

Ocean Voyager

Testing Progression and Results

Master validation record: software simulation → embedded/HIL → physical propulsion → powered pool autonomy • Updated through September 2026

Purpose and continuity

This master document preserves the original testing progression from the first simulator through synchronized manual-HIL replay, then appends the subsequent in-house refinements and powered-pool campaign. The organizing principle remains the same: test components were progressively replaced by real hardware, real sensing, physical actuation and less idealized assumptions. Later stages did not simply add hardware; they repeatedly changed the controller when physical evidence invalidated an earlier assumption.

Stages 1–9 below preserve the substantive wording and quantitative results of the earlier progression document. Stages 10–23 consolidate the later manual-HIL refinements and pool-testing record. Detailed pool experiments and detailed software algorithms remain in their dedicated companion documents; this master version emphasizes what was tested, what changed, what was observed, and what each stage motivated.

At-a-glance progression

Phase	What became real	Primary validation question
Simulation / SIL (1–5)	Mission logic, scenario complexity, trained RGB model	Does the decision architecture work under controlled and stochastic conditions?
Embedded + stationary HIL (6–7)	Raspberry Pi execution; MLX90640 + RGB camera	Does the same logic run on target hardware and preserve thermal→RGB verification?
Manual-motion mission HIL (8–10)	Physical robot geometry and changing real viewpoints	What breaks when movement, duplicates, lane displacement and boundary error become physical?
Powered platform bring-up (11–13)	Pumps, driver, voltage, hull/buoyancy	Can the boat move repeatably enough to support autonomy?
Powered navigation experiments (14–19)	Real-water dynamics, wall interaction, continuous vision	Which navigation assumptions survive unequal thrust, drift, coasting and wall contact?
Integrated sensing + autonomy (20–22)	Opportunistic target pursuit with active RGB+thermal fusion	Can sensing interrupt exploration briefly and safely, then return control to navigation?
Post-run reconstruction (23)	Action/sensor evidence mapped into an estimated route	Can the physical run be audited and re-rendered without claiming unavailable ground truth?

Chronological testing stages

1. Core mission logic + simulation harness

Test / implementation. Core logic: coverage/navigation, thermal-candidate handling, approach, RGB verification/voting, confirm/reject, return home, and logging. Test harness: simulated pool, motion, targets, thermal/RGB observations, range/FOV/contrast, and non-target conditions.

Results / learning. Established the end-to-end sequence: coverage → thermal scan → candidate → approach → RGB verification → decision → resume/return home. This controller became the software base for later stages.

2. Controller robustness testing

Test / implementation. The Stage-1 controller was retained. The test harness repeatedly generated uncertain positive/negative verification observations so the same controller could be exercised under stochastic sensing outcomes.

Results / learning. 46 full-controller trials produced 35 DONE and 11 ABORT (DONE rate 0.76). Successful verification outcomes were 15 at 3/5 positive, 15 at 4/5, and 5 at 5/5. All 11 ABORT runs reached VERIFY but failed the required 3/5 vote.

3. Verification-rule stress testing

Test / implementation. The five-observation voting component was isolated for large-scale testing. Test logic varied the vote threshold and generated correlated observation errors; the rest of the mission did not need to be rerun for each statistical trial.

Results / learning. With 1,000 Monte Carlo trials per threshold and per-observation positive probability 0.60, DONE rates were 0.911 (2/5), 0.675 (3/5), and 0.342 (4/5). A correlated-error sweep used 10,000 trials at each of six p values from 0.0–0.5 (60,000 total); under 3/5 voting, DONE rate decreased from ~0.680 to ~0.642.

4. Integrated scenario testing

Test / implementation. The same core controller was returned to the full simulation harness. Scenarios varied target number/location, delayed detection, thermal range/contrast, seaweed/non-target candidates, coverage, boundaries, and return home.

Results / learning. All 8 missions in one expanded scenario set returned home. Representative 3-bottle surveys confirmed all 3 targets with 100% thermal-scan coverage. Time-boxed 3-bottle and 4-bottle-plus-seaweed patrols each reached 98.91% unique path coverage and 100% thermal-scan coverage; the latter confirmed 4/4 bottles and rejected the seaweed candidate.

5. Trained RGB model evaluation

Test / implementation. The simulated/mock visual-verification source was replaced by the trained YOLOv8n verifier, first evaluated independently on deployment-oriented images.

Results / learning. On a 145-image deployment-oriented split containing bottle positives and marine hard negatives, precision was 0.947, recall 0.930, mAP@0.5 0.960, and mAP@0.5:0.95 0.831.

6. Core software deployed on Raspberry Pi

Test / implementation. The Stages 1–4 controller was moved to Raspberry Pi 4. Sensor adapters remained mocked/simulated and propulsion commands were logged rather than sent to motors.

Results / learning. The target-platform suite completed with 10 tests passing and 1 skipped. The Pi executed mission logic and generated forward, turn, stop, approach, and avoidance decisions without enabling propulsion.

7. Real thermal + RGB sensor HIL

Test / implementation. The core controller remained on the Pi. Mock sensor adapters were replaced by the physical MLX90640 and Raspberry Pi AI/RGB camera. Propulsion remained disabled; cold-water targets against a warmer floor/background provided controlled thermal contrast and were moved through different viewpoints.

Results / learning. Repeated real-sensor trials demonstrated the intended gating sequence: the MLX90640 thermal candidate occurred first and initiated RGB/YOLO verification. Each investigation logged thermal delta/bearing/preview, RGB frames, detections and the controller decision. The HIL logs therefore captured both positive waste confirmations and thermal candidates subsequently rejected by vision as non-waste.

8. Full mission HIL with manual robot motion

Test / implementation. Physical robot + Pi + real thermal/RGB sensors + trained detector + core controller operated together. Controller-requested translations/rotations were executed manually; propulsion remained disabled. Multiple waste and non-waste targets were encountered during the coverage run.

Results / learning. A representative 8×8-ft manual HIL run lasted 1,799.24 s and finished at the recorded home coordinate (0,0). The run logged 3 unique confirmed-waste objects and 1 rejected non-waste candidate. One rejection followed three RGB frames with no expected waste label. A later view of an already confirmed target was also suppressed as a duplicate rather than counted again. Real

movement exposed lane displacement and boundary error; the logged trajectory reached ~11.08 ft on one coordinate, motivating lane rejoin and the hard boundary guard.

9. Synchronized replay of physical HIL runs

Test / implementation. No new robot logic was introduced. The animation reconstructed the Stage-8 run from the mission log, trajectory, real MLX90640 previews, RGB/YOLO images and decisions.

Results / learning. The replay provided a frame-by-frame record of the same physical HIL results: thermal candidate first, followed by the corresponding RGB evidence and confirm/reject/duplicate/avoidance decision. Confirmed waste, rejected non-waste and collision-risk events were visually distinguished, while the map showed the physical/manual trajectory. The replay also exposed the out-of-bounds trajectory and sensor-display synchronization issues that were subsequently corrected.

10. Pre-pool robustness refinements from manual HIL

Test / implementation. Before powered autonomy, the manual-HIL findings were turned into explicit safeguards: collision-aware pre-move vision, lane rejoin and heading restoration after target investigation, duplicate-object suppression, explicit non-waste rejection, and a hard operating-boundary guard. The replay system was also refined so thermal evidence persisted until its corresponding RGB decision and target markers represented the actual candidate bearing/centroid.

Results / learning. This phase converted problems exposed by physical/manual movement into controller behavior rather than treating them as replay artifacts. It also clarified that navigation safety and debris classification could share RGB observations while remaining separate decisions.

11. Physical propulsion integration and fault isolation

Test / implementation. The platform moved from manual-motion HIL to powered water testing using two small 3–5 V submersible pumps through a DRV8833. Early pool/kitchen/bathtub checks deliberately isolated propulsion from autonomy and sensing. Left/right pulses, tubing configuration, wiring, common ground, and pump start/flow behavior were tested independently.

Results / learning. A loose common-ground connection was found and reseated, restoring both pumps in one failure episode. Intermittent audible-but-low/no-flow behavior also pointed to limited voltage margin and possible priming/air-pocket effects. Keeping propulsion and sensor troubleshooting separate prevented electrical, hydraulic and I²C faults from being conflated.

12. Voltage, straight-line and turning calibration

Test / implementation. The motor supply was moved from approximately 3 V (2×AA) to approximately 4.5 V (3×AA) while retaining the same basic pumps/driver. Repeated manual commands were used to characterize forward bias and turn authority rather than assuming symmetric propulsion.

Results / learning. At 4.5 V the platform produced much more dependable continuous motion. One early straight-line calibration used approximately 1.6 s left versus 1.0 s right; later control schemes used different timing but confirmed the same underlying asymmetry. Very short ~0.2–0.25 s steering pulses were often physically ineffective, especially at a wall. OFF was also shown to mean coasting rather than an instantaneous stop.

13. Hull, buoyancy and structural redesign

Test / implementation. Pool trials exposed mechanical sensitivity as part of the control problem. The earlier taller/double-decker arrangement was replaced by a lower single-deck electronics arrangement, with broad plastic plate/lid surfaces beneath and across the boat to increase flotation footprint and stability.

Results / learning. The lower, wider structure reduced sensitivity to roll, uneven loading and trapped-water imbalance. More stable pump immersion and hull attitude made subsequent propulsion/navigation behavior more repeatable. Mechanical configuration therefore became part of the experimental baseline rather than a separate packaging concern.

14. Deterministic pool navigation and localization assumptions

Test / implementation. Initial powered-navigation plans used square-boundary or lawnmower coverage, calibrated straight segments and nominal turns, with an approximate positive-coordinate pool map. This preserved the structured coverage concept used in simulation.

Results / learning. Real water amplified the weaknesses already seen in manual HIL: unequal thrust, drift, momentum, variable turns and uncertain pose. The project therefore separated real-time navigation from post-run plotting: the controller did not need a perfect continuous (x,y) estimate in order to explore, provided enough actions and sensor evidence were logged to reconstruct a plausible trajectory later.

15. Visual wall-distance calibration and CLOSE/OPEN experiments

Test / implementation. RGB navigation experiments tested whether the forward scene could be classified as OPEN versus CLOSE using center-view edge-density (E_C) and scale/distance-transform (S_C) metrics. Far, medium, ~1-ft, ~0.5-ft and hard-contact observations were collected.

Results / learning. The metric ranges overlapped substantially across ordinary viewpoints, so a single threshold was not a reliable distance estimator. A repeated hard-contact signature near $E_C \approx 0$ and $S_C \approx 65534.6$ was useful for recognizing that collision had already occurred, but not for choosing a pre-collision avoidance direction. Temporal evidence became more important than one-frame classification.

16. Navigation-controller evolution: continuous vision and recovery (v9–v23)

Test / implementation. Successive controllers explored latched turn direction, continuous visual navigation, latest-frame asynchronous camera capture, sustained turn-until-open behavior, bounded turning, and an explicit STUCK recovery concept. The goal was to keep sensing live while giving the motors enough time to create a physically meaningful maneuver.

Results / learning. The experiments showed that perception frequency and motor duration must be decoupled. Camera processing could remain near ~10 fps while a turn lasted seconds. Rapid LEFT/RIGHT alternation and tiny pulses often produced little useful rotation; sustained turning could also fail when the hull was physically pinned. Recovery sometimes required creating clearance before trying another turn.

17. Visual drift estimation and counter-steering experiments

Test / implementation. Optical-flow-based drift estimation and configurable counter-steering were tested as ways to compensate for the platform's repeatable curvature. Drift could be monitored or used for bounded adaptation, while timed counter-steer provided a simpler open-loop alternative.

Results / learning. These experiments were useful for characterizing bias but did not justify making accurate heading estimation a prerequisite for exploration. The key design separation was retained: drift correction could improve motion, while the mission policy itself should tolerate imperfect heading and position.

18. Strategic pivot to stochastic exploration

Test / implementation. The controller was simplified to choose among FORWARD, LEFT, RIGHT and PAUSE/OFF using configurable probabilities and duration ranges. A separate stochastic action bank was used after confirmed collision. Random seeds remained available for repeatable debugging.

Results / learning. A useful v45 normal bank favored FORWARD (0.70, 2–12 s) with LEFT 0.07 (1–6 s), RIGHT 0.17 (1–6 s), and PAUSE 0.06 (0.2–4 s). The collision bank used LEFT 0.15 (0.5–6 s), RIGHT 0.45 (1–6 s), and PAUSE 0.40 (0.5–3 s). The asymmetry was intentional and tuned against observed clockwise/right bias. Wider durations produced genuine movement/exploration instead of tiny static circles.

19. Collision handling as recovery rather than a perfect-avoidance requirement

Test / implementation. Earlier pool work tried to avoid every wall contact. The later controller accepted occasional contact for this prototype, detected the hard-wall condition, stopped the current behavior, and switched to the separately configurable collision-random bank until the view cleared.

Results / learning. This removed perfect pre-collision avoidance as a blocking dependency. It did not establish collision-free navigation; rather, it demonstrated that the mission could recover from wall situations and continue exploring while more sophisticated avoidance remained a future validation target.

20. Opportunistic object investigation and anti-loop safeguards

Test / implementation. Object handling was reframed as a short opportunistic diversion from exploration. The controller could approach a plausible object enough to investigate it and avoid driving into it when possible, but target behavior was bounded by a pursuit time limit. After any approach, a mandatory cooldown prevented immediate re-entry.

Results / learning. The cooldown protected against repeatedly seeing the same bottle or a software/detection loop monopolizing the mission. Exploration was therefore guaranteed to resume. Wall handling retained higher priority and could preempt a target pursuit.

21. Thermal/RGB pool fusion and controlled warm-bottle test

Test / implementation. RGB and MLX90640 sensing remained independently auditable. Because ordinary bottle/pool temperature contrast could be marginal for an end-to-end fusion demonstration, a bottle containing moderately warmer water was used as a controlled thermal target. The system required corroborated RGB and thermal evidence, including bearing agreement, before the strongest target behavior could take over.

Results / learning. The controlled contrast made it possible to test the complete real-sensor fusion path without claiming that arbitrary floating plastic will always be thermally separable in real water. Independent sensor logging also helped distinguish RGB-side problems from thermal-side problems during troubleshooting.

22. Integrated powered pool demonstration — Run 04

Test / implementation. With both vision and thermal modes active, v45 combined continuous stochastic exploration, hard-wall recovery, independent sensor processing, fused candidate generation, bounded target steering/approach, target cooldown, and return to exploration. Raw RGB frames and thermal data were archived throughout the run.

Results / learning. The run demonstrated repeated RGB and thermal observations, multiple fused events, target pursuit followed by cooldown suppression, collision recovery, and continued navigation. It served as a systems-integration demonstration rather than a proof of guaranteed coverage, centimeter-level localization, collision-free operation, or general thermal detectability.

23. Post-run trajectory reconstruction and synchronized sensor replay

Test / implementation. Because the powered boat did not have external ground-truth localization, plotting was treated as post-processing. Timestamped motor states were integrated with display calibrations (approximately 0.175 ft/s forward, 30°/s left and 45°/s right) and pool/wall constraints. Archived RGB, thermal frames, detections and fusion events were synchronized to the reconstructed route.

Results / learning. The resulting replay made the behind-the-scenes run inspectable and supported alternative future reconstruction methods because raw evidence had been retained. The displayed trajectory is explicitly an estimate, not measured ground truth; drift, momentum, asymmetric thrust and wall interaction are only approximated.

Cross-stage engineering lessons

- **Same mission intent, changing implementation.** The project retained the broad scan/investigate/verify/log mission intent, but real-world evidence forced navigation, recovery, timing, fusion and plotting mechanisms to evolve.
- **Replace assumptions one at a time.** Mock sensing, real sensing, manual motion, propulsion, voltage, wall logic and fusion were isolated where possible. This made failures diagnosable instead of attributing every problem to 'autonomy.'
- **Physical dynamics are part of the algorithm.** Unequal thrust, inertia, wall friction, hull stability and pump immersion materially changed which software strategies were viable.
- **Fresh perception matters more than processing every frame.** Latest-frame camera handling prevented a moving robot from making decisions from a backlog of stale images.
- **Perception rate and maneuver duration are independent.** Fast camera updates did not imply 0.1–0.2 s motor actions. The boat often needed seconds of sustained actuation to produce a meaningful physical turn.
- **Hard collision detection is not pre-collision avoidance.** A saturated wall image can reliably indicate contact while providing little directional information about how the robot should have avoided it.
- **Uncertain localization does not prevent useful exploration.** The eventual stochastic policy reduced dependence on brittle real-time coordinates, while rich logging preserved the ability to reconstruct and improve trajectories later.

- **Object investigation must not monopolize the mission.** Bounded pursuit and cooldown guarantee a return to exploration even if the same target is repeatedly visible or a detector misbehaves.
- **Independent sensor evidence is diagnostically valuable.** Thermal and RGB remained separately logged and interpretable before fusion, making sensor-side failures easier to identify.
- **Replay is an engineering instrument.** Synchronized trajectory, motor, RGB, thermal and decision evidence exposed timing, boundary and controller problems that were difficult to see from console logs alone.

Current demonstrated scope and limits

The current powered-pool result is best described as an integrated prototype demonstration. The platform has demonstrated physical stochastic exploration, recognition/recovery from hard wall situations, independently active RGB and thermal sensing, corroborated RGB–thermal fusion, bounded target pursuit, cooldown-based return to exploration, and synchronized post-run reconstruction. The evidence does not establish centimeter-accurate localization, guaranteed complete coverage, collision-free navigation, or general thermal detectability of arbitrary plastic debris. Those require separately controlled ground-truth experiments.

Companion technical records

Use the Physical Pool Experiments & Engineering Learnings document for the detailed pool chronology, controller-version experiments, mechanical evolution and Run 04 interpretation. Use the Software Architecture, Navigation Algorithms & Runtime Options document for thread structure, latest-frame acquisition, motor execution, thermal/RGB processing, fusion, stochastic action selection, target-control equations and command-line option meanings. This master document is intentionally the chronological validation spine connecting those deeper records.