Antflie: Frugal Visual Teach and Repeat on Narrow FoV Micro-Drones

Gabriel Gattaux, Hamidou Diallo, Julien R. Serres, Antoine Wystrach, and Franck Ruffier

***Abstract*—We present an insect-inspired visual teach-and-repeat framework demonstrated on Antflie, a 33-gram MAV equipped with an ultra-low-resolution camera (24 × 24 px) and a narrow 87° field of view (FoV). During a one-shot outbound teach flight, the MAV performs periodic scans and classifies views as left or right of the route using inertial and optic-flow cues. These lateralized views are stored as compact visual memories in a Mushroom Body network with a footprint under 4 kB. In the repeat phase, the MAV flies the inbound route by retracing the outbound path, and autonomously lands at its home location using only visual familiarity through direct sensorimotor coupling, rather than map-based reasoning. Offline simulations show that the Route Lateralized (R-Lat) algorithm in Antflie matches the accuracy of a state-of-the-art insect visual compass (V-Comp) while running up to 20× faster and supporting narrow FoVs. Real-world indoor experiments further demonstrate 24 autonomous inbound repeats totaling 110 meters of flight, with a 13-cm median lateral error and a mean landing error of 34 cm. These results highlight the feasibility of frugal, bio-inspired, vision-only navigation for MAVs operating under strict size, weight, power, and cost constraints, inspired by the navigation of *Cataglyphis* and *Melophorus* ants.**

***Index Terms*—Aerial systems: perception and autonomy, biologically-inspired robots, insect neural network, micro/nano robots, vision-based navigation.**

I. INTRODUCTION

MICRO Aerial Vehicles (MAVs) hold promise for tasks like subterranean exploration, ecological monitoring, or last-mile delivery, but their ultralight designs impose strict size, weight, power, and cost (SWaP-C) constraints [1], [2]. These limitations rule out heavy sensors such as LiDAR, making vision-based navigation more attractive due to its compactness and efficiency. In GPS-denied missions, homing (the ability to autonomously return to the starting point) is critical, especially when the outbound path offers the safest return route, as in Martian lava tubes exploration [3]. Visual Teach-and-Repeat (VT&R) algorithms can achieve this by recording landmarks during an outbound teach phase and using them to retrace the path during a repeat phase, forming a visual Ariadne's thread [4], [5].

However, typical VT&R pipelines rely on high-resolution imagery, stereo cameras, and compute-heavy algorithms such as Visual Odometry (VO), visual Simultaneous Localization and Mapping (vSLAM), or Visual Place Recognition (VPR) [6], [7], which are generally unsuitable for MAVs that must operate under severe SWaP-C.

In parallel, insects like ants and bees navigate long routes through complex environments using low-resolution vision and minimal neural resources [8]. They rely on egocentric visual familiarity, answering "Where am I going?" rather than "Where am I?" [9], by storing sparse associative view memories in the Mushroom Body (MB) insect brain area [10]. This has inspired biologically grounded VT&R approaches that avoid explicit mapping and localization [11].

In this letter, the lateralized route–centric framework tested on a ground robot with panoramic camera [12], here termed Route-Lateralized (R-Lat), is extended to operate with arbitrary camera FoVs without requiring calibration or panoramic vision. Experiments were performed with Antflie, a 33-gram, 9 cm MAV quadcopter, operating with a narrow FoV and ultra-low-resolution vision (87°, 24 × 24 px). This SWaP-C constrained, lightweight, and biologically inspired VT&R (see Fig. 1) is divided into two phases. During the outbound (teach) phase, Antflie performs periodic scans and uses the embedded Inertial Measurement Unit (IMU) and onboard optic flow processing (OF) to categorize views as "left" or "right" of the route, forming the local compass, encoding compact route-lateralized visual memories in a MB neural network, with a memory footprint under 4 kB. The local compass provides heading computation relative to the local route frame. During the inbound (repeat) phase, navigation is achieved purely through visual familiarity as a feedback signal, without maps, localization, or odometry.

The main contributions are: (1) Extension of R-Lat framework to arbitrary camera FoVs by using a local compass, removing the need for panoramic input; (2) Real-world MAV demonstration of familiarity-based VT&R with autonomous landing under extreme SWaP-C constraints, including a Lyapunov stability experimental proof; and (3) Benchmark against the Insect Visual

Received 9 September 2025; accepted 11 February 2026. Date of publication 24 February 2026; date of current version 28 February 2026. This article was recommended for publication by Associate Editor Alberto Candela and Editor Soon-Jo Chung upon evaluation of the reviewers' comments. The work of Gabriel Gattaux was supported in part by the Aix Marseille University and the French Ministry of Defense under Grant AID, A01D22020549 and in part by CNRS and AMU support. The work of Hamidou Diallo was supported in part by France 2030 via A∗MIDEX–ARKAIA under Grant AMX-19-IET-003 and in part by Aqueducopter. The work of Julien R. Serres and Franck Ruffier was supported in part by CNRS and in part by AMU support. This work was supported by French National Research Infrastructure ROBOTEX 2.0 under Grant ROBOTEX ANR-10-EQPX-44-01 and Grant TIRREX ANR-21-ESRE-0015. *(Corresponding author: Gabriel Gattaux.)*

Gabriel Gattaux is with CNRS, ISM, Aix Marseille Université, 13284 Marseille, France (e-mail: gabriel.gattaux@univ-amu.fr).

Hamidou Diallo is with INRIA, INSA Lyon, Université Lyon, 69621 Villeurbanne, France.

Julien R. Serres is with CNRS, ISM, Aix Marseille Université, 13284 Marseille, France, and also with the Institut Universitaire de France, 75231 Paris, France.

Antoine Wystrach is with Université Toulouse, 31400 Toulouse, France.

Franck Ruffier is with CNRS, ISM, Aix Marseille Université, 13284 Marseille, France, and also with CNRS Lab-STICC, ENSTA IP Paris, 91120 Brest, France.

Digital Object Identifier 10.1109/LRA.2026.3667486

2377-3766 © 2026 IEEE. All rights reserved, including rights for text and data mining, and training of artificial intelligence and similar technologies.
Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

Authorized licensed use limited to: ENSTA. Downloaded on June 19,2026 at 15:11:08 UTC from IEEE Xplore. Restrictions apply.

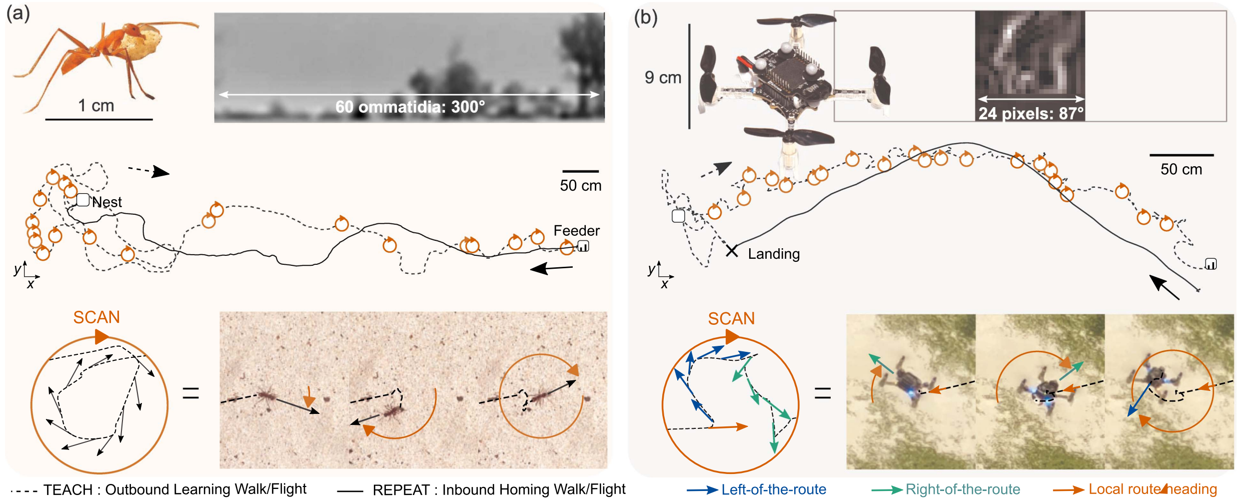

Fig. 1. Bio-inspired homing via outbound scans and route-lateralized visual memories, bird-eye views. (a) Desert ant navigation (adapted from [13], [14]). (b) Antflie quadcopter mimics this with low-res, narrow-FoV scans for learning and visual familiarity for homing/landing.

Compass (hereafter V-Comp), showing comparable accuracy while running up to 20 times faster and better supporting narrow FoVs. The experimental data are available online,[1] the visualization code and a video are provided in the supplementary material and online.[2]

II. RELATED WORKS

VT&R methods fall broadly into two categories: map-based and mapless. Map-based approaches rely on explicit state estimation and localization, whereas mapless approaches use direct image-driven or reactive control.

A. Map-Based VT&R

Global Metric Maps: These methods build globally consistent pose graphs using vSLAM or VO. Mahdavian et al. [15] implemented VT&R with ORB-SLAM and YOLOv3 on a ground robot, requiring stereo vision and GPU. While accurate, such solutions are computationally heavy. Nourizadeh et al. [16] proposed a lightweight alternative using odometry and visual offsets to maintain a global pose. However, it remains highly wheel odometry-dependent.

Local Metric Maps: These approaches constrain localization to a taught trajectory. Furgale and Barfoot [4] built overlapping local submaps but required stereo vision and 358.4 kB/m storage. For MAVs, Warren et al. [5] and Pfrunder et al. [17] demonstrated VO-based VT&R using stereo and monocular vision respectively, but relied on calibrated cameras and GPU processing, making them unsuitable for ultra-light MAVs.

Topological Maps: Topological VT&R represents environments as graphs of visual places, avoiding full metric accuracy [18]. Early systems by Matsumoto et al. [19] relied on view databases and local image interpolation to generate motion commands. While efficient, they required omnidirectional or wide FoV cameras. These systems also scaled poorly in memory and often needed initial pose alignment or route indexing. Modern approaches use deep VPR: Sun et al. [20] employed self-supervised monocular descriptors for day-night matching, while Camara et al. [21] combined convolutional neural network-based features with image offsets for localization and heading correction. However, these methods depend on 1080p input, pre-trained networks, and GPU compute, unsuitable for ultra-light MAVs.

B. Mapless VT&R

Krajník et al. [22] proposed a system replaying recorded velocities and correcting heading via image offsets, while image-based visual servoing methods, like Caron et al.'s photometric control [23], use direct image gradients but demand high-resolution input, heavy computation or velocity profile encoding, making them unsuitable for ultra-light MAVs.

Insect-Inspired Navigation: Ants and bees navigate long distances using low-resolution vision and minimal neural resources, inspiring biologically grounded mapless VT&R. The V-Comp model [24], [25] stores panoramic outbound views and performs in silico scans during repeat, but suffers from high computational cost and cumulative storage. Van Dijk et al. [26] implemented a Fourier-based familiarity method on a MAV using a panoramic camera (adding $\approx 48\%$ mass) and relying on odometry, which contrasts with evidence that insects navigate with egocentric (retinotopic), orientation-specific visual familiarity rather than whole-image signatures or odometry [27]. To reduce storage, insect MB models were used alongside V-Comp [28], however, they remain limited to ground robots, required panoramic vision and scanning at each repeat step. While aerial adaptations with downward-facing views [29] or tree detection [30] offer partial solutions, real-world VT&R in MAVs remains unachieved.

Prior Works: Inspired by studies showing that the central complex in insect brains may modulate memory encoding in the MB based on the nest's relative direction [31], the R-Lat model was introduced [12]. The approach consisted of categorizing views as "left" or "right" relative to the route direction, enabling egocentric route following on a ground robot equipped with panoramic vision and in silico scans during the learning phase. Overall, most VT&R approaches are typically evaluated on ground robots, either with GPU computation, using panoramic cameras, or rely heavily on odometry during repeat phase. Transitioning from panoramic views with virtual scans on a ground robot to a physically constrained MAV with a narrow

---

[1] https://doi.org/10.57745/AC9ILI

[2] https://youtu.be/hQBL9Npc6cQ

Authorized licensed use limited to: ENSTA. Downloaded on June 19,2026 at 15:11:08 UTC from IEEE Xplore. Restrictions apply.

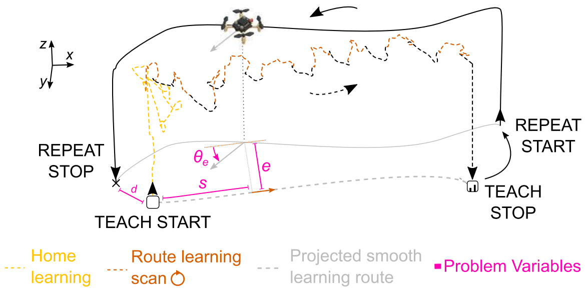

Fig. 2. Problem statement overview. During the Teach phase, the MAV is teleoperated (dashed) and stores a home memory (yellow) and route memories (orange). Teleoperation pauses briefly for autonomous yaw scans, whose drift forms the orange segment. In the Repeat phase, the displaced MAV returns using only visual familiarity, minimizing e, θ_e, s and d, and lands when the home vicinity is recognized.

FoV remains a challenge. For this reason, we introduce a compact VT&R framework enabling robust inbound route following and autonomous landing on an ultra-low-resolution narrow-FoV MAV (Antflie, Fig. 1) using minimal memory.

III. METHODOLOGY

A. Problem Statement

We address a route-based visual homing task in which a MAV must autonomously return to its starting point after a single outbound, teleoperated teach flight. Teleoperation denotes manual joystick control, with the MAV following pilot inputs except during brief autonomous scan events (rotation). During teaching, the MAV stores a home memory near takeoff and route memories along the outbound path, with autonomous yaw scans triggered at fixed intervals to acquire laterally oriented views. Before the inbound repeat flight, the MAV is displaced and must navigate home using only visual memories and familiarities inputs.

The objective is to minimize the final landing error d (Euclidean distance between takeoff and landing, Fig. 2). In the Frenet-Serret frame of the outbound path, this requires reducing the lateral error e, angular error θ_e, and longitudinal error s. If the MAV re-enters the taught path with approximately correct heading, a bearing-only controller minimizing θ_e can stabilize e, while constant forward velocity and end-route detection helps reduce s and thus landing error d.

To assess stability, we use the Lyapunov candidate

$$V(t) = \frac{1}{2}\left(K_h\theta_e^2 + K_e e^2 + K_s s^2\right), \tag{1}$$

where gains K_h, K_e, and K_s are set to 1 here. Convergence of $V(t)$ over time indicates stable homing.

B. Route Lateralized (R-Lat): Image Processing & Encoding

The R-Lat pipeline (Fig. 3) transforms each incoming frame into a compact neural code inspired by the insect optic lobes and MB, which stores visual snapshots as sparse associative patterns across Kenyon Cells (KCs). The processing consists of three steps.

1) Preprocessing: The tiny camera provides $324\times324px$ grayscale image (87° FoV). Each frame is blurred, downsampled to $24\times24px$, edge-enhanced with a Sobel filter, and flattened into a 576-D Projection Neuron (PN) vector.

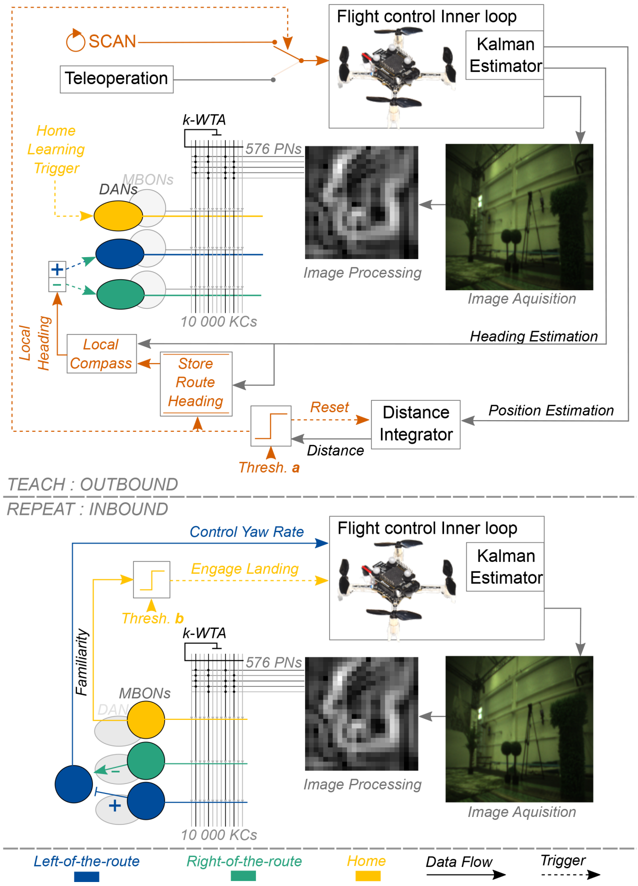

Fig. 3. Route Lateralized (R-Lat) navigation algorithm. During the teleoperated teach phase, the MAV stores first a home memory and left/right route memories by performing periodic autonomous yaw scans. In the repeat phase, left/right familiarity drives yaw velocity, and home familiarity triggers landing.

2) Expansion to Kenyon Cells: Each of the $N = 10{,}000$ KCs receives 4 randomly assigned PN inputs through a fixed binary matrix $W_1 \in \{0,1\}^{10{,}000\times576}$ [32], producing $EPSP = W_1 \cdot PN$.

3) Sparse Coding (k-WTA): Inspired by the inhibitory APL neuron in the MB, we apply a k-Winner-Take-All rule that keeps only the top $k = 1\%$ of activations. This yields $N_u = 100$ active KCs in a 10,000-bit binary Action Potential (AP) vector such that:

$$AP_i = \begin{cases} 1, & EPSP_i \in \{k \,\text{largest element}\} \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

This sparse code is used both for learning (teach phase) and for visual familiarity (repeat phase).

C. Route Lateralized (R-Lat): Teach Phase

During the Teach phase, the MAV forms three visual memories in the MB: one home memory and two route memories (left/right). All memories use the same anti-Hebbian synaptic rule: presenting a KC activation pattern (AP) depresses the active synapses of a selected MB Output Neuron (MBON). Each MBON stores a plastic binary weight vector $W_2 \in \{0,1\}^{10\,000}$, initialized to 1, and updated as:

$$W_{2i} = \begin{cases} 0, & \text{if } AP_i = 1, \\ W_{2i}, & \text{otherwise,} \end{cases} \tag{3}$$

Authorized licensed use limited to: ENSTA. Downloaded on June 19,2026 at 15:11:08 UTC from IEEE Xplore. Restrictions apply.

thus learning visual snapshots by removing the connections of the active KCs.

1) Home Learning: At takeoff, the pilot triggers a brief "home learning" event as an external dopaminergic (DAN) signal selecting the home MBON. Consequently, the acquired KCs patterns are written into W_2^H using (3). This forms a compact visual representation of the home vicinity.

2) Route Learning: As the drone is teleoperated along the outbound route, it periodically performs physical yaw scans of fixed amplitude to collect laterally oriented views (inspired by ants in Fig. 1(a)). Each scan produces a sequence of frames that are categorized as facing left or right relative to the route direction, and learned into the corresponding MBONs W_2^L and W_2^R. To determine whether the view is facing left or right of the route direction during the physical scan, we introduce the local compass.

a) Local compass for view categorization: A Kalman filter fuses IMU, ventral OF, and ToF measurements to estimate (i) yaw and (ii) local forward displacement. A scan is triggered every $a = 30$ cm of travel, and the current yaw at that moment is stored as the reference route direction. During the scan, the heading error $\theta_e = \theta_{\text{current}} - \theta_{\text{route}}$ determines whether each view faces left ($\theta_e > 0$) or right ($\theta_e < 0$). The corresponding DAN signal selects W_2^L or W_2^R, and (3) stores the KC pattern. Because scans reset the local displacement, it reduces the odometric drift during learning. This process results in three compact MB memories (home, left, right) learned in one outbound traversal.

D. Route Lateralized (R-Lat): Repeat Phase

During the Repeat phase, the current image is encoded into a KC activation vector AP (Section III-B) and compared with the three learned memories: left (W_2^L), right (W_2^R), and home (W_2^H) to output three familiarity values.

1) Familiarity: For any MBON, familiarity is defined as

$$\lambda = \frac{1}{N_u} \sum_{i=1}^{N} AP_i \, W_{2i}, \tag{4}$$

where $\lambda = 0$ indicates perfect familiarity and $\lambda = 1$ complete novelty.

2) Route Following: Yaw rate signal (ω) is driven by the familiarity difference between left and right memories:

$$\omega = k \cdot (\lambda^L - \lambda^R). \tag{5}$$

3) Home Detection and Landing: Landing is triggered when the home familiarity satisfies $\lambda^H < b = 0.1$. This threshold was tuned empirically: all views recorded in the home vicinity yield $\lambda^H < 0.1$, whereas stricter values (e.g., $b = 0.05$) occasionally failed to trigger landing under realistic image noise.

During Repeat, no odometry or heading estimation is used for navigation. The MAV flies at a fixed forward speed of 0.2 m/s. All high-level decisions are based purely on visual familiarity. The onboard Kalman filter is used exclusively for low-level stabilization and good execution of yaw rate and landing maneuvers.

IV. SIMULATION

We conducted offline simulations to compare R-Lat and V-Comp algorithms under controlled variations of world complexity, FoV, scan amplitude, and lateral error (Table I). The synthetic images were captured in the Gazebo ROS simulator. In the simulation context, we refer to the phases as *train* and *test*. The training route was L-shaped, 10 m long, and discretized into 20 positions (0.5 m spacing). The scan step was fixed to 5°, as in prior works [12], [28], which provides a good compromise between angular resolution and runtime. This step was kept constant across FoVs to isolate FoV and scan amplitude effects.

TABLE I
SIMULATION PARAMETERS AND TESTED VALUES

Parameter	Values
World complexity (H)	[2.3, 3.0, 4.6, 5.3]
FoV (deg)	[360, 315, 270, 225, 180, 135, 90, 45, 15]
Scan amplitude (deg)	[360, ±135, +90, ±45, ±15]
Lateral error (m)	[-0.8, -0.4, -0.2, 0, 0.2, 0.4, 0.8]

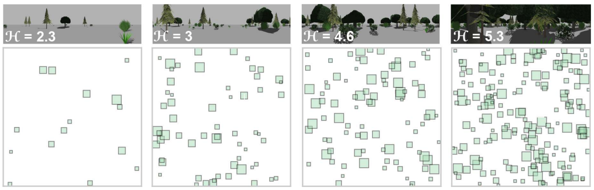

Fig. 4. Different world visual complexities measured by entropy (H), tested in simulation.

A. Simulation Setup

1) Train Phase: During training, a key difference between the two methods is when the scan amplitude is used: R-Lat applies it during training only, while V-Comp applies it during testing. In V-Comp, the number of stored views per memory is fixed at 20 (one per route position), independent of scan amplitude. With 4 worlds (Fig. 4) and 9 FoVs, this produces 36 distinct training setups. In R-Lat, the number of stored views per memory depends on scan amplitude: for example, ±45° with 5° steps yields 9 views per side (180 per memory, 360 total for left/right memories). A full 360° scan stores 3600 views per memory (7200 total). With 4 worlds, 9 FoVs, and 5 scan amplitudes, R-Lat produces 180 distinct training setups. Though R-Lat's training set size increases with scan amplitude, the size of the MB network and the storage footprint of the learned synaptic weights remain constant. It is noteworthy that each image passes through the model only once, as one-shot learning.

2) Evaluation Metrics: For both methods, we treat the MB-model output as a binary classifier of egocentric route direction sign and compute Accuracy $= \frac{TP+TN}{TP+TN+FP+FN}$, Precision $= \frac{TP}{TP+FP}$ and Recall $= \frac{TP}{TP+FN}$. Here TP/FP/TN/FN are defined on left/right turn decisions. For R-Lat, positive (P) = turn-left and negative (N) = turn-right. For V-Comp, a scan over orientations yields the angle that minimizes familiarity; the sign of this angle (left/right of the current heading) is mapped to the same P/N labels, so both methods are evaluated on an equivalent binary left/right task. We also record mean computation time per decision and a qualitative deployability for a narrow-FoV MAV.

3) Test Phase: In R-Lat, each memory setup is tested with 10,080 views, using the same grid of lateral errors, positions, and orientations. Scan amplitude only affects how many views are stored during training; at test time R-Lat always evaluates two familiarity scores (left/right) per pose, so the per-decision cost is independent of scan amplitude. In V-Comp, the number of tested

Authorized licensed use limited to: ENSTA. Downloaded on June 19,2026 at 15:11:08 UTC from IEEE Xplore. Restrictions apply.

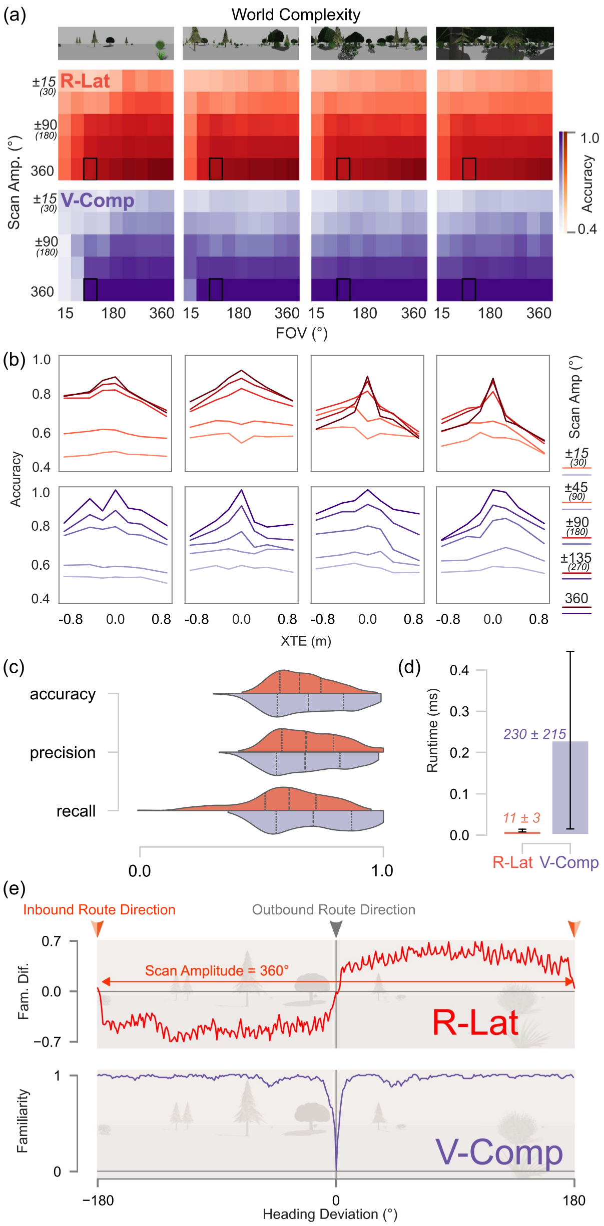

Fig. 5. Offline simulation analysis of R-Lat vs. V-Comp across FoV, scan amplitude, lateral error, and scene complexity. (a) Accuracy heatmaps; closed box = Antflie setup. (b) Accuracy vs. lateral error. (c) Accuracy/precision/recall. (d) Mean runtime. (e) Familiarity response.

views increases with scan amplitude because the algorithm evaluates a scan at each test step. For example, a 360° scan requires 725,760 tested views, whereas ±15° requires 60,480 views. Both methods still have 180 distinct testing setups overall.

B. Simulation Results

Fig. 5 summarizes the comparison between R-Lat and V-Comp across FoV, scan amplitude, lateral error, and scene complexity. In Figure 5(a) ($d = 0$), scan amplitude is the dominant factor for accuracy, with FoV only becoming limiting below 90° for both methods. R-Lat maintains consistent performance across scene complexities, reaching a plateau at ±90°, while V-Comp benefits more from maximum scan amplitudes. For a fixed FoV of 90° (Fig. 5(b)), V-Comp shows slightly higher tolerance to large cross-track error in high-complexity worlds, but the gap closes for smaller scan amplitudes.

Overall accuracy (Fig. 5(c)) varies less for R-Lat, while V-Comp attains a slightly higher mean. Precision is similar between methods, but R-Lat's lower recall suggests occasional missed turning decisions, which, due to their low occurrence, would be corrected in closed-loop navigation.

Runtime analysis (Fig. 5(d)) reveals a clear advantage for R-Lat (11 ± 3 ms) over V-Comp (230 ± 215 ms), as R-Lat's evaluation cost is constant, whereas V-Comp's scales directly with scan amplitude. Familiarity responses (Fig. 5(e)) show R-Lat producing a strong signed left/right signal with clear zero crossings at 0° (outbound) and 180° (inbound). In contrast, V-Comp's familiarity curve for a single stored view is symmetric by construction; the left/right decision in V-Comp is obtained from the signed orientation of the minimum of this curve, rather than from its magnitude.

We compared scan requirements for panoramic and narrow-FoV versions of R-Lat and V-Comp. In the training phase, panoramic R-Lat and V-Comp obtain multiple orientations via software rotation of stored panoramic images, while narrow-FoV versions must perform a physical yaw sweep through the scan amplitude (Fig. 5(e)). In R-Lat, the number of scan steps equals the scan amplitude divided by the angular resolution, with local compass needed only for the narrow-FoV case. In V-Comp, both panoramic and narrow-FoV store only the outbound-facing view at each location; the panoramic version generates heading in silico, whereas the narrow-FoV version must physically sweep toward the outbound-facing view, requiring local compass.

In the testing phase, R-Lat (both panoramic and narrow) requires no scans and no local compass, evaluating only two familiarity scores per location. V-Comp, however, must scan at test time: panoramic V-Comp does this in silico, while narrow-FoV V-Comp performs a continuous physical sweep through the scan amplitude, again requiring local compass.

In summary, R-Lat and V-Comp achieve comparable accuracy for FoVs above 90°, with scan amplitude being the main determinant of performance. Scene complexity has limited influence, but R-Lat offers major benefits in computation time, direction discrimination, and narrow-FoV feasibility.

V. REAL-WORLD EXPERIMENTS

A. Experimental Setup

Experiments were conducted in the Mediterranean Flight Arena (MFA), equipped with artificial visual cues and an 18-camera VICON motion capture system for ground-truth tracking. The Antflie quadcopter (Fig. 6) is a 33 g Crazyflie 2.1 micro-drone carrying an AI-deck with an ultra-low-power Himax camera (0.2 g, 87° FoV, 324 × 324 px), a PMW3901 OF sensor, and a VL53L1x ToF for altitude control, powered by a 1S–250 mAh LiPo battery. State estimation via an onboard Kalman filter ran on the AI-deck, while the neural network was executed on an external PC (Python) for rapid prototyping, using an Intel i7 (8th gen) CPU. Images were transmitted over Wi-Fi via ROS, and the PC computed the desired commands during the Repeat phase, relaying them to the MAV via radio dongle. We report the metrics using medians and Median Absolute Deviation (MAD).

Authorized licensed use limited to: ENSTA. Downloaded on June 19,2026 at 15:11:08 UTC from IEEE Xplore. Restrictions apply.

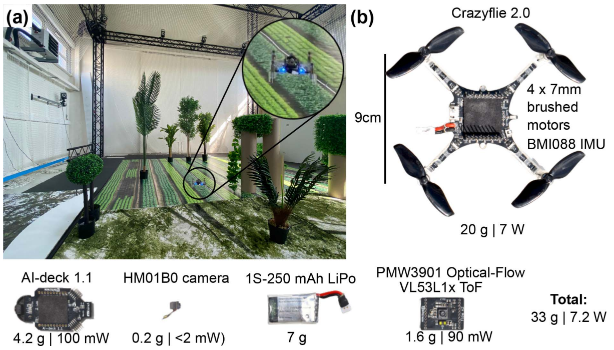

Fig. 6. Experimental setup comprising (a) the Mediterranean Flight Arena (MFA) with motion capture and visual cues, and (b) the 33 g Antflie drone featuring low-power sensors for autonomous flights.

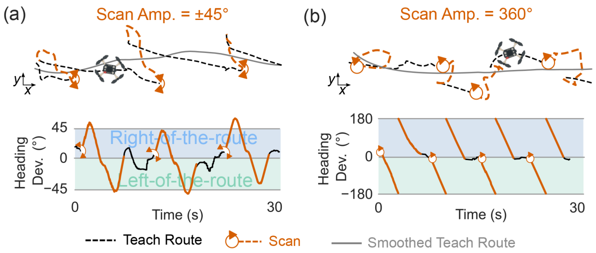

Fig. 7. Real-world validation of local heading estimation during teach-phase scans. With (a) ± 45° and (b) 360° scans, Antflie centers rotations on the instantaneous route direction from IMU and OF, and the resulting deviations from the smoothed path (gray) enable accurate view categorization on the right or on the left of the route without global route knowledge.

B. Teach-Phase Validation

We first confirmed that Antflie could align its outbound scans with the instantaneous route direction using IMU, OF and ToF without access to the global smoothed path. In both ±45° and full 360° scans (Fig. 7), measured local headings (orange) closely tracked deviations from the smoothed outbound path (gray), confirming that the local compass provides accurate egocentric orientation within the route frame. This capability is a prerequisite for fully local, odometry-light route learning with narrow FoV vision. The larger drift during the ±45° scans stems from the two rapid yaw reversals, whereas the 360° scan is a smooth continuous rotation that allows the hover controller to recenter the MAV.

C. Experimental Results

1) Baseline Repeatability on a Straight Route: We validated the complete VT&R pipeline on a 3 m straight route using periodic ±45° scans (Fig. 7(a), Fig. 8 Exp. S). After a single outbound learning flight, Antflie completed n = 3 autonomous repeats with a median lateral error of 4 cm and a median heading error of 5.0° (Table II, Exp. S). This baseline confirmed that route memories and familiarity steering achieve highly repeatable path reproduction under minimal geometric complexity.

2) Inbound L-Shaped Routes and Autonomous Landing: We next evaluated Antflie on four distinct L-shaped routes (Fig. 8), each learned in a single outbound flight with periodic 360° scans. In total, $n = 24$ autonomous inbound flights were performed from displaced start positions. Start conditions spanned absolute lateral errors from 5 mm to 73 cm and absolute heading errors from 0.5° to 67°. Due to the battery constraints of the 33 g platform, each outbound smoothed route was restricted to ∼ 4.7 m in length because of the scanning maneuvers consuming energy, even though the drone physically traveled ∼ 10 m during learning. During the Repeat phase, the closed-loop control pipeline, including image capture, Wi-Fi transfer, neural network computation, and radio transmission of commands, operated at 8 Hz. Combined with the fixed forward speed of 0.20 m/s, this corresponds to one steering decision every 2.5 cm of travel.

Path fidelity: Although all outbound trajectories share a broad “L-like” topology, their geometries differ markedly in curvature, turning angle, and overall shape (Fig. 8). Inbound trajectories consistently reproduced the specific outbound path that was learned in each experiment, rather than converging to a generic L pattern or straight beeline. This shows that route following is driven by route-specific visual familiarity, not by a geometric prior on path shape.

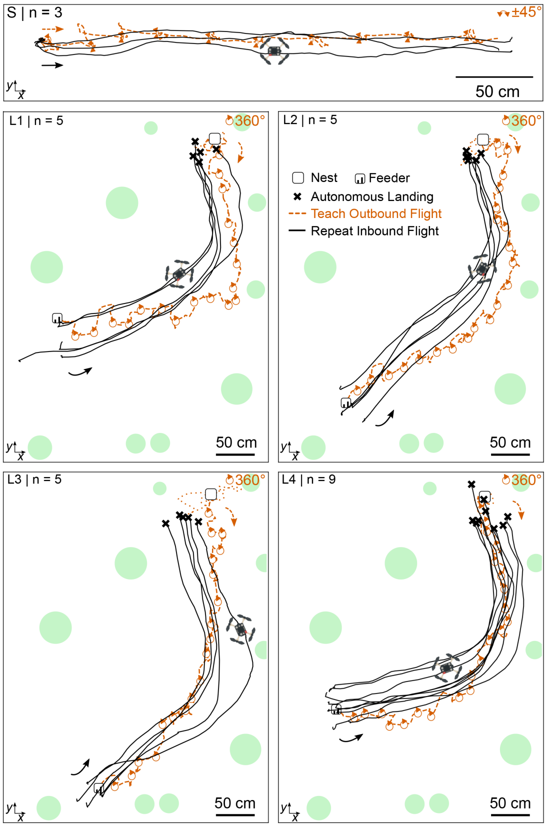

Fig. 8. Teaching one straight route (S) with ± 45° scans and autonomous repeats, and four different L-shaped (L) routes with 360° scans followed by autonomous inbound repeats and landing. The MAVs consistently returned from the feeder to the nest using only visual route memories.

TABLE II
MEDIAN (± MAD) OF ERROR METRIC PER EXPERIMENT

Exp.	Heading error[°]	Lateral error[m]	Landing error[m]
S	5±3	0.04±0.02	–
L1	23±13	0.21±0.07	0.23±0.08
L2	27±20	0.22±0.09	0.32±0.02
L3	31±20	0.14±0.09	0.45±0.04
L4	14±10	0.12±0.07	0.39±0.06
ALL	17±12	0.13±0.08	0.34±0.09

Authorized licensed use limited to: ENSTA. Downloaded on June 19,2026 at 15:11:08 UTC from IEEE Xplore. Restrictions apply.

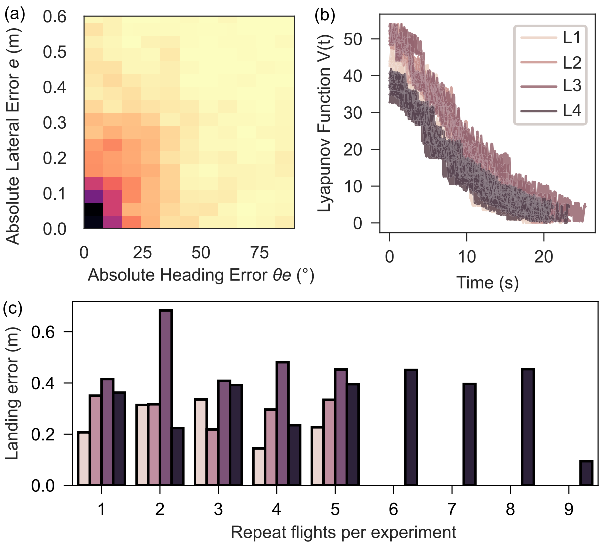

Fig. 9. (a) Lateral/heading error heatmap. (b) Lyapunov $V(t)$ over normalized time. (c) Landing errors per trajectory and experiment.

Error stability: Across all repeats, the absolute heading error was 17.0° ± 12.3° and lateral error was 0.13 m ± 0.08 m (Table II, Fig. 9(a)). The Lyapunov function $V(t)$ (1) decreased monotonically in all cases, reaching $V(t)/V(0) < 1\%$ within a median of 16.6s (IQR: 15.3–19.0s) (Fig. 9(b)), confirming rapid convergence of cross-track (lateral), heading, and along-track errors.

Landing repeatability: Visual home familiarity triggered autonomous descent when $\lambda^H < 0.1$. Across all 24 landings, the median error was 0.34 m (MAD: 0.09 m), with a best case of 0.09 m (equal to body length) and worst case of 0.68 m (7 times the body length). The coefficient of variation (C_V) was 37%, indicating strong repeatability under purely vision-based narrow-FoV and ultra-low resolution conditions.

Memory footprint: When storing a learned route for future repeat flights, each MB memory (W_2) is stored as a binary bit-packed 10,000-bit vector (1 b/synapse), requiring only 1.22 kB per memory. Thus, the three memories together occupy only 3.6 kB on disk. This fixed-size representation is more efficient than Compressed Sparse Row (CSR) formats (tested in [12]), whose size depends on sparsity. For comparison, storing the same synaptic weights as CSV files requires ≈120 kB, about 34 times larger, and storing the raw image would be even more memory-intensive.

VI. CONCLUSION

We demonstrated that route-lateralized visual memories learned during outbound physical scans, combined with a local compass, enable a 33-gram MAV with narrow FoV and ultra-low-resolution vision to perform reliable autonomous inbound homing and landing, without maps, global localization, or odometry during repeat, and using only local segment odometry during teach. Furthermore, the R-Lat model with limited FoV is biologically plausible: the physical scans account for the regular active sampling head and body rotations displayed by ants and bees, whose visual fields also contain blind spots [14], and the local odometry, involved during scanning behaviours, appears to reside in the insect central complex [33]. In simulation, our framework matched a state-of-the-art insect visual compass under diverse conditions and shows high promise for real-time capability with narrow FoV. In real flights, Antflie reproduced taught routes over a cumulative distance of 110 m with low lateral error (0.13 ± 0.08 m) and executed 24 autonomous landings with high precision (0.34 ± 0.09 m), showing high repeatability. The error analysis in [12], showed that, in our case $N = 10,000$ KCs, it supports routes of about 35 m before left/right misclassification exceeds 1%, then, capacity scales linearly with N (e.g., $N = 50,000$ supports 200 m while still requiring only 18 kB).

With its frugal bitwise computation, fixed 4 kB memory footprint, and 24 × 24 visual input, R-Lat is well suited for possible embedded deployment on microcontrollers such as the Crazyflie's ARM Cortex-M7 or FPGA. Unlike our prototype's 8 Hz Wi-Fi–limited loop, an onboard implementation would run at much higher control rates, enabling faster MAV or micro-robot navigation. Importantly, R-Lat has constant runtime during the repeat phase: each decision requires only two bitwise AND + popcount operations on a fixed-size 10,000-bit vector. In contrast, V-Comp must evaluate a full orientation scan at repeat time, making its computational cost grow linearly with scan amplitude and difficult to execute on low-power processors. Overall, R-Lat is practical across platforms and cameras—requiring no deblurring, calibration, stereo, or panoramic vision—and offers a clear advantage over V-Comp for real-time embedded operation. A few kilobytes of ant-inspired memory suffice for reliable, empirically Lyapunov-stable homing and landing on a micro-drone: no maps, no SLAM, only ultra-frugal vision.

Future works will address perceptual aliasing and multiple route memories by maintaining several MB memories linked through a lightweight topological structure, linked by time or space-dependent memories, an extension of home and route memories, such as places or "route" cells, enhancing the algorithm performance in longer or more complex scenarios. A promising direction is to investigate how robustness to dynamic scene changes depends on camera FoV. While our results demonstrate that a narrow FoV is sufficient for stable route following in static conditions, prior panoramic implementations [12] showed robustness to motion and illumination variations. This motivates a systematic comparison to determine whether wider FoVs intrinsically confer greater robustness. Finally, landing accuracy depends on lateral error, the home-detection threshold, and the extent of the home-learning area, tightening or tuning these factors should reduce dispersion.

REFERENCES

[1] T. A. Ward, C. J. Fearday, E. Salami, and N. B. Soin, "A bibliometric review of progress in micro air vehicle research," *Int. J. Micro Air Veh.*, vol. 9, no. 2, pp. 146–165, Jun. 2017.

[2] G.-Z. Yang et al., "The grand challenges of science robotics," *Sci. Robot.*, vol. 3, no. 14, Jan. 2018, Art. no. eaar7650 .

[3] A. Patel, S. Karlsson, B. Lindqvist, C. Kanellakis, A.-A. Agha-Mohammadi, and G. Nikolakopoulos, "Towards energy efficient autonomous exploration of Mars lava tube with a martian coaxial quadrotor," *Adv. Space Res.*, vol. 71, no. 9, pp. 3837–3854, May 2023.

[4] P. Furgale and T. D. Barfoot, "Visual teach and repeat for long-range rover autonomy," *J. Field Robot.*, vol. 27, no. 5, pp. 534–560, Sep. 2010.

[5] M. Warren, M. Greeff, B. Patel, J. Collier, A. P. Schoellig, and T. D. Barfoot, "There's no place like home: Visual teach and repeat for emergency return of multirotor UAVs during GPS failure," *IEEE Robot. Automat. Lett.*, vol. 4, no. 1, pp. 161–168, Jan. 2019.

[6] B. Al-Tawil, T. Hempel, A. Abdelrahman, and A. Al-Hamadi, "A review of visual SLAM for robotics: Evolution, properties, and future applications," *Front. Robot. AI*, vol. 11, Apr. 2024, Art. no. 1347985.

Authorized licensed use limited to: ENSTA. Downloaded on June 19,2026 at 15:11:08 UTC from IEEE Xplore. Restrictions apply.

[7] S. Lowry et al., “Visual place recognition: A survey,” *IEEE Trans. Robot.*, vol. 32, no. 1, pp. 1–19, Feb. 2016.

[8] B. Webb and A. Wystrach, “Neural mechanisms of insect navigation,” *Curr. Opin. Insect Sci.*, vol. 15, pp. 27–39, Jun. 2016.

[9] R. Wehner, M. Boyer, F. Loertscher, S. Sommer, and U. Menzi, “Ant navigation: One-way routes rather than maps,” *Curr. Biol.*, vol. 16, no. 1, pp. 75–79, Jan. 2006.

[10] P. Ardin, F. Peng, M. Mangan, K. Lagogiannis, and B. Webb, “Using an insect mushroom body circuit to encode route memory in complex natural environments,” *PLOS Comput. Biol.*, vol. 12, no. 2, Feb. 2016, Art. no. e1004683.

[11] G. C. H. E. D. Croon, J. J. G. Dupeyroux, S. B. Fuller, and J. A. R. Marshall, “Insect-inspired AI for autonomous robots,” *Sci. Robot.*, vol. 7, no. 67, Jun. 2022, Art. no. eabl6334.

[12] G. G. Gattaux, A. Wystrach, J. R. Serres, and F. Ruffier, “Route-centric ant-inspired memories enable panoramic route-following in a car-like robot,” *Nature Commun.*, vol. 16, no. 1, 2025, Art. no. 8328.

[13] C. A. Freas and K. Cheng, “Visual learning, route formation and the choreography of looking back in desert ants, melophorus bagoti,” *Animal Behav.*, vol. 222, Apr. 2025, Art. no. 123125.

[14] S. Schwarz, A. Narendra, and J. Zeil, “The properties of the visual system in the australian desert ant melophorus bagoti,” *Arthropod Struct. Develop.*, vol. 40, no. 2, pp. 128–134, Mar. 2011.

[15] M. Mahdavian, K. Yin, and M. Chen, “Robust visual teach and repeat for UGVs using 3D semantic maps,” *IEEE Robot. Autom. Lett.*, vol. 7, no. 4, pp. 8590–8597, Oct. 2022.

[16] P. Nourizadeh, M. Milford, and T. Fischer, “Teach and repeat navigation: A robust control approach,” in *Proc. IEEE Int. Conf. Robot. Automat.*, Yokohama, Japan, May 2024, pp. 2909–2916.

[17] A. Pfrunder, A. P. Schoellig, and T. D. Barfoot, “A proof-of-concept demonstration of visual teach and repeat on a quadrocopter using an altitude sensor and a monocular camera,” in *Proc. IEEE Can. Conf. Comput. Robot Vis.*. Montreal, QC, Canada, May 2014, pp. 238–245.

[18] E. Garcia-Fidalgo and A. Ortiz, “Vision-based topological mapping and localization methods: A survey,” *Robot. Auton. Syst.*, vol. 64, pp. 1–20, Feb. 2015.

[19] Y. Matsumoto, M. Inaba, and H. Inoue, “Visual navigation using view-sequenced route representation,” in *Proc. IEEE Int. Conf. Robot. Automat.*, Minneapolis, MN, USA, 1996, pp. 83–88.

[20] L. Sun et al., “Robust and long-term monocular teach and repeat navigation using a single-experience map,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Prague, Czech Republic, Sep. 2021, pp. 2635–2642.

[21] L. G. Camara, T. Pivonka, M. Jilek, C. Gabert, K. Kosnar, and L. Preucil, “Accurate and robust teach and repeat navigation by visual place recognition: A CNN approach,” in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*. Las Vegas, NV, USA, Oct. 2020, pp. 6018–6024.

[22] T. Krajnik, F. Majer, L. Halodova, and T. Vintr, “Navigation without localisation: Reliable teach and repeat based on the convergence theorem,” in *Proc. 2018 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1657–1664.

[23] G. Caron, E. Marchand, and E. M. Mouaddib, “Photometric visual servoing for omnidirectional cameras,” *Auton. Robots*, vol. 35, no. 2-3, pp. 177–193, Oct. 2013.

[24] J. Zeil, M. I. Hofmann, and J. S. Chahl, “Catchment areas of panoramic snapshots in outdoor scenes,” *J. Opt. Soc. Amer. A*, vol. 20, no. 3, Mar. 2003, Art. no. 450.

[25] J. C. Knight et al., “Insect-inspired visual navigation on-board an autonomous robot: Real-world routes encoded in a single layer network,” in *Proc. Artif. Life Conf. Proc.*, 2019, pp. 60–67.

[26] T. V. Dijk, C. D. Wagter, and G. C. H. E. D. Croon, “Visual route following for tiny autonomous robots,” *Sci. Robot.*, vol. 9, no. 92, Jul. 2024, Art. no. eadk0310.

[27] M. Mangan and B. Webb, “Spontaneous formation of multiple routes in individual desert ants (cataglyphis velox),” *Behav. Ecol.*, vol. 23, no. 5, pp. 944–954, 2012.

[28] G. Gattaux, A. Wystrach, F. Ruffier, and J. Serres, “Enhancing ant-inspired visual compass with focused visual scan in a compact robot,” in *Proc. 7th IEEE Int. Conf. Artif. Intell. Circuits Syst.*, 2025, pp. 1–5.

[29] J. Stankiewicz and B. Webb, “Looking down: A model for visual route following in flying insects,” *Bioinspiration Biomimetics*, vol. 16, no. 5, Sep. 2021, Art. no. 055007.

[30] W. Kuang, H. W. Ho, Y. Zhou, and S. A. Suandi, “ForaNav: Insect-inspired online target-oriented navigation for MAVs in tree plantations,” *IEEE Robot. Automat. Lett.*, vol. 10, no. 7, pp. 7063–7069, Jul. 2025.

[31] A. Wystrach, F. L. Moël, L. Clement, and S. Schwarz, “A lateralised design for the interaction of visual memories and heading representations in navigating ants,” *Animal Behav. Cognit.*, vol. 32, Aug. 2020, Art. no. hal-03052606.

[32] S. J. Caron, V. Ruta, L. F. Abbott, and R. Axel, “Random convergence of olfactory inputs in the drosophila mushroom body,” *Nature*, vol. 497, no. 7447, pp. 113–117, 2013.

[33] C. A. Freas and A. Wystrach, “Scanning and active sampling behaviours emerge from conserved insect neural circuits,” bioRxiv, vol. 2025, 2025, Art. no. 10.

Authorized licensed use limited to: ENSTA. Downloaded on June 19,2026 at 15:11:08 UTC from IEEE Xplore. Restrictions apply.