Reliability-Budgeted Edge-Cloud Self-Adaptation for Continual Multimodal UAV Vision

Junwei Zhao[1], **Qianchun Luo**[2], **Jie Wu**[1,3]
[1]China Telecom Cloud Computing Research Institute
[2]China Telecom Unmanned Technology Co., Ltd. [3]Temple University
jwz@pku.edu.cn, sqlqc@189.cn, wujie@chinatelecom.cn

Abstract

Continual multimodal UAV vision requires visual models to remain reliable as atmospheric conditions evolve during long-duration flights. Focusing on haze-induced degradation, we formulate UAV dehazing as a closed-loop edge-cloud self-adaptation problem with an explicit reliability budget for per-run false-upload control. The onboard model couples a frozen anchor head with a cloud-adaptive head, evaluated by a joint critic integrating a physics-based rehaze constraint and a scale-aligned infrared-edge consistency term to suppress degenerate adaptations. Our main contribution is a sequential upload criterion based on a non-negative e-process, yielding per-run false-upload control under conditional calibration without requiring frame independence or exchangeability. The cloud updates only the adaptive head from uploaded segments, while the UAV accepts or rejects returned parameters through a shadow-window sign test with an acceptance budget. Experiments on five UAV datasets show consistent improvements under haze shifts.

1 Introduction

During continuous UAV flights, haze density and its spatial distribution change along the trajectory [1, 2], inducing domain shift that degrades the performance of edge-side dehazing models (Fig. 1). This motivates adaptation under a drifting test distribution without ground-truth supervision. Test-Time Adaptation (TTA) and its variants update a pretrained model online from unlabeled test data, typically through repeated gradient steps during inference [3]. Existing TTA work centers on recognition tasks [4], while long-horizon online updates may accumulate drift and require explicit stability control [5]. Pixel-level restoration under continuously changing multimodal haze has not been sufficiently studied in current TTA benchmarks.

Image dehazing has been formulated as a pixel-level restoration problem spanning physics-inspired priors and deep networks [6, 7, 8, 9]. Beyond single-modality dehazing, recent VIS-IR methods exploit cross-modal fusion to recover structure under dense haze, including VIFNet [10] and HDCFN [11]. These methods use the lower scattering sensitivity of thermal infrared compared with visible light (Fig. 2), but they follow an offline-training protocol and apply a fixed forward pass at deployment [7]. They do not adapt to time-varying haze, and the compute budget of edge devices constrains the form of any online update [12].

Edge-cloud collaboration provides system-level mechanisms for jointly executing and updating models across resource-limited devices and cloud servers [13]. Prior work focuses on computation partitioning, offloading, split inference, and communication-efficient updates, with triggers usually defined by latency, bandwidth, or energy [14, 15]. Pixel-level restoration is not the primary target, and existing triggers do not control the long-run false-upload probability under nominal conditions. For VIS-IR dehazing, the design of a device-side trigger that decides when adaptation is warranted under time-varying haze remains open.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

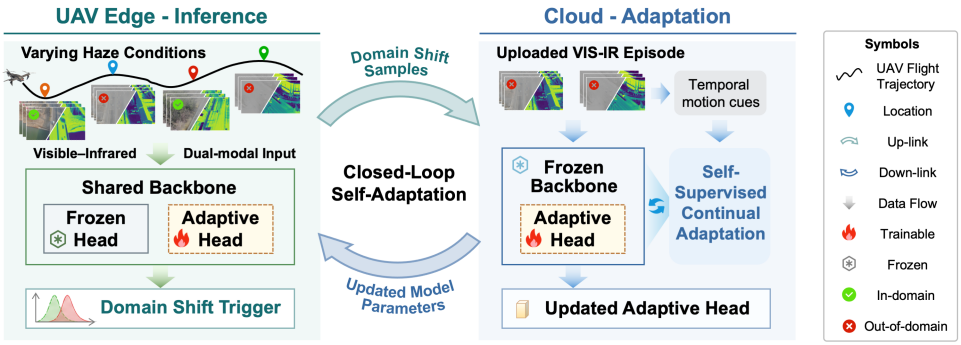


Figure 1: VIS-IR UAV dehazing under varying haze in an edge-cloud self-adaptation loop. The edge decides when to invoke adaptation through a sequential test, and the cloud performs parameter-isolated self-supervised updates on the uploaded episode and returns updated parameters.

To this end, we propose an edge-cloud self-adaptation framework for continual VIS-IR UAV dehazing under drifting haze. The framework comprises edge-side inference with sequential-test triggering and cloud-side parameter-isolated adaptation. On the edge, a shared-trunk dual-head model uses a frozen anchor head and an adaptive head updated by the cloud. A joint critic combines a physics-grounded rehaze term with a scale-aligned IR-edge term, and the critic margin drives a sigmoid-gated mixture between the two heads. Triggering is posed as a sequential test on a non-negative e-process built from two drift indicators. The trigger has a false-trigger probability at most α on every monitoring run, provided the per-frame bets satisfy conditional calibration. A slack correction estimated on contiguous nominal clips uses binned conditional means of upper-tail bets. The guarantee assumes conditional means at most one for the corrected bets given the deployment history. On the cloud, the trunk and anchor head are frozen, and only the adaptive head is updated. The returned parameters are selected on a held-out validation split of the uploaded episode and accepted by a shadow-window sign test with budget ρ.

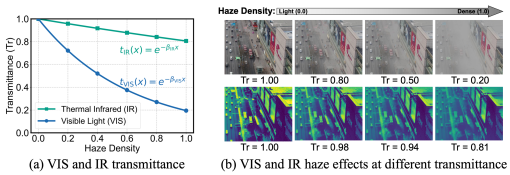
We evaluate the framework on VTUAV, DroneVehicle, CART, and M3FD with synthesized drifting haze, and on CityUAV with naturally occurring haze. The model attains higher PSNR/SSIM and lower LPIPS than prior dehazers and four advanced TTA baselines. With 5.19M parameters and 284.1 GFLOPs, the on-device model runs at 39.2 ms per frame on Jetson Orin Nano. Under the heaviest drift state, the closed loop improves PSNR by 1.61 dB over the no-adaptation variant and by 1.77 dB over the strongest TTA baseline, while empirical false-trigger rates estimated over nominal monitoring runs remain below the corresponding budgets.

Our contributions are summarized as follows.

- We provide a theoretical upload-triggering criterion for continual UAV dehazing. The proposed slack-calibrated e-process turns cloud invocation from heuristic thresholding into a sequential test, yielding per-run false-upload control under conditional betting calibration.
- We develop a self-supervised VIS-IR critic for ground-truth-free dehazing adaptation. The critic combines confidence-weighted physical rehazing with an internally scale-aligned IR-edge constraint, reducing the rehaze-consistent degenerate solution family and supplying a unified signal for head mixing, cloud-side learning, and update validation.
- We design a reliability-budgeted edge-cloud self-adaptation loop built around a frozen anchor head and a cloud-adaptive head. Parameter-isolated cloud updates, conservative interpolation, shadow-window validation, and rollback enable online adaptation while preserving an anchor-guided fallback path during long-horizon UAV deployment.

2 Related Work

As summarized in Tab. 1, our formulation lies at the intersection of three research lines. Existing dehazers do not trigger updates after deployment. Edge-cloud systems trigger by system objectives and do not target pixel-level restoration. TTA pipelines update without ground truth but usually rely on heuristic thresholds. We instead pose upload triggering as a sequential test with Type-I control under conditional calibration.



(a) VIS and IR transmittance (b) VIS and IR haze effects at different transmittance

Figure 2: Transmittance decay and VIS/IR comparison under haze. (a) Transmittance decay comparing $e^{-\beta_{\mathrm{IR}}x}$ and $e^{-\beta_{\mathrm{VIS}}x}$ with $\beta_{\mathrm{IR}} \ll \beta_{\mathrm{VIS}}$. (b) VIS and IR images under different haze densities, where IR preserves structure better as haze increases.

Research Line	Example Ref.	VI	PixR	Trig	noGT
Image Dehazing	[Zhao *et al.*, 2025]	✓	✓	×	×
Edge-Cloud Colla.	[Chen *et al.*, 2025]	✓	×	✓	×
Test-Time Adapt.	[Liu *et al.*, 2025]	×	✓	×	✓
Ours	–	✓	✓	✓	✓

Table 1: Comparison with related research lines. VI denotes visible-infrared input. PixR denotes pixel-level restoration. Trig denotes device-side adaptation triggering. noGT denotes adaptation without ground truth.

2.1 Image Dehazing

Image dehazing is typically formulated as pixel-level restoration from a single visible input, ranging from prior-based models to deep networks trained on synthetic degradations and hybrid designs that combine physical cues with learned features [16, 17, 18, 19, 20]. Beyond single-modality dehazing, recent multi-modal restoration exploits complementary sensors such as VIS-IR to improve structure recovery under severe haze [10, 11, 21]. Existing methods are trained offline and applied with fixed inference, without an explicit mechanism to trigger adaptation under continuously varying haze [7].

2.2 Edge-Cloud Collaboration

Edge-cloud collaboration explores how to distribute computation and data between resource-limited devices and cloud servers, covering offloading policies [22, 23, 24], split inference [25, 26], and collaborative training [27]. Many systems incorporate selective transmission, but triggers are usually defined by latency, energy, bandwidth, or uncertainty for upload scheduling [28, 29]. Our work uses edge-cloud collaboration for pixel-level restoration in changing haze, where the cloud performs self-supervised adaptation and improves the edge model's robustness to domain shift.

2.3 Test-Time Adaptation

TTA and continual adaptation mitigate distribution shift by updating model parameters on unlabeled target data at inference time [30, 31, 32]. Common self-supervised signals include entropy minimization [33] and consistency regularization [34]. Recent restoration-oriented variants use degradation adaptation or dynamic shift detection [35, 36]. These methods are valuable baselines, yet they perform local or persistent updates without an explicit edge-cloud upload budget and without rollback through a frozen anchor. Our framework differs by criterion-triggered adaptation, parameter isolation, and false-upload control under conditional calibration.

3 Method

3.1 Problem Formulation

A UAV captures a continuous VIS-IR stream $\mathcal{X} = \{(x_t^{\mathrm{vis}}, x_t^{\mathrm{ir}})\}_{t\geq 1}$, with $x_t^{\mathrm{vis}} \in \mathbb{R}^{H\times W\times 3}$ and $x_t^{\mathrm{ir}} \in \mathbb{R}^{H\times W\times 1}$. The visible image follows the time-indexed scattering model

$$x_t^{\mathrm{vis}} = J_t \odot T_t + (1 - T_t) \odot A_t, \qquad T_t = \exp(-\beta_t d_t). \tag{1}$$

The clear image J_t, transmittance T_t, depth d_t, atmospheric light A_t, and haze coefficient β_t are unobserved at deployment and drift along the flight trajectory. We seek an on-device dehazer that adapts to this drift without ground truth, under explicit control of false uploads.

The framework couples a single trunk with a frozen anchor head Θ_J^{fix} and a cloud-updatable adaptive head Θ_J^{adp}, as shown in Fig. 3 and Algorithm 1. At each frame, the device scores both heads with a joint critic, emits a safe-gated mixture, and updates a non-negative e-process on two scalar drift indicators. A cloud upload is triggered at a stopping time controlled by α. The cloud freezes the trunk and anchor head, updates only Θ_J^{adp} on the training split of the uploaded episode, and returns a conservatively interpolated parameter vector chosen on a held-out validation split. The device accepts or rejects the update through a shadow-window sign test with budget ρ.

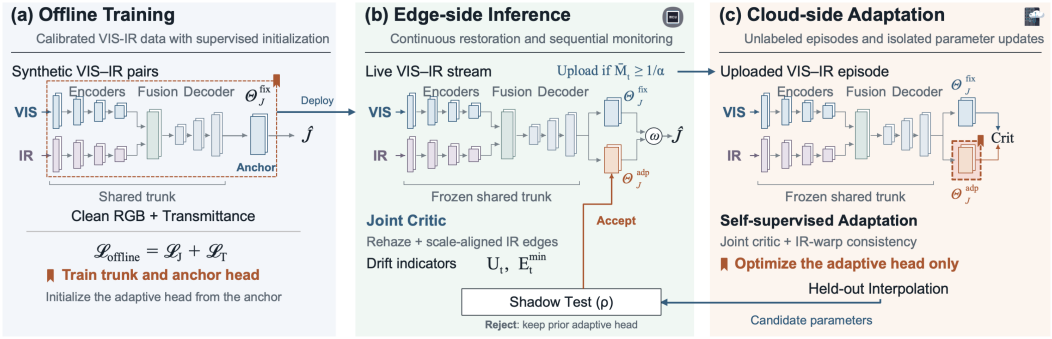


Figure 3: Overview of the edge-cloud self-adaptation framework for continual VIS-IR dehazing. (a) Offline training fits the trunk and frozen anchor head. (b) On-device inference scores both heads with a joint critic, emits a safe-gated mixture, and monitors an e-process at budget α. (c) The cloud updates only the adaptive head on the training split, returns conservatively interpolated parameters, and a shadow-window sign test at budget ρ governs acceptance or rollback.

Two design choices distinguish this framework from prior TTA [35, 36, 38] and edge-cloud adaptation systems [39, 40]. First, upload triggering is posed as a sequential test with per-run Type-I control under conditional calibration. Second, the self-supervised critic combines physically grounded rehazing with scale-aligned IR-edge consistency, reducing a rehaze-consistent but edge-inconsistent family of degenerate solutions.

3.2 Network and Offline Training

Architecture. The trunk consists of paired VIS and IR encoders at L scales, a heteroscedastic haze-state branch that outputs a per-pixel transmittance proxy $\hat{T}_t \in (0,1)$ and confidence surrogate $C_t \in (0,1)$, a confidence-guided VIS-IR fusion module, and a shared decoder. Two residual heads, Θ_J^{fix} and Θ_J^{adp}, decode the fused features to $\hat{J}_t^{\text{fix}}$ and $\hat{J}_t^{\text{adp}}$. Detailed operators are in Appendix A.

Training objective. The trunk and Θ_J^{fix} are trained on synthetic tuples $(I^{\text{vis}}, I^{\text{ir}}, J^{\text{vis}}, T)$ generated by the depth-assisted scattering model [41], by minimizing $\mathcal{L}_{\text{offline}} = \mathcal{L}_J + \mathcal{L}_T$. Here $\mathcal{L}_J$ is an ℓ_1 restoration loss on $\hat{J}^{\text{fix}}$ and $\mathcal{L}_T$ is a heteroscedastic regression loss on $\hat{T}$ (Appendix B). After offline training, $(\Theta_{\text{sh}}, \Theta_J^{\text{fix}})$ are frozen, and Θ_J^{adp} is initialized as Θ_J^{fix}.

3.3 Joint Self-Supervised Critic

For a candidate restoration $\hat{J}_t$, the rehaze term solves for atmospheric light by confidence-weighted ridge regression (Appendix C), re-synthesizes $\widetilde{x}_t^{\text{vis}}(\hat{J}_t)$, and measures

$$\text{Crit}_{\text{rh}}(\hat{J}_t) = \frac{\text{mean}\big(|x_t^{\text{vis}} - \widetilde{x}_t^{\text{vis}}(\hat{J}_t)| \odot C_t\big)}{\text{mean}(C_t) + \epsilon}. \tag{2}$$

The scattering equation alone does not identify J_t. A restoration $\hat{J} = \lambda J^* + (1-\lambda)A^*$ paired with rescaled transmittance can exactly reproduce x^{vis} while $\hat{J} \neq J^*$ (Appendix D).

To provide an additional structural constraint, we use a scale-aligned IR-edge term. Let a_t^{ir} be a detached gradient-alignment factor estimated from the frozen head and the IR image (Appendix E). We define

$$\text{Crit}_{\text{ir}}(\hat{J}_t) = \text{mean}\big(|\nabla \hat{J}_t - a_t^{\text{ir}} \nabla x_t^{\text{ir}}| \odot M_t^{\text{ir}}\big), \tag{3}$$

where ∇ is a fixed Sobel operator and M_t^{ir} is computed from x_t^{ir} alone. The alignment factor is applied inside the norm, so it corrects the VIS-IR gradient scale rather than merely reweighting the loss magnitude.

The joint critic is

$$\text{Crit}(\hat{J}_t) = \text{Crit}_{\text{rh}}(\hat{J}_t) + \lambda_{\text{ir}} \text{Crit}_{\text{ir}}(\hat{J}_t). \tag{4}$$

It is used for head mixing, triggering, cloud adaptation, and acceptance testing. To reduce shortcut adaptation, $(\hat{T}_t, C_t, A_t, M_t^{\text{ir}}, a_t^{\text{ir}})$ are detached in all gradient-bearing usages.

Algorithm 1 Edge-cloud self-adaptation for continual VIS-IR dehazing

Require: Stream $\mathcal{X}$, calibration fold $\mathcal{D}_0^{\mathrm{cal}}$, budgets α, ρ, episode length T, window length W
Ensure: Output $\hat{J}_t^{\mathrm{out}}$ and adaptive head Θ_J^{adp}

Offline train $(\Theta_{\mathrm{sh}}, \Theta_J^{\mathrm{fix}})$ via $\mathcal{L}_{\mathrm{offline}}$, set $\Theta_J^{\mathrm{adp}} \leftarrow \Theta_J^{\mathrm{fix}}$
Calibrate fit b_U, b_E on $\mathcal{D}_0^{\mathrm{cal}}$, compute k_W from ρ, set $\bar{M}_0 \leftarrow 1$
for $t = 1, 2, \ldots$ **do**
    Compute $E_t^{\mathrm{fix}}, E_t^{\mathrm{adp}}$ and output $\hat{J}_t^{\mathrm{out}}$ via Eq. (5)
    Update $\bar{M}_t \leftarrow \bar{M}_{t-1}\bar{e}_t$ from $(U_t, E_t^{\mathrm{min}})$
    if $\bar{M}_t \geq 1/\alpha$ **then**
        Split uploaded episode into $\mathcal{X}_{\mathrm{tr}}$ and $\mathcal{X}_{\mathrm{val}}$
        Cloud trains $\Theta_J^{\mathrm{adp,tr}}$ on $\mathcal{X}_{\mathrm{tr}}$ and returns $\Theta_J^{\mathrm{adp},\prime}$ selected on $\mathcal{X}_{\mathrm{val}}$
        Run shadow window $\mathcal{W}$ and apply Eq. (11)
        Accept $\Theta_J^{\mathrm{adp}} \leftarrow \Theta_J^{\mathrm{adp},\prime}$ if the test passes, otherwise rollback to $\Theta_J^{\mathrm{adp},0}$
        Reset $\bar{M}_t \leftarrow 1$ ▷ reset is unconditional, applied after both accept and rollback
    end if
end for

3.4 Edge-Side Inference and Sequential Triggering

For each frame, the device evaluates $E_t^h = \mathrm{Crit}(\hat{J}_t^h)$, $h \in \{\mathrm{fix}, \mathrm{adp}\}$. It emits a restoration through safe-gated mixing and updates an e-process for upload triggering.

Output mixture. The deployed restoration is a sigmoid-gated mixture

$$\hat{J}_t^{\mathrm{out}} = \hat{J}_t^{\mathrm{fix}} + \omega_t(\hat{J}_t^{\mathrm{adp}} - \hat{J}_t^{\mathrm{fix}}), \qquad \omega_t = \sigma\left(\frac{E_t^{\mathrm{fix}} - E_t^{\mathrm{adp}}}{\tau_m}\right), \tag{5}$$

with σ the sigmoid and temperature $\tau_m = 0.05$. The gate gives the adaptive head more than half of the weight only when $E_t^{\mathrm{adp}} < E_t^{\mathrm{fix}}$, and at most half otherwise. The mapping is continuous in the critic margin, which avoids switching artifacts at $E_t^{\mathrm{fix}} \approx E_t^{\mathrm{adp}}$.

Trigger critic. We monitor haze uncertainty $U_t = \mathrm{mean}_p((1 - C_{t,l^\star}(p))^2)$ and best-head error $E_t^{\mathrm{min}} = \min(E_t^{\mathrm{fix}}, E_t^{\mathrm{adp}})$. A trigger fires when their joint evidence exceeds a calibrated budget.

E-process construction. On a nominal calibration fold $\mathcal{D}_0^{\mathrm{cal}}$, we construct bounded non-negative upper-tail bets. We scale these bets using slack factors estimated from nominal contiguous clips to obtain b_U and b_E. Theorem 1 establishes false-upload control under H_0 when $\mathbb{E}[b_U(U_t) \mid \mathcal{F}_{t-1}] \leq 1$ and $\mathbb{E}[b_E(E_t^{\mathrm{min}}) \mid \mathcal{F}_{t-1}] \leq 1$. These conditional calibration assumptions allow temporally dependent frames. Appendix F details the calibration procedure and discusses calibration uncertainty. The per-frame bet is

$$\bar{e}_t = \tfrac{1}{2}\big(b_U(U_t) + b_E(E_t^{\mathrm{min}})\big), \qquad \bar{M}_t = \prod_{s=1}^{t} \bar{e}_s, \qquad \bar{M}_0 = 1. \tag{6}$$

The trigger fires at

$$\tau = \inf\{t \geq 1 : \bar{M}_t \geq 1/\alpha\}. \tag{7}$$

False-trigger control.

Theorem 1 (False-trigger control under conditional calibration)**.** *Within one monitoring run, let $\mathcal{F}_t$ be the filtration generated by the offline quantities and deployment information available through frame t. Under H_0, assume for every t that $\mathbb{E}[b_U(U_t) \mid \mathcal{F}_{t-1}] \leq 1$ and $\mathbb{E}[b_E(E_t^{\mathrm{min}}) \mid \mathcal{F}_{t-1}] \leq 1$. Then $\bar{M}_t$ in Eq.* (6) *is a non-negative $(\mathcal{F}_t)$-supermartingale with $\bar{M}_0 = 1$, and the stopping rule in Eq.* (7) *satisfies*

$$\Pr(\tau < \infty \mid H_0) \leq \alpha. \tag{8}$$

The result does not require frame independence, exchangeability, or stationarity beyond the stated conditional calibration condition.

Method	Pub.	Year	Mod.	VTUAV			DroneV			CART			M3FD			Complexity	
				PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	Params	GFLOPs
SFSNiD	CVPR	2024	VIS	25.31	0.874	0.401	18.58	0.663	0.554	20.46	0.794	0.428	19.35	0.726	0.498	8.35M	557.6
ConvIR	T-PAMI	2024	VIS	26.47	0.878	0.382	18.96	0.674	0.530	21.27	0.798	0.415	20.28	0.739	0.468	8.64M	375.3
HazeFlow	ICCV	2025	VIS	27.25	0.885	0.348	19.47	0.683	0.508	21.53	0.812	0.385	20.41	0.745	0.452	50.86M	915.4
VIFNet	Neurocomp.	2024	VIS+IR	28.11	0.889	0.325	20.48	0.702	0.457	22.41	0.819	0.362	21.63	0.765	0.404	9.78M	708.5
HDCFN	ACM MM	2025	VIS+IR	28.43	0.891	0.307	21.05	0.721	0.425	22.77	0.835	0.348	21.85	0.780	0.392	6.83M	305.2
Ours (anchor only)		2026	VIS+IR	**29.27**	**0.899**	**0.292**	**22.27**	**0.739**	**0.411**	**23.15**	**0.852**	**0.334**	**22.29**	**0.791**	**0.376**	**5.19M**	**284.1**

Table 2: Quantitative comparison on four UAV datasets and model complexity. The table reports static dehazing performance through the frozen anchor head, without cloud adaptation.

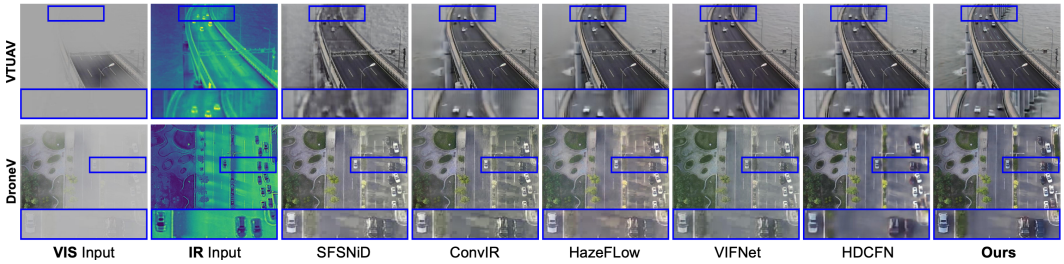


Figure 4: Representative visual results on VTUAV and DroneV datasets.

The proof is in Appendix G. H_0 permits temporally correlated video when past frames do not predict excess upper-tail drift after the slack correction. If nominal correlation makes the next frame predictably more upper-tailed than the slack absorbs, H_0 is violated.

Episode upload. At $t_0 = \tau$, the device uploads $\mathcal{X}_{t_0:t_0+T-1}$ with telemetry $\{U_t, E_t^{\mathrm{fix}}, E_t^{\mathrm{adp}}, \omega_t\}$, suspends further triggers, and resets the e-process. Monitoring resumes after the acceptance decision.

3.5 Cloud-Side Adaptation and Acceptance

Adaptation objective. The cloud freezes $(\Theta_{\mathrm{sh}}, \Theta_J^{\mathrm{fix}})$ and updates only Θ_J^{adp}. The uploaded episode is split in temporal order at $n_{\mathrm{tr}} = \lfloor 0.8T \rfloor$, with the first n_{tr} frames forming $\mathcal{X}_{\mathrm{tr}}$ and the remaining $T - n_{\mathrm{tr}}$ frames forming $\mathcal{X}_{\mathrm{val}}$. Training uses only $\mathcal{X}_{\mathrm{tr}}$, so the validation split is genuinely held out from gradient updates. With per-frame averages $\bar{\mathcal{L}}_{\mathrm{rehaze}}$, computed by averaging Eq. (4) over $\mathcal{X}_{\mathrm{tr}}$, and $\bar{\mathcal{L}}_{\mathrm{temp}}$, an IR-warp consistency loss between consecutive training frames, the cloud minimizes the Laplacian uncertainty-weighted objective

$$\mathcal{L}_{\mathrm{cloud}} = \frac{\bar{\mathcal{L}}_{\mathrm{rehaze}}}{\sigma_1} + \frac{\bar{\mathcal{L}}_{\mathrm{temp}}}{\sigma_2} + \log(\sigma_1 \sigma_2), \tag{9}$$

where $\sigma_1, \sigma_2 > 0$ are learnable task scales. No gradients propagate through the correspondence estimator or through $(\hat{T}_t, C_t, M_t^{\mathrm{ir}}, a_t^{\mathrm{ir}})$. The Laplacian form matches the absolute residuals used by both losses. Full details are in Appendix H.

Conservative interpolation. Let $\Theta_J^{\mathrm{adp},0}$ denote the deployed parameters and $\Theta_J^{\mathrm{adp,tr}}$ the trained parameters. The cloud returns

$$\Theta_J^{\mathrm{adp},\prime} = (1-\gamma^\star)\Theta_J^{\mathrm{adp},0} + \gamma^\star \Theta_J^{\mathrm{adp,tr}}, \quad \gamma^\star = \min\left(\arg\min_{\gamma \in \Gamma} \mathcal{L}_{\mathrm{cloud}}\big(\Theta_J^{\mathrm{adp}}(\gamma); \mathcal{X}_{\mathrm{val}}\big)\right). \tag{10}$$

Here $\Gamma \subset [0,1]$ is a finite grid containing 0 and 1. All candidates use the same fixed trained task scales. If no positive grid point improves over $\gamma = 0$, the cloud returns $\Theta_J^{\mathrm{adp},0}$.

Acceptance and rollback. The device evaluates $\Theta_J^{\mathrm{adp},\prime}$ alongside $\Theta_J^{\mathrm{adp},0}$ over a shadow window $\mathcal{W}$ of length W, recording $d_t = E_t^{\mathrm{adp,new}} - E_t^{\mathrm{adp,old}}$. The update is accepted if

$$\sum_{t \in \mathcal{W}} \mathbf{1}[d_t < 0] \geq k_W, \qquad k_W = \min\Big\{k \in \{0,\dots,W\} : \sum_{j=k}^{W} \binom{W}{j} 2^{-W} \leq \rho\Big\}. \tag{11}$$

Method	PSNR ↑					SSIM ↑					LPIPS ↓				
	S1	S2	S3	S4	S5	S1	S2	S3	S4	S5	S1	S2	S3	S4	S5
SFSNiD [46]	26.57	25.06	22.61	20.02	17.18	0.883	0.870	0.832	0.782	0.715	0.376	0.414	0.492	0.591	0.693
ConvIR [47]	27.83	26.20	23.62	21.03	18.05	0.889	0.875	0.840	0.795	0.726	0.362	0.391	0.468	0.564	0.663
HazeFlow [48]	28.16	26.93	24.41	21.82	19.04	0.892	0.883	0.848	0.807	0.745	0.335	0.367	0.440	0.532	0.637
VIFNet [10]	28.91	27.92	25.78	23.62	20.86	0.896	0.885	0.858	0.829	0.783	0.304	0.330	0.402	0.495	0.587
HDCFN [11]	29.14	28.21	26.36	24.27	21.64	0.899	0.885	0.867	0.834	0.792	0.289	0.314	0.384	0.475	0.563
HDCFN+EntMin [33]	29.18	28.16	26.05	23.41	19.73	0.899	0.882	0.860	0.811	0.731	0.288	0.319	0.395	0.504	0.628
HDCFN+ConsReg [34]	29.21	28.27	26.18	23.85	20.48	0.900	0.884	0.863	0.819	0.756	0.286	0.315	0.391	0.494	0.607
HDCFN+TAO [35]	29.30	28.35	26.42	24.35	21.94	0.900	0.887	0.868	0.836	0.799	0.287	0.314	0.382	0.470	0.547
HDCFN+Hybrid-TTA [36]	29.36	28.48	26.61	24.55	22.18	0.901	0.889	0.871	0.840	0.806	0.284	0.310	0.376	0.461	0.535
Ours (anchor only)	29.47	28.63	26.82	24.88	22.34	0.902	0.892	0.869	0.844	0.811	0.283	0.305	0.374	0.460	0.524
Ours (full loop)	**30.18**	**29.13**	**27.52**	**25.44**	**23.95**	**0.904**	**0.898**	**0.882**	**0.857**	**0.833**	**0.275**	**0.295**	**0.351**	**0.416**	**0.473**

Table 3: Dehazing performance under five synthetic haze states. S1, light haze; S2, moderate haze; S3, dense haze; S4, low-light haze; S5, non-uniform haze. The middle block applies four representative TTA methods to HDCFN, the strongest VIS-IR static baseline. "Anchor only" freezes the cloud loop, while "full loop" enables it.

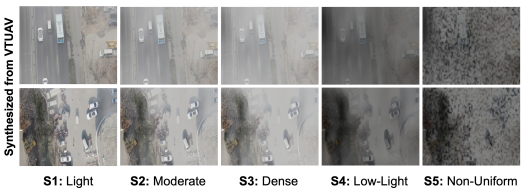


Figure 5: Examples of the synthesized haze shift.

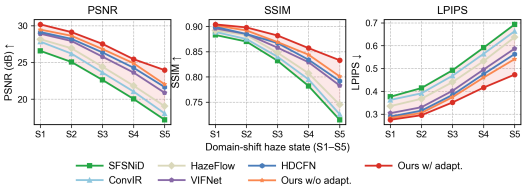


Figure 6: Performance trends under drifting haze.

If the set is empty, we set $k_W = W + 1$, so the acceptance condition is unattainable and the returned update is rejected. The binomial tail is conservative under independent signs with null success probability at most $1/2$. Consecutive video frames are correlated, so the acceptance step is treated as a calibrated deployment heuristic rather than a theorem. On rejection, the device rolls back to $\Theta_J^{\mathrm{adp},0}$.

4 Experiment

4.1 Experimental Settings

Datasets. We conduct experiments on four synchronized RGB-IR datasets with synthesized haze and one with naturally occurring haze. VTUAV [42] provides paired UAV imagery across urban and rural scenes. DroneVehicle [43] (abbreviated as DroneV) covers drone-mounted vehicle imagery. CART [44] contains high-resolution outdoor RGB-IR images. M3FD [45] contains campus and roadway scenes. CityUAV [11] contains UAV imagery captured in real-world foggy urban environments and provides haze scenes without paired clean ground truth, used in the no-reference evaluation.

Implementation. The framework is implemented in PyTorch. Cloud-side adaptation runs on NVIDIA A100 GPUs and edge inference runs on Jetson Orin Nano with batch size 1. We use Adam with $(\beta_1, \beta_2) = (0.9, 0.999)$, learning rate 10^{-4} decayed to 10^{-6}, training batch size 4, and 512×512 inputs. Unless otherwise stated, $\alpha = 0.05$, $\rho = 0.05$, $T = 32$, $W = 32$, and the trigger is evaluated at $1/4$ resolution. The betting functions are calibrated on a nominal fold of 10^4 frames disjoint from training, validation, and evaluation streams.

4.2 Comparison with Recent Dehazing Methods and TTA Baselines

We compare with five recent dehazers and four TTA baselines. Static baselines are retrained with official settings under the same data splits and haze synthesis protocol. The TTA methods include two general schemes (entropy minimization and consistency regularization) and two restoration-oriented schemes (TAO and Hybrid-TTA). All TTA variants are grafted onto HDCFN, the strongest VIS-IR static baseline, with schedules adapted to the shared streaming protocol. To avoid giving TTA baselines access to our anchor-adaptive architecture and cloud-validation loop, we keep the comparison on HDCFN as a fixed external backbone for recent TTA methods. We report PSNR [49], SSIM [49], and LPIPS [50]. The experimental analysis is organized as follows.

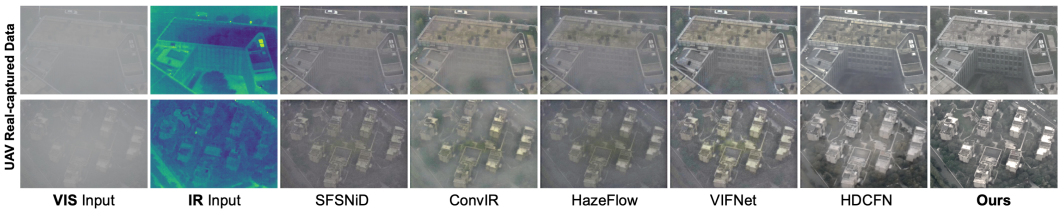


Figure 7: Representative dehazing results on CityUAV collected by UAVs in foggy urban scenes.

Method	NIQE ↓	PIQE ↓	FADE ↓
SFSNiD [46]	19.43	40.76	1.842
ConvIR [47]	18.85	38.63	1.713
HazeFlow [48]	18.37	37.58	1.605
VIFNet [10]	17.21	37.42	1.482
HDCFN [11]	16.54	36.57	1.387
HDCFN+EntMin [33]	17.02	37.21	1.456
HDCFN+ConsReg [34]	16.85	36.94	1.421
HDCFN+TAO [35]	16.41	36.18	1.342
HDCFN+Hybrid-TTA [36]	16.28	35.92	1.318
Ours (anchor only)	16.12	36.04	1.295
Ours (full loop)	**15.87**	**35.32**	**1.214**

Table 4: Real haze evaluation on CityUAV.

Stage	Latency
Edge per-frame inference Dual-head forward on Jetson Orin Nano	39.2 ms
E-process update per frame Two scalar bets and one product update	0.26 ms
Episode upload Compressed $T = 32$ VIS+IR over 20 Mbps	1.42 s
Cloud-side training 50 Adam steps on Θ_J^{adp} on one A100 GPU	5.21 s
Validation pass for $\gamma^\star$ Forward-only sweep on $\mathcal{X}_{\mathrm{val}}$	0.34 s
Parameter return Adaptive head, 0.66 MB FP32	0.27 s
Cumulative	**7.28** s

Table 5: Latencies of edge-cloud processing stages.

(i) Anchor-only dehazing. With cloud updates disabled, the frozen anchor already attains the best static performance among the compared methods across the four datasets (Tab. 2). It exceeds HDCFN by 0.38–1.22 dB in PSNR and serves as the fixed reference throughout subsequent adaptation.

(ii) Computational cost. The on-device model contains 5.19 M params and requires 284.1 GFLOPs, lower than HDCFN (6.83 M / 305.2 GFLOPs) and VIFNet (9.78 M / 708.5 GFLOPs). Per-frame runtimes on Jetson Orin Nano are 39.2 ms for ours, 43.5 ms for HDCFN, and 137.9 ms for VIFNet.

(iii) Distribution drift setup. We synthesize five-state haze streams on VTUAV using $I = J \odot T + A \odot (1 - T)$ with $T = \exp(-\beta d)$. S1 to S3 use progressively lower mean transmission. S4 combines dense haze with photometric attenuation. S5 uses a low-frequency spatial field $\beta(\cdot)$. Representative samples of the five states are shown in Fig. 5. Synthesis details are provided in Appendix K.

(iv) Performance under drifting haze. Tab. 3 reports state-wise results, and Fig. 6 visualizes the PSNR, SSIM, and LPIPS trends across S1 to S5. Naive TTA (EntMin, ConsReg) helps slightly on S1 but accumulates error and falls behind static HDCFN at S4 and S5, reflecting the absence of explicit stopping. TAO and Hybrid-TTA recover most of this loss but still trail our anchor-only model from S1 to S4 because their backbone is weaker, and they trail it on S5 as drift becomes severe enough that local TTA without rollback cannot keep pace. Activating the full loop gives the most stable trend across S1-S5. At S5 the loop improves over the no-adaptation variant by $+1.61$ dB PSNR, $+0.022$ SSIM, and -0.051 LPIPS, and over the strongest TTA baseline by $+1.77$ dB PSNR.

4.3 Evaluation on Real-World UAV Hazy Data

We evaluate on CityUAV [11], which contains UAV imagery with naturally occurring haze and no paired ground truth. We report three no-reference metrics: NIQE [51], PIQE [52], and FADE [53]. Tab. 4 reports static dehazers, four TTA variants applied to the HDCFN, and ours. Fig. 7 shows representative dehazing results. The full loop attains superior score among comparisons.

Tab. 5 reports the latency breakdown of edge inference and cloud adaptation. Edge-side inference averages 39.2 ms per frame on Jetson Orin Nano, and the trigger-to-return round trip averages 7.28 s, dominated by cloud-side training and the upload of the $T = 32$ episode. On the evaluated CityUAV stream, the observed latency remains below the typical interval between successive triggers. These measurements support the feasibility of edge-cloud adaptation under the evaluated conditions.

Variant	PSNR ↑	SSIM ↑	LPIPS ↓
Full model	**27.24**	**0.875**	**0.362**
w/o IR modality	25.85	0.837	0.428
w/o a_t^{ir}	26.13	0.853	0.398
w/o $\mathcal{L}_J$	20.91	0.703	0.522
w/o $\mathcal{L}_T$	26.04	0.844	0.410
w/o $\bar{\mathcal{L}}_{\mathrm{temp}}$	26.91	0.869	0.374
w/o $\bar{\mathcal{L}}_{\mathrm{rehaze}}$	26.42	0.858	0.390

Table 6: Ablation on learning objectives.

Variant	PSNR ↑	SSIM ↑	LPIPS ↓
Full loop	**27.24**	**0.875**	**0.362**
w/o dual-head	26.18	0.852	0.408
w/o cons. interpolation	26.81	0.866	0.379
w/o sign-test	26.45	0.858	0.394
w/o rollback	26.51	0.860	0.391

Table 7: Ablation on edge-cloud loop.

Variant	PSNR ↑	SSIM ↑	LPIPS ↓
Full model	**27.24**	**0.875**	**0.362**
w/o $E_t^{\min}$	25.76	0.846	0.413
fixed threshold trigger	26.56	0.862	0.389

Table 8: Ablation on device-side triggering.

α	Bound	Empirical prob.	PSNR ↑	SSIM ↑	LPIPS ↓
0.10	$\leq 10\%$	$(5.5 \pm 1.1)\%$	27.21	0.874	0.364
0.05	$\leq 5\%$	$(2.5 \pm 0.8)\%$	**27.24**	**0.875**	**0.362**
0.01	$\leq 1\%$	$(0.5 \pm 0.4)\%$	26.87	0.869	0.376

Table 9: Effect of α on false-trigger probability.

T	VTUAV			DroneV		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
16	26.95	0.869	0.377	22.71	0.762	0.421
32	**27.24**	**0.875**	**0.362**	**22.94**	**0.770**	**0.414**
64	27.19	0.874	0.364	22.87	0.768	0.417

Table 10: Sensitivity to episode length T.

4.4 Ablation Studies

(i) Learning objectives and modality. Tab. 6 reports the full model and six ablated variants. Removing IR drops PSNR by 1.39 dB, supporting the structural value of cross-modal fusion and the IR-edge critic. Replacing the in-norm gradient alignment with a no-scaling form (w/o a_t^{ir}) drops PSNR by 1.11 dB, suggesting that the in-norm alignment factor contributes to suppressing the rehaze-consistent degenerate family rather than a constant rescaling. Dropping $\mathcal{L}_J$ gives the largest degradation because supervised restoration anchors the offline model. Removing $\mathcal{L}_T$ weakens the haze proxy used by the rehaze critic. During cloud adaptation, $\bar{\mathcal{L}}_{\mathrm{rehaze}}$ contributes more than $\bar{\mathcal{L}}_{\mathrm{temp}}$.

(ii) Edge-cloud loop. Tab. 7 ablates the loop architecture. Removing the dual-head design drops PSNR by 1.06 dB because the adaptive head loses its anchor fallback when an updated parameter set degrades on rare frames. Setting $\gamma = 1$ in place of conservative interpolation drops PSNR by 0.43 dB, consistent with overfitting to a short uploaded episode. Removing the sign-test acceptance and removing rollback give comparable drops of 0.79 and 0.73 dB, since both let unverified parameters reach edge-side inference.

(iii) Trigger design. Tab. 8 shows that removing $E_t^{\min}$ drops PSNR by 1.48 dB, indicating that best-head inconsistency detects capability mismatch that haze uncertainty alone misses. Replacing the e-process with a fixed threshold drops PSNR by 0.68 dB and reflects both spurious uploads under correlated streams and missed shifts.

(iv) Trigger reliability. Tab. 9 reports empirical per-run false-trigger probability over 400 nominal VTUAV clips of length $L = 64$ frames drawn from $\mathcal{D}_0^{\mathrm{val}}$, disjoint from offline training and from the S1-S5 drift evaluation. Observed probabilities lie below the corresponding bounds 10%, 5%, and 1%. The setting $\alpha = 0.05$ yields the highest end-to-end PSNR among the three.

(v) Episode length. Tab. 10 varies $T \in \{16, 32, 64\}$. $T = 32$ gives the best PSNR on both VTUAV and DroneV. $T = 16$ provides less adaptation signal per cycle, while $T = 64$ increases the haze-state gap between early and late frames within a single drifting episode.

5 Conclusion

We presented a reliability-budgeted edge-cloud self-adaptation framework for continual VIS-IR UAV dehazing under drifting haze conditions. A joint rehaze and scale-aligned IR-edge critic reduces the degenerate solutions admitted by single-view rehaze, and upload triggering is posed as a sequential test on a non-negative e-process that yields per-run false-upload control under conditional calibration. Across the evaluated UAV streams, the closed loop consistently improves dehazing quality over no-adaptation and TTA baselines, while keeping inference on the edge and restricting cloud-side updates to the adaptive head. Future work will explore stable upload scheduling under fluctuating communication conditions, where transient link degradation may delay or interrupt cloud returns.

References

[1] Hao Yang, Liyuan Pan, Yan Yang, and Wei Liang. Language-driven all-in-one adverse weather removal. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 24902–24912, 2024.

[2] Siyuan Yao, Rui Zhu, Ziqi Wang, Wenqi Ren, Yanyang Yan, and Xiaochun Cao. UMDATrack: Unified multi-domain adaptive tracking under adverse weather conditions. In *Proceedings of the International Conference on Computer Vision*, pages 6466–6475, 2025.

[3] Zixin Wang, Yadan Luo, Liang Zheng, Zhuoxiao Chen, Sen Wang, and Zi Huang. In search of lost online test-time adaptation: A survey. *International Journal of Computer Vision*, 133(3):1106–1139, 2025.

[4] Guowei Wang and Changxing Ding. Effortless active labeling for long-term test-time adaptation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 25633–25642, 2025.

[5] Trung Hieu Hoang, MinhDuc Vo, and Minh Do. Persistent test-time adaptation in recurring testing scenarios. *Advances in Neural Information Processing Systems*, 37:123402–123442, 2024.

[6] Yuda Song, Zhuqing He, Hui Qian, and Xin Du. Vision transformers for single image dehazing. *IEEE Transactions on Image Processing*, 32:1927–1941, 2023.

[7] Chengyu Fang, Chunming He, Fengyang Xiao, Yulun Zhang, Longxiang Tang, Yuelin Zhang, Kai Li, and Xiu Li. Real-world image dehazing with coherence-based pseudo labeling and cooperative unfolding network. *Advances in Neural Information Processing Systems*, 37:97859–97883, 2024.

[8] Yuxin Feng, Long Ma, Xiaozhe Meng, Fan Zhou, Risheng Liu, and Zhuo Su. Advancing real-world image dehazing: Perspective, modules, and training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):9303–9320, 2024.

[9] Ruiyi Wang, Yushuo Zheng, Zicheng Zhang, Chunyi Li, Shuaicheng Liu, Guangtao Zhai, and Xiaohong Liu. Learning hazing to dehazing: Towards realistic haze generation for real-world image dehazing. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 23091–23100, 2025.

[10] Meng Yu, Te Cui, Haoyang Lu, and Yufeng Yue. VIFNet: An end-to-end visible-infrared fusion network for image dehazing. *Neurocomputing*, page 128105, 2024.

[11] Junwei Zhao, Qianchun Luo, Shiliang Zhang, Shen Gao, and Jie Wu. HDCFN: Haze distribution-aware cross-modal fusion network for infrared-guided dense haze removal in uavs. In *Proceedings of the ACM International Conference on Multimedia*, pages 10867–10875, 2025.

[12] Chung-Hsuan Hu, Zheng Chen, and Erik G. Larsson. Energy-efficient federated edge learning with streaming data: A lyapunov optimization approach. *IEEE Transactions on Communications*, 73(2):1142–1156, 2025.

[13] Yingchao Wang, Chen Yang, Shulin Lan, Liehuang Zhu, and Yan Zhang. End-edge-cloud collaborative computing for deep learning: A comprehensive survey. *IEEE Communications Surveys and Tutorials*, 26(4):2647–2683, 2024.

[14] Shaohui Lin, Xiaoxi Zhang, Yupeng Li, Carlee Joe-Wong, Jingpu Duan, Dongxiao Yu, Yu Wu, and Xu Chen. Online management for edge-cloud collaborative continuous learning: A two-timescale approach. *IEEE Transactions on Mobile Computing*, 2024.

[15] Jeong-A Lim, Joohyun Lee, Jeongho Kwak, and Yeongjin Kim. Cutting-edge inference: Dynamic DNN model partitioning and resource scaling for mobile ai. *IEEE Transactions on Services Computing*, 2024.

[16] Yuning Cui, Mingyu Liu, Wenqi Ren, and Alois Knoll. Hybrid frequency modulation network for image restoration. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 722–730, 2024.

[17] Zixuan Chen, Zewei He, and Zhe-Ming Lu. Dea-net: Single image dehazing based on detail-enhanced convolution and content-guided attention. *IEEE Transactions on Image Processing*, 33:1002–1015, 2024.

[18] Wei Dong, Han Zhou, Ruiyi Wang, Xiaohong Liu, Guangtao Zhai, and Jun Chen. Dehazedct: Towards effective non-homogeneous dehazing via deformable convolutional transformer. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 6405–6414, 2024.

[19] Fu-Jen Tsai, Yan-Tsung Peng, Yen-Yu Lin, and Chia-Wen Lin. PHATNet: A physics-guided haze transfer network for domain-adaptive real-world image dehazing. In *Proceedings of the International Conference on Computer Vision*, pages 5591–5600, 2025.

[20] Xinran Wang, Guang Yang, Tian Ye, and Yun Liu. Dehaze-retinexgan: Real-world image dehazing via retinex-based generative adversarial network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39(8), pages 7997–8005, 2025.

[21] Xingchen Zhang and Yiannis Demiris. Visible and infrared image fusion using deep learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(8):10535–10554, 2023.

[22] Kathakoli Sengupta, Zhongkai Shangguan, Sandesh Bharadwaj, Sanjay Arora, Eshed Ohn-Bar, and Renato Mancuso. Unified local-cloud decision-making via reinforcement learning. In *Proceedings of the European Conference on Computer Vision*, pages 185–203, 2024.

[23] Quan Chen, Ming Yi, Jing Li, Ning Li, Hong Gao, and Zhipeng Cai. Optimal and approximate parallelism-based computation offloading algorithms for real-time multimodal learning at the edge. In *IEEE Conference on Computer Communications*, pages 1–10, 2025.

[24] Di Yu, Changze Lv, Xin Du, Linshan Jiang, Wentao Tong, Zhenyu Liao, Xiaoqing Zheng, and Shuiguang Deng. ECC-SNN: Cost-effective edge-cloud collaboration for spiking neural networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 6904–6912, 2025.

[25] Guanqun Wang, Jiaming Liu, Chenxuan Li, Yuan Zhang, Junpeng Ma, Xinyu Wei, Kevin Zhang, Maurice Chong, Renrui Zhang, Yijiang Liu, et al. Cloud-device collaborative learning for multimodal large language models. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 12646–12655, 2024.

[26] Jiawei Shao and Xuelong Li. AI flow at the network edge. *IEEE Network*, 2025.

[27] Yongheng Deng, Feng Lyu, Tengxi Xia, Yuezhi Zhou, Yaoxue Zhang, Ju Ren, and Yuanyuan Yang. A communication-efficient hierarchical federated learning framework via shaping data distribution at edge. *IEEE/ACM Transactions on Networking*, 32(3):2600–2615, 2024.

[28] Sibendu Paul, Utsav Drolia, Y. Charlie Hu, and Srimat Chakradhar. AQuA: A new image quality metric for optimizing video analytics systems. *ACM Transactions on Embedded Computing Systems*, 22(4):1539–9087, 2023.

[29] Hasan Burhan Beytur, Ahmet Gunhan Aydin, Gustavo de Veciana, and Haris Vikalo. Optimization of offloading policies for accuracy-delay tradeoffs in hierarchical inference. In *IEEE Conference on Computer Communications*, pages 1989–1998, 2024.

[30] Zhitong Gao, Shipeng Yan, and Xuming He. ATTA: Anomaly-aware test-time adaptation for out-of-distribution detection in segmentation. *Advances in Neural Information Processing Systems*, 36:45150–45171, 2023.

[31] Jin Shin and Hyun Kim. L-TTA: Lightweight test-time adaptation using a versatile stem layer. *Advances in Neural Information Processing Systems*, 37:39325–39349, 2024.

[32] Mouxing Yang, Yunfan Li, Changqing Zhang, Peng Hu, and Xi Peng. Test-time adaptation against multi-modal reliability bias. In *International Conference on Learning Representations*, 2024.

[33] Zhengqing Gao, Xu-Yao Zhang, and Cheng-Lin Liu. Unified entropy optimization for open-set test-time adaptation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 23975–23984, 2024.

[34] Rui Yan, Hongyu Qu, Xiangbo Shu, Wenbin Li, Jinhui Tang, and Tieniu Tan. DTS-TPT: dual temporal-sync test-time prompt tuning for zero-shot activity recognition. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1534–1542, 2024.

[35] Yuanbiao Gou, Haiyu Zhao, Boyun Li, Xinyan Xiao, and Xi Peng. Test-time degradation adaptation for open-set image restoration. In *International Conference on Machine Learning*, pages 16167–16177, 2024.

[36] Hyewon Park, Hyejin Park, Jueun Ko, and Dongbo Min. Hybrid-TTA: Continual test-time adaptation via dynamic domain shift detection. In *Proceedings of the International Conference on Computer Vision*, pages 2877–2886, 2025.

[37] Jinliang Liu and Zongxin Yang. Test-time adaptation for real-world video adverse weather restoration with meta batch normalization. *IEEE Transactions on Circuits and Systems for Video Technology*, 35(6):5533–5544, 2025.

[38] Jiaming Liu, Ran Xu, Senqiao Yang, Renrui Zhang, Qizhe Zhang, Zehui Chen, Yandong Guo, and Shanghang Zhang. Continual-MAE: Adaptive distribution masked autoencoders for continual test-time adaptation. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 28653–28663, 2024.

[39] Liang Wang, Kai Lu, Nan Zhang, Xiaoyang Qu, Jianzong Wang, Jiguang Wan, Guokuan Li, and Jing Xiao. Shoggoth: Towards efficient edge-cloud collaborative real-time video inference via adaptive online learning. In *ACM/IEEE Design Automation Conference*, pages 1–6, 2023.

[40] Shijing Hu, Zhihui Lu, Xin Xu, Ruijun Deng, Xin Du, and Qiang Duan. LAECIPS: Large vision model assisted adaptive edge-cloud collaboration for iot-based embodied intelligence system. *Journal of Industrial Information Integration*, page 100955, 2025.

[41] Yudong Liang, Bin Wang, Wangmeng Zuo, Jiaying Liu, and Wenqi Ren. Self-supervised learning and adaptation for single image dehazing. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 1137–1143, 2022.

[42] Pengyu Zhang, Jie Zhao, Dong Wang, Huchuan Lu, and Xiang Ruan. Visible-thermal UAV tracking: A large-scale benchmark and new baseline. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 8886–8895, 2022.

[43] Yiming Sun, Bing Cao, Pengfei Zhu, and Qinghua Hu. Drone-based RGB-infrared cross-modality vehicle detection via uncertainty-aware learning. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(10):6700–6713, 2022.

[44] Connor Lee, Matthew Anderson, Nikhil Ranganathan, Xingxing Zuo, Kevin Do, Georgia Gkioxari, and Soon-Jo Chung. Caltech aerial RGB-Thermal dataset in the wild. In *European Conference on Computer Vision*, pages 236–256, 2024.

[45] Jinyuan Liu, Xin Fan, Zhanbo Huang, Guanyao Wu, Risheng Liu, Wei Zhong, and Zhongxuan Luo. Target-aware dual adversarial learning and a multi-scenario multi-modality benchmark to fuse infrared and visible for object detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 5802–5811, 2022.

[46] Xiaofeng Cong, Jie Gui, Jing Zhang, Junming Hou, and Hao Shen. A semi-supervised nighttime dehazing baseline with spatial-frequency aware and realistic brightness constraint. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 631–640, 2024.

[47] Yuning Cui, Wenqi Ren, Xiaochun Cao, and Alois Knoll. Revitalizing convolutional network for image restoration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):9423–9438, 2024.

[48] Junseong Shin, Seungwoo Chung, Yunjeong Yang, and Tae Hyun Kim. HazeFlow: Revisit haze physical model as ode and non-homogeneous haze generation for real-world dehazing. In *Proceedings of the International Conference on Computer Vision*, pages 6263–6272, 2025.

[49] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[50] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018.

[51] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a completely blind image quality analyzer. *IEEE Signal Processing Letters*, 20(3):209–212, 2012.

[52] Narasimhan Venkatanath, D Praneeth, Maruthi Chandrasekhar Bh, Sumohana S Channappayya, and Swarup S Medasani. Blind image quality evaluation using perception based features. In *National Conference on Communications*, pages 1–6, 2015.

[53] Lark Kwon Choi, Jaehee You, and Alan Conrad Bovik. Referenceless prediction of perceptual fog density and perceptual image defogging. *IEEE Transactions on Image Processing*, 24(11):3888–3901, 2015.

[54] Glenn Shafer. Testing by betting: A strategy for statistical and scientific. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 184(2):407–431, 2021.

[55] Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4):576–601, 2023.

[56] Jean Ville. *Etude critique de la notion de collectif*, volume 3. Gauthier-Villars Paris, 1939.

[57] René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1623–1637, 2020.

[58] Reiner Birkl, Diana Wofk, and Matthias Müller. MiDaS v3. 1–a model zoo for robust monocular relative depth estimation. *arXiv preprint arXiv:2307.14460*, 2023.

Technical Appendix

A Architecture Details

The model comprises paired VIS–IR encoders, a haze-state branch, confidence-guided fusion, a shared decoder, and two restoration heads. The encoders produce features at four scales with resolutions 1, $1/2$, $1/4$, and $1/8$ relative to the input. The decoder combines the fused features through skip connections and upsampling.

Haze-state quantities. At scale l, the haze-state branch predicts a transmittance logit $\mu_{t,l}$ and a log-variance map $s_{t,l}$. We use

$$\hat{T}_{t,l} = \sigma(\mu_{t,l}), \qquad C_{t,l} = \sigma(-s_{t,l}).$$

The trigger statistic U_t uses $C_{t,l^\star}$ at $l^\star = 3$. For the rehaze critic, $\hat{T}_{t,l^\star}$ and $C_{t,l^\star}$ are bilinearly upsampled to the output resolution. The IR critic uses the output luminance and IR image at that resolution. The confidence map is a learned weighting signal, not a calibrated probability of restoration correctness.

Fusion and restoration. At each scale, sigmoid gates $g_{t,l}^{\mathrm{safe}}$ and $g_{t,l}^{\mathrm{haze}}$ are computed from the paired features, with the latter additionally receiving $\hat{T}_{t,l}$. Fusion uses

$$g_{t,l} = (1 - C_{t,l}) \odot g_{t,l}^{\mathrm{safe}} + C_{t,l} \odot g_{t,l}^{\mathrm{haze}}, \qquad F_{t,l} = g_{t,l} \odot F_{t,l}^{\mathrm{vis}} + (1 - g_{t,l}) \odot F_{t,l}^{\mathrm{ir}}.$$

Lower confidence shifts the weighting toward the haze-agnostic gate. The two residual restoration heads receive the same decoded features. After offline training, Θ_J^{adp} is initialized from Θ_J^{fix}. The shared trunk and anchor head are held fixed during deployment. Only the adaptive restoration head is updated by the cloud.

B Offline Loss Functions

Offline training uses synthetic tuples $(I^{\mathrm{vis}}, I^{\mathrm{ir}}, J^{\mathrm{vis}}, T)$ generated by the synthesis procedure in Appendix K. The objective is

$$\mathcal{L}_{\mathrm{offline}} = \mathcal{L}_J + \mathcal{L}_T, \qquad \mathcal{L}_J = \mathrm{mean}\left(|\hat{J}^{\mathrm{fix}} - J^{\mathrm{vis}}|\right).$$

The haze-state term supervises scales $\mathcal{S} = \{1, 2, 3\}$:

$$\mathcal{L}_T = \frac{1}{|\mathcal{S}|} \sum_{l \in \mathcal{S}} \mathrm{mean}\left((\hat{T}_l - T_l)^2 \odot \exp(-s_l) + s_l\right),$$

where T_l is the synthetic transmittance resized to scale l. This is the heteroscedastic Gaussian regression form, up to a constant multiplier and an additive constant. The adaptive head is copied from the anchor after offline training and is not trained separately at this stage. The synthesis range includes the five evaluated haze states. This coverage does not itself establish calibration of the deployment-time trigger.

C Closed-Form Estimation of Atmospheric Light

For each candidate restoration $\hat{J}_t$, atmospheric light is represented by a spatially constant vector in $[0, 1]^3$. Using the frozen trunk outputs $\hat{T}_t$ and C_t, its confidence-weighted ridge estimate is

$$A_t(\hat{J}_t) = \mathrm{clip}_{[0,1]^3}\left(\frac{\sum_p C_t(p)(1 - \hat{T}_t(p))\left(x_t^{\mathrm{vis}}(p) - \hat{J}_t(p)\hat{T}_t(p)\right)}{\sum_p C_t(p)(1 - \hat{T}_t(p))^2 + \epsilon_A}\right), \qquad \epsilon_A > 0.$$

Clipping is componentwise. The reconstructed observation is

$$\widetilde{x}_t^{\mathrm{vis}}(\hat{J}_t) = \hat{J}_t \odot \hat{T}_t + (1 - \hat{T}_t) \odot A_t(\hat{J}_t).$$

The rehaze score is Eq. (2), with the residual averaged over pixels and RGB channels and the scalar confidence map broadcast across channels. Its denominator uses a positive numerical stabilizer. For gradient-bearing evaluations, $A_t(\hat{J}_t)$ is computed for the current candidate and then detached together with $\hat{T}_t$ and C_t. Thus the update does not differentiate through the atmospheric-light solver.

D Underdetermination of Single-Modality Rehaze

We consider the single-view scattering model with spatially constant atmospheric light. The construction gives a family of image and transmittance pairs that reproduce the same observation.

On a finite image domain, let $x(p) = J^*(p)T^*(p) + (1 - T^*(p))A^*$, with $0 < T^*(p) < 1$. For $\lambda \in (\max_p T^*(p), 1]$, define

$$\hat{J}_\lambda(p) = \lambda J^*(p) + (1-\lambda)A^*, \qquad \hat{T}_\lambda(p) = T^*(p)/\lambda.$$

Then $0 < \hat{T}_\lambda(p) < 1$, and direct substitution gives

$$\hat{J}_\lambda(p)\hat{T}_\lambda(p) + (1 - \hat{T}_\lambda(p))A^* = J^*(p)T^*(p) + (1 - T^*(p))A^* = x(p).$$

For $\lambda < 1$, the restoration differs from J^* wherever $J^*(p) \neq A^*$. This equality concerns the joint scattering inverse problem. The deployed critic instead fixes $\hat{T}_t$ and uses the regularized estimate in Appendix C. The IR term supplies an additional structural criterion for comparing candidate restorations, without implying unique recovery of the clear image.

E Cross-Modal Critic Details

For RGB images, let $Y(I) = 0.299I^R + 0.587I^G + 0.114I^B$, and set $Y(I) = I$ for single-channel inputs. The gradient notation in Eq. (3) uses a fixed Sobel pair:

$$\nabla I = (K_x * Y(I), K_y * Y(I)).$$

The IR edge weights and alignment factor are

$$M_t^{\mathrm{ir}}(p) = \sigma\left(\kappa_m(\|\nabla x_t^{\mathrm{ir}}(p)\|_2 - \eta_m)\right),$$

$$a_t^{\mathrm{ir}} = \mathrm{clip}_{[a_{\min}, a_{\max}]}\left(\frac{\sum_p M_t^{\mathrm{ir}}(p)\langle \nabla \hat{J}_t^{\mathrm{fix}}(p), \nabla x_t^{\mathrm{ir}}(p)\rangle}{\sum_p M_t^{\mathrm{ir}}(p)\|\nabla x_t^{\mathrm{ir}}(p)\|_2^2 + \epsilon_g}\right),$$

where $\kappa_m > 0$, $\eta_m \geq 0$, $\epsilon_g > 0$, and $0 < a_{\min} < a_{\max}$. Both quantities are shared across candidates for the same frame and detached during backpropagation.

The IR score is exactly Eq. (3): the absolute value is elementwise, and the mean is over pixels and both gradient components. The mask is broadcast across those components, with no division by its mean. The scale factor changes the IR target inside the residual, whereas λ_{ir} weights this term in the joint critic.

F Construction of the E-Process

We construct the upload statistic using bounded betting functions [54, 55]. Calibration uses the frozen reference scores $Z_{U,t} = U_t$ and $Z_{E,t} = E_t^{\mathrm{fix}}$, while deployment evaluates b_E at $E_t^{\min}$. Three disjoint nominal folds are used for rank-transform fitting, slack fitting, and empirical evaluation, denoted $\mathcal{D}_0^{\mathrm{cal}}$, $\mathcal{D}_0^{\mathrm{slack}}$, and $\mathcal{D}_0^{\mathrm{val}}$, respectively. They do not overlap with the drifting evaluation streams or cloud-adaptation episodes.

Betting functions. For $S \in \{U, E\}$, the $n_{\mathrm{cal}} = 10^4$ calibration scores define the mid-rank transform and raw bet

$$\hat{F}_S(z) = \frac{1}{n_{\mathrm{cal}}}\sum_{i=1}^{n_{\mathrm{cal}}}\left(\mathbf{1}[Z_S^{(i)} < z] + \frac{1}{2}\mathbf{1}[Z_S^{(i)} = z]\right),$$

$$\tilde{b}_S(z) = 1 + \eta_S\left(\hat{F}_S(z) - \frac{1}{2}\right), \qquad 0 \leq \eta_S < 2.$$

Thus $\tilde{b}_S$ is positive, non-decreasing, and takes values in $[1-\eta_S/2, 1+\eta_S/2]$. Its empirical average on the calibration scores is exactly one, including ties.

On contiguous slack clips, let $\hat{\mu}_{S,b}$ be the mean raw bet for adjacent-frame pairs whose previous reference score lies in bin b. Only bins with observed pairs are used, and pairs crossing clip boundaries are excluded. The deployed bet is

$$b_S(z) = \tilde{b}_S(z)/c_S, \qquad c_S = \max\left\{1, \max_b \hat{\mu}_{S,b} + \Delta_S\right\}, \quad \Delta_S \geq 0.$$

The transforms, bins, and slack factors are fixed before monitoring.

Calibration assumption. Let $\mathcal{F}_0$ contain the offline model, reference data, and fitted calibration quantities. The filtration $\mathcal{F}_t$ additionally contains the observations, decisions, and algorithmic randomness available through frame t. For the reference-based calibration used here, we assume that the nominal deployment law P satisfies

$$\mathbb{E}_P[\tilde{b}_S(Z_{S,t}) \mid \mathcal{F}_{t-1}] \leq c_S, \qquad S \in \{U, E\},$$

almost surely at every monitored frame, including after restarts. Because $E_t^{\min} \leq E_t^{\text{fix}}$ and b_E is non-decreasing, $b_E(E_t^{\min}) \leq b_E(E_t^{\text{fix}})$ pointwise. The reference-score condition therefore implies the conditional calibration of both deployed bets required by Theorem 1.

Temporal dependence is allowed subject to this condition. Nominal clips are used for empirical fitting and evaluation. Validity of the fitted bounds on the deployment stream remains an assumption on its conditional law.

Monitoring rule. The device uses $\bar{e}_t$, $\bar{M}_t$, and τ from Eqs. (6)–(7), with $\bar{M}_0 = 1$ and $\alpha \in (0, 1)$. The arithmetic combination of the two bets does not require independence between the indicators.

G Proof of Theorem 1

Consider one monitoring run under a nominal law satisfying the theorem's conditional calibration assumption. Linearity gives $\mathbb{E}[\bar{e}_t \mid \mathcal{F}_{t-1}] \leq 1$. The bets are nonnegative, bounded, and adapted, so their product is integrable at every finite time and satisfies

$$\mathbb{E}[\bar{M}_t \mid \mathcal{F}_{t-1}] = \bar{M}_{t-1}\mathbb{E}[\bar{e}_t \mid \mathcal{F}_{t-1}] \leq \bar{M}_{t-1}.$$

Thus $\bar{M}_t$ is a nonnegative supermartingale starting at one. For any finite n, optional stopping at $\tau \wedge n$ yields

$$\frac{1}{\alpha}\Pr(\tau \leq n \mid \mathcal{F}_0) \leq \mathbb{E}[\bar{M}_{\tau \wedge n} \mid \mathcal{F}_0] \leq 1.$$

Letting $n \to \infty$ gives $\Pr(\tau < \infty \mid \mathcal{F}_0) \leq \alpha$, and averaging over $\mathcal{F}_0$ proves the stated result. This is the threshold-crossing form of Ville's inequality [56].

Monitoring after adaptation. After either acceptance or rejection, a new run starts from one. Its start is determined by the information available at that time. Conditional on this information, the same proof applies whenever the calibration assumption holds throughout the new nominal run. The frozen trunk and anchor preserve the reference scoring functions, and $b_E(E_t^{\min}) \leq b_E(E_t^{\text{fix}})$ holds for any adaptive head. Application after a restart also requires the conditional calibration bound under the subsequent input law. The level α applies to each nominal monitoring run separately.

H Cloud-Side Adaptation Loss

At a trigger time t_0, the device collects the episode $\mathcal{X}_{t_0:t_0+T-1}$ and suspends further triggers until the acceptance decision. Let $\Theta_J^{\text{adp},0}$ be the deployed adaptive head. The cloud freezes the shared trunk and anchor, and updates the adaptive head together with the task scales.

Data split and losses. With $n_{\mathrm{tr}} = \lfloor 0.8T \rfloor$, the first n_{tr} frames form $\mathcal{X}_{\mathrm{tr}}$ and the remaining frames form $\mathcal{X}_{\mathrm{val}}$. For $T = 32$, these contain 25 and 7 frames. Both splits have at least two frames in the evaluated settings. Gradient updates use only $\mathcal{X}_{\mathrm{tr}}$.

For either contiguous split, let $\mathcal{I}$ be its frame-index set. Its joint-critic and temporal losses are

$$\bar{\mathcal{L}}_{\mathrm{rehaze}} = \frac{1}{|\mathcal{I}|} \sum_{t \in \mathcal{I}} \mathrm{Crit}(\hat{J}_t^{\mathrm{adp}}),$$

$$\bar{\mathcal{L}}_{\mathrm{temp}} = \frac{1}{|\mathcal{I}| - 1} \sum_{t:\, t,t+1 \in \mathcal{I}} \frac{\mathrm{mean}\left(|\hat{J}_{t+1}^{\mathrm{adp}} - W_{t \to t+1}(\hat{J}_t^{\mathrm{adp}})| \odot m_t \right)}{\mathrm{mean}(m_t) + \epsilon_m}, \qquad \epsilon_m > 0.$$

Here $W_{t \to t+1}$ backward-samples frame t onto the frame-$t+1$ grid using a frozen IR correspondence estimator. The nonnegative mask m_t weights correspondences using forward–backward consistency. Temporal pairs never cross the train–validation boundary. Correspondence fields and masks are detached, while gradients through the warped restoration remain active. The joint critic uses the stop-gradient conventions in Appendices C and E.

Task scales and parameter selection. The two averaged losses enter Eq. (9). Positive scales are parameterized as $\sigma_i = \mathrm{softplus}(q_i) + \epsilon_\sigma$, with $\epsilon_\sigma > 0$, and learned on $\mathcal{X}_{\mathrm{tr}}$. This is a Laplace-style weighting of the residual objectives, without an independence claim for video pixels or frames.

After training, let $\Theta_J^{\mathrm{adp}}(\gamma) = (1-\gamma)\Theta_J^{\mathrm{adp},0} + \gamma\Theta_J^{\mathrm{adp,tr}}$. All interpolation candidates in Eq. (10) are evaluated on $\mathcal{X}_{\mathrm{val}}$ using the same trained task scales and no further gradient updates. The search uses a fixed finite grid $\Gamma \subset [0,1]$ containing 0 and 1, and selects the smallest minimizer. Including $\gamma = 0$ ensures that the selected validation loss does not exceed that of the deployed head. If no positive grid point strictly improves the objective, the deployed parameters are returned.

I Sign-Test Acceptance

The returned candidate is compared with the deployed adaptive head on W new frames observed after its arrival. Both parameter vectors remain fixed during this shadow window, and the device continues to output the mixture of the anchor and the old adaptive head. For each frame, let $d_t = E_t^{\mathrm{adp,new}} - E_t^{\mathrm{adp,old}}$. The candidate is accepted according to Eq. (11), with ties counted as non-improvements. If no binomial threshold satisfies the specified level, $k_W = W+1$ and acceptance is impossible. Rejection retains $\Theta_J^{\mathrm{adp},0}$. Monitoring resumes from $\bar{M}_0 = 1$ after either decision.

For the statistical interpretation, condition on all information available before the shadow window, including candidate selection. If the signs are conditionally independent and each has improvement probability at most $1/2$, their sum is stochastically dominated by $\mathrm{Binomial}(W, 1/2)$, giving an acceptance probability at most ρ under this null. Consecutive video frames need not satisfy these conditions, so ρ is a nominal level for the empirical acceptance rule. The thresholds in Table 11 follow directly from the binomial formula.

J Acceptance Sensitivity and Conservative Interpolation

Table 11 reports the evaluated combinations of window length and nominal level on S5. The stress-test candidates are obtained by training on mismatched episode pairs. Their acceptance frequencies describe this constructed candidate set.

W	ρ	k_W	PSNR (dB) ↑	Accept rate	Stress-test accept rate
16	0.05	12	23.71	0.81	0.044
32	0.05	22	**23.95**	0.86	0.041
64	0.05	40	23.93	0.84	0.038
32	0.01	24	23.62	0.69	0.010
32	0.10	21	23.84	0.91	0.092

Table 11: Acceptance sensitivity on S5.

Interpolation. The return strategies use $\gamma = 0$, $\gamma = 1$, or validation selection. On the evaluated S4 and S5 streams, validation selection improves PSNR over the full-update variant by 0.18 dB on average. The selected coefficient is below one in 34% of accepted cycles.

IR-edge weight. On S5, $\lambda_{\mathrm{ir}} \in \{0, 0.5, 1, 2\}$ gives PSNR values 22.78, 23.61, 23.95, and 23.74, respectively. The default $\lambda_{\mathrm{ir}} = 1$ gives the highest PSNR among these evaluated weights.

Figure 8 illustrates the effect of mean transmittance $\bar{T}$ on the synthesized observations.



Figure 8: Effect of mean transmittance $\bar{T}$ on synthesized haze.

K Haze Generation

Synthetic visible observations are generated using a depth-guided atmospheric scattering model by

$$I(p) = J_{\mathrm{syn}}(p)T(p) + A(1 - T(p)), \qquad T(p) = \exp(-\beta(p)d(p)).$$

Here A is spatially constant atmospheric light, $\beta(p) \geq 0$ controls haze density, and J_{syn} is the visible image supplied to the scattering operation after any state-specific photometric preprocessing. The nonnegative depth proxy d is derived from a pretrained MiDaS model [57, 58] and is used as a relative synthesis quantity rather than a metric distance. The same proxy is reused when generating different states from the same source image.

Table 12 summarizes the haze and illumination settings. S1–S3 use spatially constant β. S4 applies photometric attenuation before scattering, and S5 uses a low-frequency spatial field $\beta(p)$. The symbol β denotes the scattering coefficient, whereas ρ denotes the nominal acceptance-test level.

State	Condition	Synthesis control
S1	Light haze	Target $\bar{T} \approx 0.75$
S2	Moderate haze	Target $\bar{T} \approx 0.60$
S3	Dense haze	Target $\bar{T} \approx 0.45$
S4	Low-light haze	S3 scattering with photometric attenuation before scattering
S5	Non-uniform haze	Spatially varying $\beta(p)$ with S3 average density

Table 12: Five-state haze synthesis protocol on VTUAV.

L Limitations

The false-upload guarantee applies to nominal monitoring runs where each scaled reference bet has conditional mean at most one given the deployment history, with calibration maps fixed before monitoring. This condition must hold after each restart, and the level α applies separately to each run.

For temporally dependent video, ρ is a nominal level for shadow-window acceptance. Image-quality gains are assessed empirically and depend on cross-modal alignment and the haze proxy. Detection delay depends on the stream and accumulated evidence. Delayed cloud returns postpone adaptation while edge inference continues using the critic-gated mixture of the frozen anchor and the last accepted adaptive head.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The paper's contributions are reflected in the abstract and outlined in the introduction. These claims, including the motivations, are aligned with and supported by our experimental results.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The limitations are discussed in Appendix.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [Yes]

Justification: For theoretical results, the paper provides the full set of assumptions along with a complete (and correct) proof.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper and supplementary material provide the details needed to reproduce the main experimental results, including the model architecture, training procedures, loss functions, datasets and evaluation protocols, and implementation settings.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: All datasets used in our experiments are publicly available UAV-related datasets. Details of the datasets and experimental settings are provided in the supplementary material.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Experimental settings and analysis of experimental results are provided.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: We report standard errors for the empirical false-trigger probabilities, with the calculation detailed in the supplementary material. We additionally report means and standard deviations of model performance in the author response.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The paper provides a detailed description of the computer resources used in the experiments.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research presented in this paper complies with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: This work improves UAV vision in challenging environments, with potential benefits for surveillance, remote sensing, and disaster assessment, where clearer imagery can support more informed decision-making.

Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cite the authors and owners for all assets used in the paper and fully adhere to the licenses and terms of use explicitly mentioned.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: No new assets are released with this submission.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Our work does not include crowdsourcing or research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve human participants.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: LLMs were used to assist with manuscript drafting and language editing, as well as implementation of the model code.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.