larger gain on mAP50-95 versus mAP50 ($0.902 \rightarrow 0.914$, +1.2 pp) suggests that synthetic diversity (i.e., object size, placement, and appearance) improves the model's generalization across scales.

Multi-Object Tracking. The baseline achieves a MOTA of 0.759 with 9 identity switches. The framework improves peak MOTA to 0.778 in run 004 (+2.5% relative), while reducing identity switches from 9 to 6 at the same configuration. Detection precision remains consistently high throughout (> 0.97), confirming that the LTX-2.3-generated sequences annotated via FastSAM and ByteTrack provide a reliable training signal.

Discussion. Across both tasks, the closed-loop feedback loop yields meaningful but modest improvement over the real-data-only baseline on the target metrics. For detection, gains concentrate on mAP50-95, reflecting improved localization under strict IoU thresholds. For MOT, the optimization finds configurations that reduce identity confusion in the tracking phase, suggesting that prompt-level control over object motion dynamics in synthetic video is a meaningful lever for tracking performance.

5 Limitations

Throughout the experiments, the proposed framework showcased the following limitations:

- *Visual artefacts.* The quality of the prompts and the values of the generative model settings significantly affect the quality of the generated data: imprecise prompts can lead to deformed objects or unnatural object morphing in videos.
- *Prompt conceptual drifting.* As the agent iterates on the prompt template used to generate new synthetic data, it occasionally removes important semantic information (e.g., "a single object") or adds undesirable details (e.g., "prominently featured among other marine debris"). While this is mitigated in the system prompt, it occasionally happens, especially with local models.

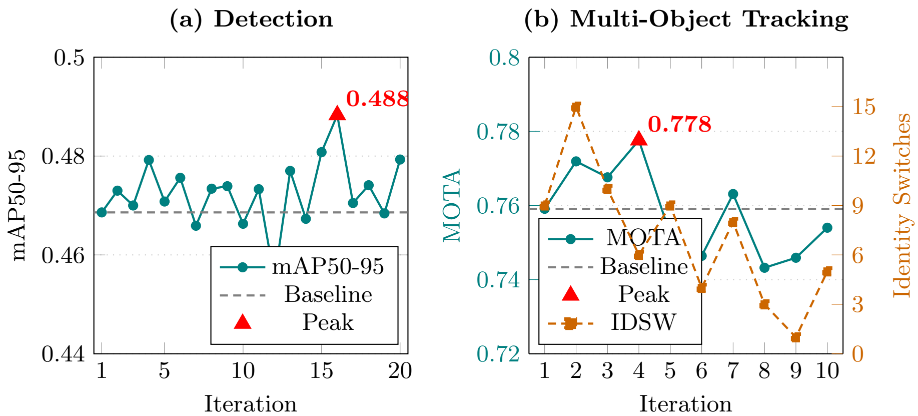

Fig. 4: Optimization trajectory across closed-loop iterations. **(a)** Detection: mAP50-95 over 20 runs. **(b)** MOT: MOTA and identity switches (IDSW) over 10 runs. Gray dashed lines mark the real-data-only baseline.

- *False positive masks*. The automatic annotation process can sometimes generate masks on the background, hindering the training process. While this is mitigated by discarding the generated data if it is not beneficial to the target metric, it still can cause a wasted iteration or, worse, a leak in the unified dataset.
- *Hitting token limits*. Locally served LLMs occasionally hit the token limit after initial epochs.
- *Local memory bottleneck*. The size and the quality of locally served LLMs are limited by the amount of available GPU VRAM, limiting the adoption of more capable and bigger LLMs.

6 Conclusions

We presented a closed-loop agentic framework that autonomously generates, annotates, and optimizes synthetic datasets for the detection and tracking of emerged marine litter, a domain where annotated data remains critically scarce.

The framework uses a reasoning agent (local or remote) in a LangGraph-orchestrated pipeline that couples SDXL inpainting and LTX-2.3 video generation with automated annotation via FastSAM and ByteTrack, closing the loop through iterative feedback from a downstream task model. Experimental evaluation on both tasks validates the approach: while the target metric is not reached, the framework is able to improve the chosen metric meaningfully, with the agent able to tweak relevant tunable settings, including the generative prompt, to close the gap towards the set objective. Future work will focus on a rigorous comparison of the performances across different local and remote agents, additionally researching agent prompt effectiveness to further increase the potential target metric gains. Further experimentation and research are additionally required on methods to obtain automatic annotations less susceptible to false positives.

Beyond the numerical gains, this work establishes a reusable blueprint for data-scarce environmental monitoring domains: by treating model performance as an optimization signal fed back to generative agents, the framework continuously evolves its own training data without human intervention, offering a scalable path toward robust, deployment-ready vision systems for marine pollution monitoring.

References

1. Abou Ali, M., Dornaika, F., Charafeddine, J.: Agentic ai: a comprehensive survey of architectures, applications, and future directions. Artificial Intelligence Review **59**(1), 11 (2026). https://doi.org/10.1007/s10462-025-11422-4
2. Anthropic: Claude sonnet 4.6 system card. Tech. rep., Anthropic (February 2026), https://www.anthropic.com/claude-sonnet-4-6-system-card, accessed: 2026-04-24
3. Bandara, E., Gore, R., Foytik, P., Shetty, S., Mukkamala, R., Rahman, A., Liang, X., Bouk, S.H., Hass, A., Rajapakse, S., Keong, N.W., Zoysa, K.D., Withanage, A., Loganathan, N.: A practical guide for designing, developing, and deploying production-grade agentic ai workflows (2025), https://arxiv.org/abs/2512.08769
4. Bilousova, O., Krinitskiy, M., Pogojeva, M., Spirina, V., Krivoshlyk, P.: A comparison of self-supervised and supervised deep learning approaches in floating marine litter and other types of sea-surface anomalies detection. Remote Sensing **18**(2) (2026). https://doi.org/10.3390/rs18020241, https://www.mdpi.com/2072-4292/18/2/241
5. CBSproject: marine_litter12 dataset. https://universe.roboflow.com/cbsproject-yd8hs/marine_litter12 (mar 2025), https://universe.roboflow.com/cbsproject-yd8hs/marine_litter12, visited on 2026-04-24
6. Dang, L.M., Sagar, A.S., Bui, N.D., Nguyen, L.V., Nguyen, T.H.: Attention-guided marine debris detection with an enhanced transformer framework using drone imagery. Process Safety and Environmental Protection **197**, 107089 (2025). https://doi.org/https://doi.org/10.1016/j.psep.2025.107089, https://www.sciencedirect.com/science/article/pii/S0957582025003568
7. Escobar-Sáanchez, G., Markfort, G., Berghald, M., Almroth, B.C., Hassellöv, M.: Aerial and underwater drones for marine litter monitoring in shallow coastal waters: factors influencing item detection and cost-efficiency. Environmental Monitoring and Assessment **194**(11), 863 (2022). https://doi.org/10.1007/s10661-022-10519-5
8. ggml authors: llama.cpp (2025), https://github.com/ggml-org/llama.cpp, release b8278.
9. Guo, D., Yang, D., Zhang, H., Song, J., Wang, P., Zhu, Q., Xu, R., Zhang, R., Ma, S., Bi, X., et al.: Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. Nature **645**(8081), 633–638 (2025)
10. HaCohen, Y., Brazowski, B., Chiprut, N., Bitterman, Y., Kvochko, A., Berkowitz, A., Shalem, D., Lifschitz, D., Moshe, D., Porat, E., et al.: Ltx-2: Efficient joint audio-visual foundation model. arXiv preprint arXiv:2601.03233 (2026)

11. Jeong, Y., Shin, J., Lee, J.S., Baek, J.Y., Schläpfer, D., Kim, S.Y., Jeong, J.Y., Jo, Y.H.: A study on the monitoring of floating marine macro-litter using a multi-spectral sensor and classification based on deep learning. Remote Sensing **16**(23) (2024). https://doi.org/10.3390/rs16234347, https://www.mdpi.com/2072-4292/16/23/4347
12. Jia, T., Vallendar, A.J., de Vries, R., Kapelan, Z., Taormina, R.: Advancing deep learning-based detection of floating litter using a novel open dataset. Frontiers in Water **Volume 5 - 2023** (2023). https://doi.org/10.3389/frwa.2023.1298465, https://www.frontiersin.org/journals/water/articles/10.3389/frwa.2023.1298465
13. Jocher, G., Qiu, J., Chaurasia, A.: Ultralytics YOLO (Jan 2023), https://github.com/ultralytics/ultralytics
14. Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., et al.: Segment anything. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 4015–4026 (2023)
15. Kumar, R., Ali, W., Ahmed, J., Ali, N.I., Usman, S.: Agentforge: Execution-grounded multi-agent llm framework for autonomous software engineering (2026), https://arxiv.org/abs/2604.13120
16. LangChain AI: LangGraph: Build resilient language agents as graphs (Mar 2026), https://github.com/langchain-ai/langgraph, python package version 1.1.3, released March 19, 2026
17. Lazzerini, G., Ferreira, F., Ridolfi, A.: Fml - floating marine litter dataset (2025). https://doi.org/10.17882/106148
18. Lee, Y., Nair, R., Zhang, Q., Lee, K., Khattab, O., Finn, C.: Meta-harness: End-to-end optimization of model harnesses (2026), https://arxiv.org/abs/2603.28052
19. Ma, D., Wei, J., Li, Y., Zhao, F., Chen, X., Hu, Y., Yu, S., He, T., Jin, R., Li, Z., Liu, M.: Mldet: Towards efficient and accurate deep learning method for marine litter detection. Ocean & Coastal Management **243**, 106765 (2023). https://doi.org/https://doi.org/10.1016/j.ocecoaman.2023.106765, https://www.sciencedirect.com/science/article/pii/S0964569123002909
20. Moorton, Z., Kurt, Z., Woo, W.L.: State of the art applications of deep learning within tracking and detecting marine debris: A survey (2024), https://arxiv.org/abs/2403.18067
21. Panwar, H., Gupta, P., Siddiqui, M.K., Morales-Menendez, R., Bhardwaj, P., Sharma, S., Sarker, I.H.: Aquavision: Automating the detection of waste in water bodies using deep transfer learning. Case Studies in Chemical and Environmental Engineering **2**, 100026 (2020). https://doi.org/https://doi.org/10.1016/j.cscee.2020.100026, https://www.sciencedirect.com/science/article/pii/S2666016420300244
22. Park, J.H., Jo, K., Baik, S.: Seediff: Off-the-shelf seeded mask generation from diffusion models. Proceedings of the AAAI Conference on Artificial Intelligence **39**(6), 6406–6415 (Apr 2025). https://doi.org/10.1609/aaai.v39i6.32686, https://ojs.aaai.org/index.php/AAAI/article/view/32686
23. von Platen, P., Patil, S., Lozhkov, A., Cuenca, P., Lambert, N., Rasul, K., Davaadorj, M., Nair, D., Paul, S., Berman, W., Xu, Y., Liu, S., Wolf, T.: Diffusers: State-of-the-art diffusion models. https://github.com/huggingface/diffusers (2022)
24. Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 10684–10695 (2022)

25. Su, J., Lan, Q., Xia, Y., Sun, L., Tian, W., Shi, T., He, L.: Difficulty-aware agentic orchestration for query-specific multi-agent workflows. In: Proceedings of the ACM Web Conference 2026 (WWW '26). pp. 2060–2070. Association for Computing Machinery, New York, NY, USA (2026). https://doi.org/10.1145/3774904.3792240
26. Team, G., Kamath, A., Ferret, J., Pathak, S., Vieillard, N., Merhej, R., Perrin, S., Matejovicova, T., Ramé, A., Rivière, M., et al.: Gemma 3 technical report. arxiv 2025. arXiv preprint arXiv:2503.19786 (2025)
27. Tran, T.L.C., Huang, Z.C., Tseng, K.H., Chou, P.H.: Detection of bottle marine debris using unmanned aerial vehicles and machine learning techniques. Drones **6**(12) (2022). https://doi.org/10.3390/drones6120401, https://www.mdpi.com/2504-446X/6/12/401
28. Varghese, R., Sambath, M.: Yolov8: A novel object detection algorithm with enhanced performance and robustness. In: 2024 International conference on advances in data engineering and intelligent computing systems (ADICS). pp. 1–6. IEEE (2024)
29. de Vries, R., Egger, M., Mani, T., Lebreton, L.: Quantifying floating plastic debris at sea using vessel-based optical data and artificial intelligence. Remote Sensing **13**(17) (2021). https://doi.org/10.3390/rs13173401, https://www.mdpi.com/2072-4292/13/17/3401
30. Wadlom, N., Shen, J., Lu, Y.: Efficient llm serving for agentic workflows: A data systems perspective (2026), https://arxiv.org/abs/2603.16104
31. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)
32. Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T.L., Gugger, S., Drame, M., Lhoest, Q., Rush, A.M.: Transformers: State-of-the-art natural language processing. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations. pp. 38–45. Association for Computational Linguistics, Online (Oct 2020), https://www.aclweb.org/anthology/2020.emnlp-demos.6
33. Yang, X., Chen, Y., Zhou, Y., Tong, F.: A three-dimensional marine plastic litter real-time detection embedded system based on deep learning. Marine Pollution Bulletin **213**, 117603 (2025). https://doi.org/https://doi.org/10.1016/j.marpolbul.2025.117603, https://www.sciencedirect.com/science/article/pii/S0025326X25000785
34. Zhang, J., Xiang, J., Yu, Z., Teng, F., Chen, X., Chen, J., Zhuge, M., Cheng, X., Hong, S., Wang, J., Zheng, B., Liu, B., Luo, Y., Wu, C.: Aflow: Automating agentic workflow generation (2025), https://arxiv.org/abs/2410.10762
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: European conference on computer vision. pp. 1–21. Springer (2022)
36. Zhao, X., Ding, W., An, Y., Du, Y., Yu, T., Li, M., Tang, M., Wang, J.: Fast segment anything (2023)
37. Đuraš, A., Wolf, B.J., Ilioudi, A., Bibuli, M., Caccia, M., Cafolla, D., Gancino, G., Gospić, D., Ivanković, D., Kalafatić, Z., et al.: A dataset for detection and segmentation of underwater marine debris in shallow waters. Scientific Data **11**(1), 921 (2024). https://doi.org/10.1038/s41597-024-03759-2