2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

IntuFly: Intuitive Continuous Hand–Gaze Control for UAVs

Junsheng Xu[1,*], Ke Ma[1,*], Xinde Li[1,2,†], Chengxiang Yu[1], Zeyu Zhang[1], Zhentong Zhang[1]

***Abstract*— Operating Unmanned Aerial Vehicles (UAVs) remains challenging for non-experts because single-modality interfaces distort intent: gesture-only systems depend on discrete vocabularies and mode switches that break continuity and raise cognitive load, while gaze-only control offers limited dimensionality and is vulnerable to Midas-touch and saccadic jitter. We present IntuFly, an intuition-driven hand–gaze framework in which hands draw the path to give continuous 3D translation and eyes set heading and lock targets, preserving intent continuity and reducing effort. To overcome cross-stream asynchrony and noise, our deployment-oriented fusion layer performs timestamp-consistent late fusion with stale-frame dropping and lightweight stabilization, yielding stable closed-loop operation at more than 25 Hz on commodity hardware. In simulation racing, novices fly faster on shorter paths than a Remote controller (RC) baseline, and intermediates select shorter, smoother yet more conservative lines; Subjective scales indicate lower workload and higher usability. In mobile target tracking, adding gaze produces faster responses with near-complete line-of-sight (LOS) coverage under identical limits. The same perception–control stack runs stably on an indoor DJI Tello platform with behavior consistent with simulation, demonstrating sim-to-real feasibility. These results show that IntuFly lowers the learning barrier for non-expert users while preserving fine control and stability, offering a deployable path toward intuitive, continuous human–UAV cooperative flight. Our code is publicly available at https://github.com/Crotonbee/IntuFly.**

I. Introduction

With the widespread adoption of UAV technology across military, agriculture, logistics, and entertainment, the barrier to entry and continuous controllability for non-expert users have become increasingly salient [1]. Human–drone interaction (HDI) has advanced rapidly in recent years [2], [3], yet consumer-facing control methods still exhibit steep learning curves, cumbersome mode switching, and limited support for continuous control: traditional RC operation relies on physical transmitters or complex ground-station interfaces, demanding extensive training and motor skills and offering limited intuitiveness [4], [5]; purely gesture-based control typically hinges on discrete gesture sets and explicit mode switches, disrupting the continuity of intent expression; purely gaze-based control, while naturally enabling "look-where-to-go" pointing, remains a single modality with limited control dimensionality and is susceptible to Midas touch (unintended activations from non-intentional fixations) and noise from jitter and rapid saccades [6], thereby constraining fine maneuvers and long-duration flight. Meanwhile,

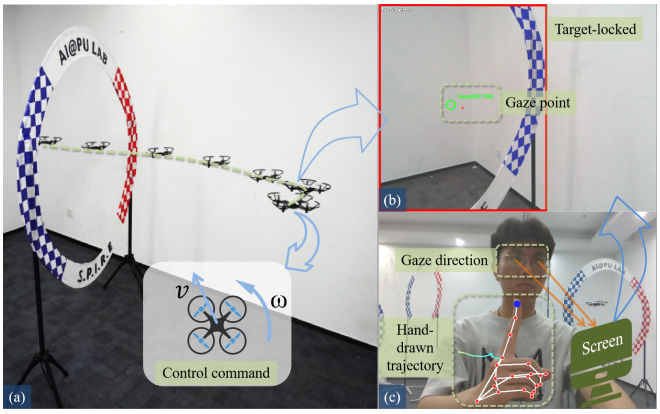

Fig. 1: (a) UAV trajectory through the frame under IntuFly control. (b) Airborne camera image feedback and gaze point target selection visualized on the screen. (c) User's hand-drawn trajectory and eye movement control input.

HDI roadmaps and recent surveys emphasize that achieving out-of-the-box operation for non-experts requires prioritizing low-cognitive-load multimodal natural interaction and continuous-control paradigms [2], [3].

We present IntuFly, a multimodal, intuition-driven UAV continuous-control framework that combines hand-drawn trajectory input with gaze-based yaw and target lock:

Hand-drawn trajectory for translation. Activated by a specific "GO" gesture, the user sketches the desired motion in midair; the system continuously maps the 3D hand trajectory to velocity commands, eliminating discontinuities introduced by discrete commands and mode switching.

Gaze for yaw and target locking. The user "looks where to go"; the UAV aligns its heading with the gaze direction or the selected target—analogous to human walk–turn–search–fixate [7], [8]. Once the gaze dwells for about 0.5 s within a detection box, the target is locked; heading is then closed-loop regulated using the offset between the target center and the image center. If the gaze leaves for more than 2 s, the target is unlocked.

With this division of labor, the user only needs to hand-draw translational intent, while the system semi-autonomously handles fine heading regulation for target alignment, substantially reducing operator burden and improving tracking stability [6]. The two channels are fused at the control layer using unified timestamps with latest-frame alignment and stale-frame dropping to mitigate latency, achieving temporal alignment and robust fusion—preserving the continuity and expressiveness of hand trajectories while delegating heading change and target selection to gaze. The interaction closely follows human intuition, significantly

*Junsheng Xu and Ke Ma contributed equally to this work.
†Corresponding author: xindeli@seu.edu.cn
[1]Southeast University, China
[2]Northeastern University, China

lowering the entry barrier and cognitive load for non-expert users. The workflow of the IntuFly system is shown in Figure 1.

We conduct a systematic evaluation in large-scale simulation and on an indoor real-world racetrack. In the racing task against an RC baseline, covering novice and intermediate operators, novices show clear gains in lap time and path length over RC, whereas intermediates exhibit smoother trajectories and shorter lines at the cost of reduced speed. In the mobile target-tracking task against the Hand-only ablation, IntuFly achieves lower latency and higher LOS coverage, consistently keeping the target centered in the Field of View (FoV). Subjective scales (Raw NASA-TLX, SUS, UEQ) [9], [10], [11]further indicate lower operator workload and improved usability and experience. The same perception–control stack transfers seamlessly to the physical platform for closed-loop operation, demonstrating sim-to-real feasibility. Overall, the results show that IntuFly provides a low-barrier, intuitive, and controllable UAV interaction for non-expert users and runs stably on real hardware.

Our main contributions are summarized as follows.

- An intuition-driven dual-channel hand–gaze paradigm for continuous control. Hand trajectories command continuous 3D translation while gaze sets yaw and target lock, overcoming gesture-only discontinuities and gaze-only dimensional limits/jitter to lower the barrier for non-experts.
- A deployment-oriented real-time fusion and stabilization stack. Centered on timestamp-consistent late fusion with stability constraints to tame cross-stream asynchrony and jitter, IntuFly sustains more than 25 Hz closed-loop control on commodity hardware and transfers directly to real robots.
- System-level validation across simulation and real hardware. IntuFly outperforms an RC baseline in racing and yields faster responses with near-complete LOS coverage in target tracking, while subjective scales show lower workload and higher usability and the same stack closes the loop on a DJI Tello, confirming sim-to-real feasibility.

II. Related Work

While naturalistic interfaces (e.g., voice, whole-body, physical controllers) are well-explored in automotive [12] and construction domains [13], these applications typically involve planar or axis-constrained motion. In contrast, UAVs demand continuous 6-DoF spatial control, necessitating modalities specifically fused for 3D flight.

A. Gesture-based Control for UAVs

Early studies explored mappings from 3D body pose and hand gestures to UAV commands, demonstrating that hand and upper-limb motions can serve as intuitive control channels; however, these interfaces typically rely on limited command vocabularies and cumbersome mode switching, leading to fatigue during prolonged use [14], [15]. More recently, MediaPipe-based real-time hand keypoint and gesture recognition has substantially lowered the deployment barrier, yet mainstream systems still trigger discrete commands from discrete gesture sets, making it difficult to provide continuous control throughout the entire flight [4], [5], [16], [17]. IntuFly integrates high-level task directives with low-level continuous control: discrete gestures issue task-oriented commands, while an online continuous trajectory input enables continuous control during flight, avoiding frequent mode switching and reducing both control complexity and the operator's entry barrier.

B. Gaze-based UAV Control and the Midas Touch Problem

Gaze provides a direct channel for conveying user intent and has been leveraged for onboard gimbal control (gaze-directed capture), point-of-interest guidance, and assisted teleoperation, offering pointing efficiency comparable to—or better than—traditional RC controllers, particularly for novice users [7], [18], [19]. Similarly, in robotic teleoperation, gaze is commonly used to infer operator intent, which the system then maps to safe, feasible trajectories [8]. However, prior studies have identified Midas touch and saccadic jitter as sources of false activations and instability, introducing unintended disturbances to control [6]. We adopt a gaze interaction scheme of yaw-by-gaze with target selection by fixation. Through calibration, fixation confirmation, and filtering, we suppress gaze-point error and inadvertent activations; moreover, we avoid delegating velocity and translation entirely to gaze, thereby reducing velocity noise induced by Midas touch and saccadic jitter.

C. Hand–Eye Multimodal Fusion Control

Single interaction modalities have inherent limitations; fusing multiple natural modalities is a principled direction for improving a UAV control system's fault tolerance, interaction efficiency, and user experience [20]. Broader HDI research supports a "look-to-select, hand-to-command" division of labor, which reduces false activations and balances channel workload [21], [22]. In UAV control, Di Vincenzo *et al.* combined wearable eye tracking with vision-based hand gestures for multimodal drone interaction [23]; however, the approach depends on dedicated eye-tracking hardware and an offset-based hand mapping that resembles joystick input, and is evaluated only in a virtual environment with real-world flight left for future work. By contrast, IntuFly follows this division of labor by delegating yaw control and target selection to gaze and using hand-drawn trajectories for continuous translational commands; it further mitigates latency via latest-frame alignment and stale-frame dropping, and improves robustness through dual-channel anti-jitter (Kalman filtering and exponential moving average) together with deadband, saturation, and slew-rate shaping, while validating the same perception–control stack both in simulation and on a real UAV.

III. Method

A. Core Design of IntuFly: Hand–Eye Division of Labor.

IntuFly centers on a two-channel natural interface: the hand governs translational motion, while the eye governs

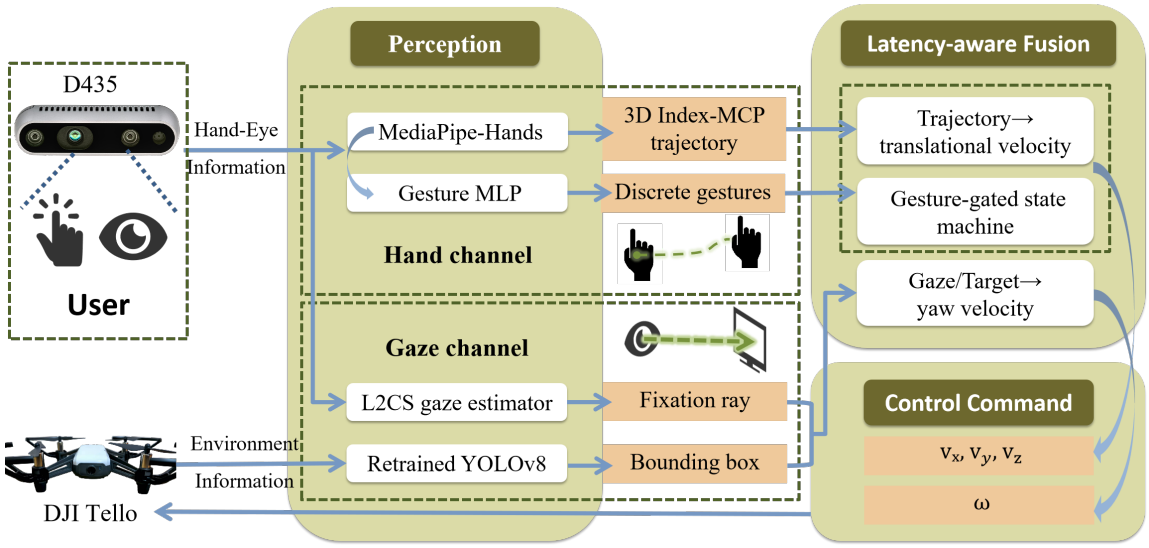

Fig. 2: Schematic of our proposed IntuFly framework. From the D435 we acquire hand–eye signals and from the onboard camera the scene images; in the perception layer, the hand channel (MediaPipe-Hands + a lightweight gesture MLP) outputs a 3D index-MCP trajectory and discrete gestures, while the gaze channel (L2CS with a retrained YOLOv8) yields a fixation ray and target boxes. A latency-aware fusion module aligns time stamps, drops stale frames, maps the trajectory to body-frame translational velocity under a gesture-gated state machine, maps gaze/target offsets to yaw rate, and finally issues the control command.

heading and point-of-regard. Concretely, a small, easy-to-learn set of discrete gestures issues top-level task controls (takeoff, emergency stop, speed up, continuous-control mode, decelerate); specifically, the "GO" gesture activates a control loop that converts hand-drawn trajectories into continuous, three-axis body-frame velocity commands. In parallel, gaze control maps image-plane pixel offset to yaw rate for continuous heading control; when the point of regard dwells within a target bounding box for 0.5 s, the target is automatically locked, and if gaze departs from the target bounding box for more than 2 s, it is unlocked. Upon target lock, the system engages assisted target-tracking. The two control streams are fused in a closed loop into a single actuation update—body-frame linear velocities and yaw rate —which is dispatched to the flight controller in one command. The overall IntuFly architecture is illustrated in Figure 2.

B. Top-Level Discrete Gesture Control and Multilayer Perceptron (MLP)-based Gesture Recognition

IntuFly adopts a division of labor in which discrete gestures handle top-level task control, while continuous channels realize low-level control. A small, easy-to-learn gesture set issues takeoff, emergency-stop, speed-up, decelerate and "enter continuous control" commands; specifically, the "GO" gesture activates and maintains the continuous trajectory control loop. The 'BRAKE' gesture triggers an emergency stop, resetting the system to a safe hovering state. To resume maneuvering, the user must re-engage the continuous control loop by performing the 'GO' gesture again. This design ensures safety by requiring explicit intent to re-initiate motion. When the user's hand is relaxed, the system enters a No-gesture state and holds the last valid output, allowing in-air hand repositioning and rest without triggering translational commands. In this design, the compact and intuitive gesture set is limited to task-level toggling and gain management, while continuous control is delegated to the trajectory loop and the gaze-driven yaw loop. Crucially, unlike traditional approaches where gestures switch between control axes, our 'GO' gesture functions solely as an activation mechanism. This distinction eliminates the cognitive load of frequent functional mode switching while maintaining necessary safety protocols, thereby lowering the entry barrier and preserving the continuity of intent expression.

To ensure robustness against inter-user variations where standard libraries often fail, we designed and trained a custom lightweight two-layer MLP. Its input is a 63-dimensional normalized feature vector derived from the 21 MediaPipe Hands keypoints. To accommodate inter-user variation in hand shapes and gesturing habits, we collected a dataset from 10 participants with 8,500 labeled samples; the model achieves 99.8% validation accuracy across all gesture classes in unseen user tests. The definition of top-level discrete gestures is shown in Figure 3.

C. Continuous Control Mapping

When the user performs the "GO" gesture, the system concurrently activates two continuous-control pathways: the hand-drawn trajectory drives three-axis body-frame translational velocities, while gaze commands yaw rate and, when required, performs target locking.

a) Hand channel: From the hand stream, we read the depth Z (meters) at the pixel of the index metacarpophalangeal (MCP) joint (u, v). The point is back-projected to

Algorithm 1 Overall process for IntuFly

Input: RGB–D (D435: hands & face), onboard-camera (UAV).

Output: $\mathbf{u} = [v_x,\ v_y,\ v_z,\ r]$.

```
Init filters and buffers; init state {airborne=0, draw=0, γ=1}.
for each control cycle do
    Analyze sense from D435: hand–3D and gaze direction.
    Grab onboard-camera frame and run YOLOv8 on it; overlay detection boxes on the screen.
    Update state by gesture (ok / go / Speed up / Decelerate / brake).
    if brake then
        Publish [0, 0, 0, 0]; continue
    end if
    v ← [0, 0, 0].
    if draw then
        KF–smooth hand; append to traj.; intent = diff of two m-frame means.
        Deadband + EMA; map to body; scale by γ; apply magnitude & slew limits → v.
    end if
    EMA–smooth gaze; try screen lock with boxes (on 0.5 s / off 2 s).
    Set yaw reference: screen target center if locked, else smoothed gaze.
    r from screen horizontal offset (deadband); limit r by mag & slew.
    if stale (> 80 ms) or not airborne then
        v ← 0;  r ← 0
    end if
    Publish [v_x, v_y, v_z, r].
end for
```

the camera frame under a pinhole model to obtain (X, Y, Z):

$$X = \frac{u - c_x}{f_x} Z, \qquad Y = \frac{v - c_y}{f_y} Z. \tag{1}$$

where f_x and f_y denote the effective focal lengths along the horizontal and vertical directions (in pixels), and c_x, c_y are the coordinates of the principal point on the image plane (in pixels).

To suppress quantization noise and jitter from missing measurements, we stabilize the temporal trajectory with a constant-velocity Kalman filter. Let $\{\mathbf{p}_t\}$ denote the smoothed 3D hand trajectory, where $\mathbf{p}_t = [X_t, Y_t, Z_t]^\top \in \mathbb{R}^3$ is the position at time t in the camera frame (meters). A stable motion-intent vector is obtained by differencing two adjacent, non-overlapping moving-average windows of

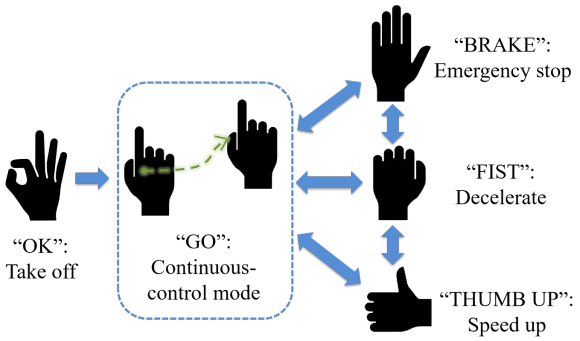

Fig. 3: Top-level discrete gesture definition

length m frames:

$$\begin{aligned} \bar{\mathbf{p}}_t^{(1)} &= \frac{1}{m} \sum_{i=t-m+1}^{t} \mathbf{p}_i, \\ \bar{\mathbf{p}}_t^{(0)} &= \frac{1}{m} \sum_{i=t-2m+1}^{t-m} \mathbf{p}_i, \\ \mathbf{e}_t &= \bar{\mathbf{p}}_t^{(1)} - \bar{\mathbf{p}}_t^{(0)}. \end{aligned} \tag{2}$$

where $\bar{\mathbf{p}}_t^{(1)}$ averages the most recent window, $\bar{\mathbf{p}}_t^{(0)}$ averages the preceding window, and $\mathbf{e}_t$ encodes the short-term displacement trend (direction) and motion strength (magnitude).

To suppress small hand jitters, we apply a symmetric deadband δ to each coordinate of the motion-intent vector $\mathbf{e}_t$:

$$\tilde{e}_{t,k} = \begin{cases} 0, & |e_{t,k}| < \delta, \\ e_{t,k} - \operatorname{sgn}(e_{t,k})\,\delta, & \text{otherwise}, \end{cases} \qquad k \in \{x, y, z\}. \tag{3}$$

We map the de-jittered intent vector to body-frame translational velocity and apply a speed multiplier s:

$$\begin{bmatrix} v_N \\ v_E \\ v_D \end{bmatrix} = \kappa s \begin{bmatrix} -\tilde{e}_{t,z} \\ \tilde{e}_{t,x} \\ \tilde{e}_{t,y} \end{bmatrix} \tag{4}$$

$$\mathbf{v}_t = (1 - \beta)\,\mathbf{v}_{t-1} + \beta\,\mathbf{v}_t^{\text{raw}}$$

where κ is a proportional gain, $\beta \in (0, 1]$ is a first-order low-pass coefficient. When Brake is triggered or $\|\tilde{\mathbf{e}}_t\|$ is below a small threshold, the commanded velocity is automatically zeroed.

b) Gaze channel: The gaze channel takes the head–eye yaw and pitch estimated by L2CS-Net [24]. After coordinate transforms and geometric projection, we obtain the gaze pixel (x_g, y_g). The yaw rate under the free-gaze policy is computed from the horizontal offset relative to the screen center x_{off}:

$$\omega_z = \operatorname{clip}\{\, k_\psi [x_{\text{off}} - \operatorname{sgn}(x_{\text{off}})\,\Delta_x]\,,\ -\omega_{\max},\ \omega_{\max}\}\,. \tag{5}$$

where k_ψ is the proportional gain, $\Delta_x > 0$ is the horizontal deadband width, $\operatorname{clip}(a, l, u)$ saturates a to $[l, u]$, and $\omega_{\max} > 0$ is the yaw-rate limit.

Once the gaze point remains inside a detected bounding box for more than 0.5 s, the system enters *target-lock* mode.

TABLE I: Sim-1 Racing Metrics (means). Paired within-subject. Unless noted, tests are paired t with 95% CI of the mean difference shown in "CI (95%)". Rows marked † use Wilcoxon signed-rank.

Group	Metric	IntuFly	RC	CI (95%)	p (Holm)	Effect
Intermed.	t (s)†	120.10	101.92	—	0.008	$r_{\mathrm{rb}} = 0.96$
Intermed.	l (m)	177.45	192.10	$[-29.74,\ 0.44]$	0.056	$d_z = -0.69$
Intermed.	v (m/s)	1.49	1.90	$[-0.55,\ -0.27]$	0.000	$d_z = -2.11$
Novice	t (s)	143.89	187.65	$[-78.03,\ -9.50]$	0.036	$d_z = -0.91$
Novice	l (m)	175.94	202.11	$[-43.49,\ -8.85]$	0.023	$d_z = -1.08$
Novice	v (m/s)	1.23	1.13	$[-0.13,\ 0.32]$	0.363	$d_z = 0.30$

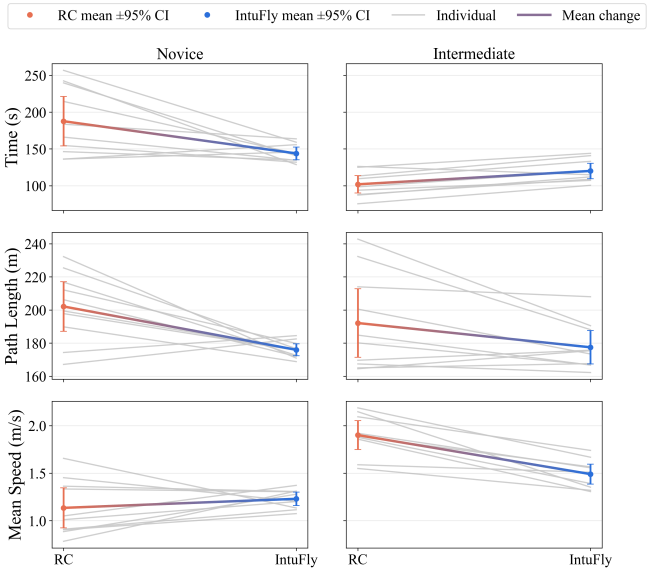

Fig. 4: The paired line graph of quantitative flight efficiency for the participants in Sim-1 using two control modes.

During lock, the yaw rate is computed from the target center's horizontal image offset Δx_{obj}:

$$\omega_z = k_{\mathrm{trk}}\ \mathrm{dz}\big(\Delta x_{\mathrm{obj}},\ c_0\big). \tag{6}$$

where k_{trk} is the proportional gain in tracking mode, $c_0 > 0$ is the deadband width, and $\mathrm{dz}(a, c)$ denotes the deadband function: it returns 0 if $|a| < c$, and $a - \mathrm{sgn}(a)\, c$ otherwise. The lock is automatically released if the gaze leaves the box for more than 2 s or the target disappears, after which the controller falls back to the free-gaze policy. This design assigns translation to the hand and heading/selection to the eyes, mitigating *Midas touch* and saccadic-noise issues of direct eye-driven velocity control while removing the need for explicit intent-then-mode-switch commands.

IV. Experiments and Results

A. Overall Setup

We evaluate IntuFly in both simulation and real-world experiments.

Simulation. Primary evaluation is conducted in Gazebo Sim with the PX4 firmware and the MAVSDK control interface. The YOLO [25] detector consumes the simulated onboard camera stream, with frames time- and frame-aligned

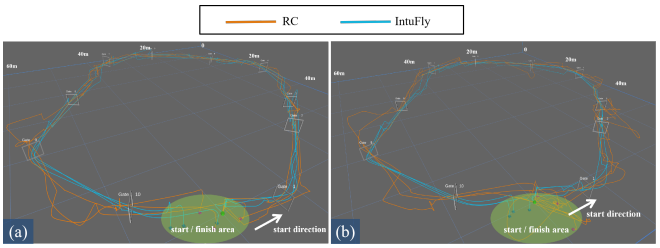

Fig. 5: (a) Typical flight trajectory of intermediate group. (b) Typical flight trajectory of novice group.

to the control loop to ensure closed-loop spatiotemporal consistency. All nodes run on ROS2 Humble.

Real-world. Hardware experiments use a DJI Tello UAV in our custom indoor test site, with the onboard camera video serving as the YOLO input. Aside from necessary sensor calibration, the perception and control stacks mirror the simulation configuration to assess sim-to-real transferability.

Sensing & compute. In both settings, an Intel RealSense D435 captures RGB-D of the user's hands and head, and model inference runs on a single NVIDIA RTX 4060 Ti (8 GB).

Ethics Statement. This human-subject study was reviewed and approved by the Academic Committee of Southeast University. All participants provided informed consent.

B. Sim-1: Racing

We compare IntuFly against a conventional RC baseline on a closed-course racetrack. The course comprises 10 race gates arranged in an approximately circular loop and is operated under no-wind conditions with no dynamic obstacles. To ensure fairness, both modalities use identical upper bounds and slew-rate limits on linear speed and yaw rate.

We stratified participants based on their cumulative UAV flight experience. The novice group ($n = 10$) reported < 4 h cumulative UAV flight experience per participant (mean ≈ 1.2 h), including $6/10$ with no prior UAV or FPV experience. The intermediate group ($n = 10$) reported > 25 h per participant (mean ≈ 29.7 h), with $5/10$ having prior FPV experience. Before formal testing, each participant completed a 20-min familiarization session with each control modality. Testing followed a within-subjects, counterbalanced order. For each modality, every participant flew three valid timed laps, with a 5-min rest between modalities to mitigate fatigue. All RC trials used an APEX4 handset. Timing started when the UAV left the launch platform, required sequential valid traversal of all gates, and stopped upon landing back on the platform.

The quantitative flight efficiency results of Sim-1 are presented in Table I. The paired connection diagram of the subject samples in Sim-1 is shown in Figure 4. Novices using IntuFly showed a significantly lower mean completion time, decreasing from 187.65 s (RC) to 143.89 s (-23.3%); mean path length decreased from 202.11 m to 175.94 m (-12.9%), and mean speed increased slightly from 1.13 m/s to 1.22 m/s ($+7.96\%$). For the intermediate cohort, results indicate a shorter-line but more-conservative-speed trade-off:

TABLE II: NASA-TLX subscales (0–100) and overall raw (means). All comparisons use paired t tests.

Scale	IntuFly	RC	CI (95%)	p (Holm)	Cohen d_z
Mental	30.50	70.71	$[-48.40, -32.02]$	1.06×10^{-8}	-2.24
Physical	46.25	52.86	$[-14.56, 1.35]$	0.099	-0.38
Temporal	47.50	55.95	$[-15.38, -1.53]$	0.039	-0.56
Performance	22.75	32.62	$[-15.59, -4.15]$	0.0054	-0.78
Effort	29.75	70.48	$[-48.91, -32.54]$	1.01×10^{-8}	-2.26
Frustration	37.50	59.76	$[-29.05, -15.47]$	4.79×10^{-6}	-1.49
Raw (mean)	35.77	57.06	$[-25.52, -17.07]$	9.58×10^{-9}	-2.29

TABLE III: UEQ six dimensions (means). All comparisons use paired t tests.

Scale	IntuFly	RC	CI (95%)	p (Holm)	Cohen d_z
Attractiveness	1.717	-0.45	[1.868, 2.465]	0.0	3.40
Dependability	1.425	-0.10	[1.149, 1.901]	0.0	1.90
Efficiency	1.462	-0.662	[1.691, 2.559]	0.0	2.29
Novelty	1.888	-1.60	[3.167, 3.808]	0.0	5.09
Perspicuity	1.412	-0.938	[1.844, 2.856]	0.0	2.17
Stimulation	1.375	-1.238	[2.138, 3.087]	0.0	2.58

mean path length with IntuFly was 177.45 m vs. 192.10 m with RC (-7.63%), but mean completion time increased from 101.93 s to 120.10 s ($+17.8\%$), and mean speed decreased from 1.90 m/s to 1.49 m/s (-23.16%). These findings suggest that IntuFly confers substantial benefits for novices—producing shorter and faster laps—whereas intermediates yielded better lines but slower times. This reflects our design priority to guarantee stability for untrained users, intentionally trading off the aggressive maneuverability of professional RC to lower the entry barrier. The typical flight trajectory diagrams for the novice group and the intermediate group are shown in Figure 5.

Tables II and Tables III respectively present the results of the NASA-TLX and UEQ user questionnaires. For subjective measures, NASA–TLX (Raw) decreased from 57.06 (RC) to 35.77 (IntuFly), with the largest reductions in Mental, Effort, and Frustration; SUS increased from 46.18 to 75.79. UEQ scores were higher across all six dimensions than RC, with large differences in magnitude. Collectively, relative to conventional RC, IntuFly markedly reduces cognitive and psychological workload, improves perceived perspicuity and operational efficiency, and exerts strong positive effects on hedonic qualities—particularly stimulation and novelty.

C. Sim-2: Target Tracking

We evaluate IntuFly against a Hand-only baseline on a mobile target tracking task to assess responsiveness and stability, thereby testing the benefit of adding the gaze channel. In each trial, participants pilot the UAV to follow a moving target observed through the simulated onboard camera. The target is a red sphere following a randomized trajectory with speed randomized in the 0.8–1.5 m/s range. YOLO detections are taken directly from the onboard video stream and are time- and frame-aligned with the control loop. Hand-only denotes an ablation of IntuFly with the gaze control channel disabled; all other processing stages, saturation bounds, and slew-rate limits are identical, and

TABLE IV: Sim-2 Target Tracking Metrics(means). All comparisons use paired t tests after normality check.

Metric	Hand-only	IntuFly	p	Cohen d_z
Latency (ms)	1259	556	2.71×10^{-6}	-3.27
LOS Coverage	0.866	0.996	3.54×10^{-6}	3.17

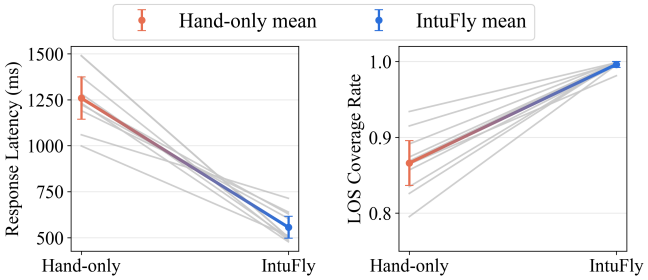

Fig. 6: Paired connection diagram of evaluation data for the subject tracking task in Sim-2.

yaw is commanded via the same hand-drawn trajectory loop triggered by the victory gesture (“peace” gesture with the index and middle fingers extended). Participants are the 10 intermediate users from Experiment B. Using a within-subjects, counterbalanced design, each participant performs 3 trials per modality, 4 min per trial, with 5 min rest between trials to mitigate fatigue.

Quantitative evaluation focuses on LOS coverage rate and response time. The LOS coverage rate lies in $[0, 1]$, where 1 indicates complete tracking coverage, and is computed as:

$$\mathrm{LOS} = 1 - \frac{T_{\mathrm{loss}}}{T_{\mathrm{total}}}, \tag{7}$$

where T_{loss} denotes the total time the target is lost (s) and T_{total} the length of the analysis window (s). Response time is defined as the interval between a change in the target’s motion direction within FoV and the onset of UAV yaw correction.

The evaluation results of Sim-2 are presented in Table IV. The paired connection diagram of participant evaluation data is shown in Figure 6. Compared with Hand-only, IntuFly reduces response latency from 1,259 ms to 556 ms , with all participants responding faster. LOS coverage increases from 86.6% (Hand-only) to 99.6% , with higher coverage for all participants. The distribution heatmap of the target in the field of view is shown in Figure 7. These results indicate that, under identical saturation and slew-rate constraints, IntuFly’s dual-channel fusion significantly reduces tracking latency and consistently keeps the target within FoV, thereby improving the continuity and precision of UAV control.

D. Real-World Validation

Real-world gate traversal. To assess sim-to-real transfer, we conducted real-hardware gate-traversal experiments with IntuFly on a custom indoor closed-loop course. The track forms a loop of five race gates of different heights. To mitigate variance from the short lap length, each trial comprised 10 consecutive laps, yielding a cumulative path

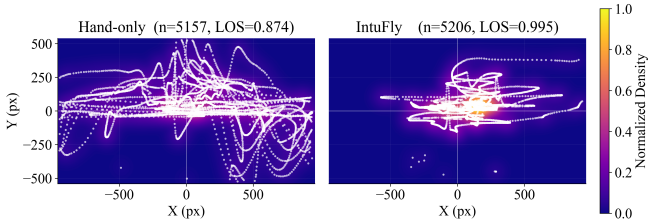

Fig. 7: Typical distribution heatmap of targets in the field of view under two control modes in Sim-2.

length about 180 m. The system maintained stable closed-loop operation at more than 25 Hz in the real environment and the end-to-end latency is 102 ms. Across five multi-lap trials, the accident-free completion rate was 92%. The mean normalized pace was 0.89 m/s, deviating by -0.46 m/s from the simulation mean; differences are primarily attributable to indoor recirculating airflow, ground effect, and sensor noise, which induce more conservative turning. Figure 8 shows a representative single-lap stack plot, with no sustained oscillations or missed gates.

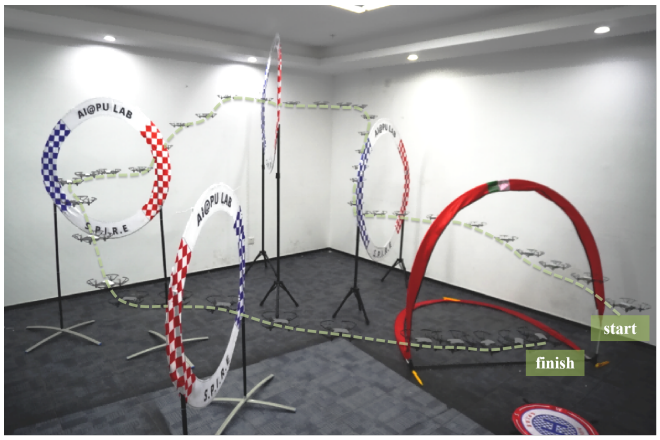

Fig. 8: Stack plot of typical flight trajectories in DJI Tello experiments

V. CONCLUSIONS

We present IntuFly, an intuition-driven natural UAV interaction paradigm that fuses hand-drawn trajectories with gaze-based yaw and target locking. Continuous hand trajectories command translational motion, while gaze governs yaw and target lock; once locked, the system autonomously regulates heading—realizing a "hand draws the path, eyes set the heading" division of labor that lowers operator burden and entry barriers while preserving continuous controllability.

Simulation results show that, in a racing task, novices achieve substantially shorter lap times and path lengths with IntuFly; intermediates exhibit shorter, smoother lines but more conservative speeds, reflecting a stability–risk trade-off. Subjective measures (Raw NASA-TLX, SUS, UEQ) likewise indicate reduced mental workload and improved usability. In a target-tracking task, the system yields faster response, smoother trajectories, and higher LOS coverage. Real-world five-gate closed-loop tests demonstrate that IntuFly runs stably at more than 25 Hz with behavior closely consistent with simulation, validating sim-to-real feasibility.

Future work will extend to more complex scenarios and explore integrating other modalities (e.g., voice) to further enhance robustness against visual distractions across broader robotic applications.

REFERENCES

[1] S. Abdalla and S. Baidya, "Uav control with vision-based hand gesture recognition over edge-computing," in *2025 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2025, pp. 481–488.

[2] J. R. Cauchard, M. Khamis, J. Garcia, M. Kljun, and A. M. Brock, "Toward a roadmap for human-drone interaction," *Interactions*, vol. 28, no. 2, pp. 76–81, 2021.

[3] V. Herdel, L. J. Yamin, and J. R. Cauchard, "Above and beyond: A scoping review of domains and applications for human-drone interaction," in *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 2022, pp. 1–22.

[4] J. Akagi, T. D. Morris, B. Moon, X. Chen, and C. K. Peterson, "Gesture commands for controlling high-level uav behavior," *SN Applied Sciences*, vol. 3, no. 6, p. 603, 2021.

[5] G. Yun, H. Kwak, and D. H. Kim, "Single-handed gesture recognition with rgb camera for drone motion control," *Applied Sciences*, vol. 14, no. 22, p. 10230, 2024.

[6] M. Parisay, C. Poullis, and M. Kersten, "Eyetap: A novel technique using voice inputs to address the midas touch problem for gaze-based interactions," *arXiv preprint arXiv:2002.08455*, 2020.

[7] P. K. BN, A. Balasubramanyam, A. K. Patil, C. B, and Y. H. Chai, "Gazeguide: An eye-gaze-guided active immersive uav camera," *Applied Sciences*, vol. 10, no. 5, p. 1668, 2020.

[8] Q. Wang, B. He, Z. Xun, C. Xu, and F. Gao, "Gpa-teleoperation: Gaze enhanced perception-aware safe assistive aerial teleoperation," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5631–5638, 2022.

[9] S. G. Hart and L. E. Staveland, "Development of nasa-tlx (task load index): Results of empirical and theoretical research," in *Advances in psychology*. Elsevier, 1988, vol. 52, pp. 139–183.

[10] J. Brooke *et al.*, "Sus-a quick and dirty usability scale," *Usability evaluation in industry*, vol. 189, no. 194, pp. 4–7, 1996.

[11] B. Laugwitz, T. Held, and M. Schrepp, "Construction and evaluation of a user experience questionnaire," in *Symposium of the Austrian HCI and usability engineering group*. Springer, 2008, pp. 63–76.

[12] B. Pfleging, S. Schneegass, and A. Schmidt, "Multimodal interaction in the car: combining speech and gestures on the steering wheel," in *Proceedings of the 4th international conference on automotive user interfaces and interactive vehicular applications*, 2012, pp. 155–162.

[13] D. Liu, J. Kim, and Y. Ham, "Multi-user immersive environment for excavator teleoperation in construction," *Automation in Construction*, vol. 156, p. 105143, 2023.

[14] C. Rognon, S. Mintchev, F. Dell'Agnola, A. Cherpillod, D. Atienza, and D. Floreano, "Flyjacket: An upper body soft exoskeleton for immersive drone control," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2362–2369, 2018.

[15] J. R. Cauchard, A. Tamkin, C. Y. Wang, L. Vink, M. Park, T. Fang, and J. A. Landay, "Drone. io: A gestural and visual interface for human-drone interaction," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 2019, pp. 153–162.

[16] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.-L. Chang, and M. Grundmann, "Mediapipe hands: On-device real-time hand tracking," *arXiv preprint arXiv:2006.10214*, 2020.

[17] G. Sung, K. Sokal, E. Uboweja, V. Bazarevsky, J. Baccash, E. G. Bazavan, C.-L. Chang, and M. Grundmann, "On-device real-time hand gesture recognition," *arXiv preprint arXiv:2111.00038*, 2021.

[18] M. Khamis, A. Kienle, F. Alt, and A. Bulling, "Gazedrone: Mobile eye-based interaction in public space without augmenting the user," in *Proceedings of the 4th ACM Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*, 2018, pp. 66–71.

[19] M. Yu, Y. Lin, X. Wang, D. Schmidt, and Y. Wang, "Human-robot interaction based on gaze gestures for the drone teleoperation," *Journal of Eye Movement Research*, vol. 7, no. 4, pp. 1–14, 2014.

[20] A. Zhou, L. Han, and Y. Meng, "Multimodal control of uav based on gesture, eye movement and voice interaction," in *International Conference on Guidance, Navigation and Control*. Springer, 2022, pp. 3765–3774.

[21] K. Pfeuffer, J. Alexander, M. K. Chong, and H. Gellersen, "Gaze-touch: combining gaze with multi-touch for interaction on the same surface," in *Proceedings of the 27th annual ACM symposium on User interface software and technology*, 2014, pp. 509–518.

[22] M. N. Lystbæk, P. Rosenberg, K. Pfeuffer, J. E. Grønbæk, and H. Gellersen, "Gaze-hand alignment: Combining eye gaze and mid-air pointing for interacting with menus in augmented reality," *Proceedings of the ACM on Human-Computer Interaction*, vol. 6, no. ETRA, pp. 1–18, 2022.

[23] M. Di Vincenzo, F. Palini, M. De Marsico, A. M. Borghi, and G. Baldassarre, "A natural human-drone embodied interface: Empirical comparison with a traditional interface," *Frontiers in Neurorobotics*, vol. 16, p. 898859, 2022.

[24] A. A. Abdelrahman, T. Hempel, A. Khalifa, A. Al-Hamadi, and L. Dinges, "L2cs-net: Fine-grained gaze estimation in unconstrained environments," in *2023 8th International Conference on Frontiers of Signal Processing (ICFSP)*. IEEE, 2023, pp. 98–102.

[25] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.