Article

Neighborhood-Constrained Video Stabilization and Frame-Centering Compensation for UAV Target Tracking

Huihui Yang[1,*] and Yongkang Xia[1]

[1] Xinjiang College of Science and Technology, Korla, China
* Correspondence: Huihui Yang, Xinjiang College of Science and Technology, Korla, China

Abstract: Camera jitter and target drift degrade visual tracking by unmanned aerial vehicles (UAVs), especially near frame boundaries. This study proposes a modular post-processing method that combines neighborhood-constrained video stabilization with frame-centering compensation. Adjacent-frame phase correlation estimates global translation. A local median and Gaussian temporal weights, combined with residual down-weighting, construct a robust camera path that suppresses isolated motion-estimation errors. A bounded centering term then moves the target toward a safety region while preserving frame content. The method requires no inertial sensor and can operate after correlation-filter, Siamese, or Transformer trackers. A pilot was conducted on an 898-frame UAV123 subset. Target-center RMS error decreased from 164.58 to 82.58 pixels (49.82%), safety-region coverage increased from 65.26% to 100%, and frame retention remained 90.36%. Under injected noise levels of 2, 4, and 8 pixels, trajectory-recovery error decreased by 57.95%, 68.70%, and 74.26%, respectively. These results indicate geometric feasibility, while full UAV123, UAV20L, and DTB70 benchmarks remain necessary before general performance claims.

Keywords: UAV tracking; video stabilization; neighborhood constraint; frame centering; camera-motion compensation

1. Introduction

1.1. Research Background and Significance

UAVs provide flexible deployment, rapid response, and diverse viewing angles. They are used in traffic inspection, disaster search and rescue, power-line inspection, border patrol, and target reconnaissance. Visual target tracking is a foundational capability for these applications and requires consistent estimates of target identity, position, and scale across a video sequence.

Compared with generic visual tracking, UAV tracking is affected by platform attitude changes, wind disturbance, camera vibration, rapid target motion, and substantial scale variation. Camera motion can move the target repeatedly within the tracker search window or even outside the frame. When the target approaches a boundary, the available context decreases and the tracker becomes more susceptible to drift. Improving detection or appearance modeling alone does not resolve repeated loss caused by unstable input geometry.

Video stabilization estimates, smooths, and inversely compensates camera motion to reduce high-frequency jitter. Conventional stabilizers usually optimize scene smoothness without considering target position. A stable scene may therefore still leave the target near an image edge. Conversely, forcing the target to the center can disturb the global stabilization path, create empty image regions, or cause abrupt camera movement. The design problem is to preserve temporal consistency and intentional UAV motion while selectively correcting the target's position within a bounded crop region.

1.2. Research Objectives

Received: 31 August 2026
Revised: 14 October 2026
Accepted: 24 October 2026
Published: 30 October 2026

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

This study develops an image-domain compensation method for UAV single-object tracking that does not require inertial measurement unit (IMU) data. The objectives are to: (1) estimate global camera motion under motion blur, repeated background structure, and local target motion; (2) regularize the camera path with temporal neighbors so that isolated errors are suppressed without erasing sustained flight motion; (3) generate a bounded centering correction from the tracker output; and (4) quantify geometric effectiveness, frame-retention cost, and computational load in a reproducible public-data pilot.

1.3. Literature Review

1.3.1. International Research

Generic single-object tracking has progressed from correlation filters and Siamese matching to Transformer-based and multi-frame architectures. OSTrack unifies feature learning and relation modeling through a one-stream template-search representation and reduces the repeated computation of two-stage pipelines [1]. ODTrack propagates dense temporal tokens across consecutive frames and emphasizes video-level context for long-term association [2]. AVTrack introduces adaptive Transformer-layer activation and view-invariant representations for resource-constrained UAV platforms and reports results on UAV123, UAVDT, and DTB70 [3]. SiamDMU updates templates with motion and semantic masks to reduce template contamination under severe appearance change [4]. These methods improve target matching and template adaptation, but their primary objective is bounding-box prediction rather than stable output geometry.

Video stabilization research provides complementary techniques. Grundmann et al. optimized camera paths with an L1 objective that removes jitter while retaining portions of intentional camera motion [5]. Jeon et al. combined particle-keypoint tracking with L1 camera-path optimization for sparse-feature scenes [6]. Lee proposed constrained camera-path planning for real-time stabilization [7]. Souza and Pedrini compared stabilization algorithms and evaluation practices [8] and later introduced a camera-motion-field metric for two-dimensional motion estimation [9]. Huang et al. decomposed motion compensation to balance content preservation and stability [10]. These studies provide the basis for camera-path modeling, but rarely include an explicit target-center objective.

At the platform-control level, Yang et al. studied image-based visual servoing with a pan-tilt camera on a fixed-wing UAV [11]. Krakow et al. jointly optimized UAV guidance and gimbal control [12]. Nousi et al. implemented embedded real-time UAV detection and tracking [13]. Such systems can center a target through mechanical or flight control, but introduce additional degrees of freedom, control coupling, and platform cost. The present method provides a digital fallback that can operate on the existing video stream.

1.3.2. Research from Chinese Institutions

Research from Chinese institutions also emphasizes lightweight models, robustness in complex scenes, and onboard deployment. Keawboontan and Thammawichai studied joint detection and association for real-time UAV multi-target tracking [14]. Liang et al. proposed a feature-aligned attention network for UAV video [15], while Zhu et al. used multi-level prediction in a Siamese tracker [16]. Correlation filtering remains attractive for engineering deployment because of its low arithmetic cost [17]. Visible-thermal UAV benchmarks extend evaluation to low-light and adverse-illumination conditions [18].

Studies in Chinese applied science and engineering journals also emphasize deployability. Yu et al. proposed a lightweight vision Transformer for UAV tracking under occlusion and viewpoint change, combining cross-layer feature fusion gating with a view-aware mutual-information constraint [19]. Yu et al. cascaded YOLOv5 with KCF through detection-triggered tracking and loop verification and evaluated center error, success AUC, and FPS on UAV123 [20]. Zhao et al. combined attention, a Kalman motion prior, and a re-detection branch for occlusion and interference handling [21]. These studies recognize onboard computation, occlusion re-detection, and viewpoint variation as practical priorities. Stabilization is still often treated as a separate preprocessing stage, and the target state is not used to shape the stabilization path.

1.3.3. Critical Review

Existing research follows three main routes: improving target matching and template updating, optimizing camera paths for overall video stability, and using a gimbal or flight controller to orient the sensor. These routes address recognition, scene stability, and target centering separately. In image-only processing, three gaps remain. First, global stabilizers often treat the background as the dominant motion source, so target motion may be mistaken for camera shake. Second, fixed-window filters do not distinguish smooth intentional flight from isolated matching errors. Third, aggressive centering produces large translations and black borders. This study couples target position to a neighborhood-regularized camera path and limits every correction through explicit amplitude and boundary constraints.

1.4. Contributions

The contributions are as follows. First, a neighborhood-regularized camera-path model combines temporal distance and measurement residuals to suppress sparse phase-correlation errors while retaining slow UAV motion. Second, a bounded target-centering term is fused with the stabilization correction and parameterized by the image boundaries and a maximum displacement. Third, a reproducible lightweight UAV123 subset and pilot protocol are provided. Fourth, the study reports both center-error improvement and its content-retention cost, thereby separating geometric effectiveness from tracking performance.

1.5. Organization

Section 2 formulates the coupled stabilization and centering problem. Section 3 presents the proposed method. Section 4 describes the public-data pilot, baselines, metrics, and results. Section 5 discusses parameter settings, limitations, and a full benchmark plan. Section 6 concludes the study and provides engineering recommendations.

2. Problem Formulation and Overall Design

2.1. Notation and Task Scene

Let I_t denote the frame at time t, $B_t=(x_t,y_t,w_t,h_t)$ the target bounding box, p_t the target center, and p^* the image center. The UAV mission requires the target to remain visible, the scene to be stable, and the camera motion to remain continuous. The input degradation contains four relevant components: low-frequency drift caused by platform attitude change, high-frequency vibration caused by wind and motors, local displacement caused by target motion, and jumps in the tracking box caused by occlusion or distractors. Slow drift often reflects intentional operator control and should be preserved. High-frequency jitter and isolated box jumps should be suppressed (As shown in Table 1).

Table 1. Main Symbols Used in the Formulation

Symbol	Meaning
I_t	Grayscale frame at time t
F_t, Γ_t, G_t	Fourier spectrum, cross-power spectrum, and normalized cross-power spectrum at time t
Δ_t	Measured inter-frame translation
M_t	Mapping from image displacement to camera-path coordinates
C_t	Cumulative camera path
R_t	Robust local reference path
N_t	Temporal neighborhood of frame t
S_t	Neighborhood-smoothed camera path
U_t	Stabilization correction
p_t, q_t	Original and stabilized target centers
p^*	Image-center reference
V_t	Frame-centering correction

d_t	Final fused displacement
T_t	Translation affine transform
ϱ_t	Minimum crop scale for border-free output
R_ret	Frame-retention ratio

2.2. Coupling between Stabilization and Tracking

Stabilizing the full frame without target information can preserve a target near the border. Translating the full frame directly with the tracking box can center the target, but tracking-box noise is transferred to the scene and creates visible jumps. Enlarging the crop or applying digital zoom removes border holes while reducing field of view and effective resolution. Therefore, the correction should satisfy three conditions: the camera path changes continuously, the target-center correction is applied gradually, and the total displacement remains inside the available crop.

2.3. Design Constraints

Four constraints define the design. Measurement uses consecutive images only; no IMU, gimbal angular velocity, or UAV telemetry is required. Path smoothing uses a finite temporal neighborhood to support online operation. Centering is applied as a bounded proportional correction rather than a one-frame snap. Every accumulated shift is amplitude-limited, and frame retention is reported with the stabilization result.

2.4. Overall Pipeline

The method consists of target-state input, global-motion estimation, neighborhood path smoothing, center compensation, and boundary-aware warping. The target state comes from an external tracker. Phase correlation estimates global translation. The neighborhood module weights path samples by temporal distance and robust residual. The centering module generates a bounded offset from the tracker output. The stabilization and centering terms are then fused into a single transform with a final displacement limit. Figure 1 presents the pipeline.

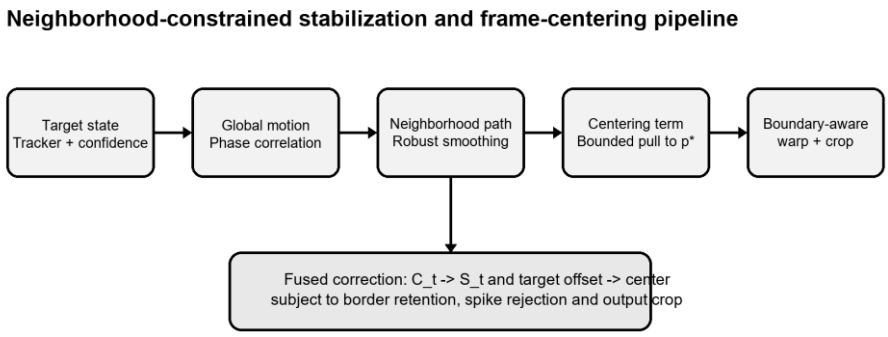


Figure 1. Neighborhood-constrained Stabilization and Frame-Centering Pipeline.

3. Neighborhood-Constrained Stabilization and Frame-Centering Compensation

3.1. Target-State Input

The external tracker provides B_t and a confidence value c_t. The target center is computed as p_t=(x_t+w_t/2,y_t+h_t/2). A three-point median or confidence threshold can reduce spikes caused by low-confidence boxes. When confidence remains below the threshold across several frames, centering is paused and the most recent valid geometric transform is retained. The module uses only the target center and does not require changes to the tracker architecture.

3.2. Global-Motion Estimation

Let I_t and I_{t-1} denote consecutive grayscale frames after downsampling, and let $F_t = \mathcal{F}\{I_t\}$ be their complex-valued two-dimensional Fourier spectra. Define the cross-

power spectrum as $\Gamma_t(\omega) = F_t(\omega)\overline{F_{t-1}}(\omega)$, where the overline denotes complex conjugation. The normalized cross-power spectrum is

$$G_t(\boldsymbol{\omega}) = \frac{\Gamma_t(\boldsymbol{\omega})}{|\Gamma_t(\boldsymbol{\omega})|+\varepsilon} \quad \textbf{(1)}$$

and the inter-frame translation is obtained from the real-valued correlation peak:

$$\Delta_t = \underset{\Delta}{\operatorname{argmax}} \Re[\mathcal{F}^{-1}\{G_t\}(\Delta)] \quad \textbf{(2)}$$

where $\varepsilon > 0$ prevents division by zero, and $\Delta_t = [\Delta_{x,t}, \Delta_{y,t}]^{\mathrm{T}}$ is the measured two-dimensional shift. The 640×360 estimate used in the pilot is mapped back to the original 1280×720 image. The cumulative camera path is

$$\boldsymbol{C}_t = \boldsymbol{C}_{t-1} + \boldsymbol{M}_t\Delta_t \quad \textbf{(3)}$$

where $\mathbf{M}_t$ maps an image displacement into camera-path coordinates. For translation-only estimation, $\mathbf{M}_t$ is the identity matrix. Affine or homography motion models can replace this mapping without changing the subsequent smoothing stage.

3.3. Neighborhood-Constrained Path Smoothing

For the temporal neighborhood $\mathcal{N}_t = \{t-h, \ldots, t+h\}$, a robust local reference is obtained from the component-wise median

$$\mathrm{R_t} = \mathrm{median}\{\mathrm{C_k}: \mathrm{k} \in \mathcal{N}_\mathrm{t}\} \quad \textbf{(4)}$$

Let $r = \|\mathbf{C}_k - \mathbf{R}_t\|_2$ be the residual of a neighboring path sample. The bounded influence function $h(r)$ and the residual-aware weight $w_{t,k}$ are defined as

$$h(\mathrm{r}) = \min\{1, \tau/\max(\mathrm{r}, \tau)\}, \mathrm{w_{t,k}} = \quad \textbf{(5)}$$

$$\exp[-\tfrac{(\mathrm{t}-\mathrm{k})^2}{2\sigma^2}]h(\|\mathrm{C_k} - \mathrm{R_t}\|_2)$$

where $\sigma > 0$ controls temporal decay and $\tau > 0$ sets the residual scale. The influence function is continuous and equals one for residuals up to τ , then decays as τ/r , so isolated phase-correlation failures receive less weight than samples consistent with the local path. The smoothed path is the normalized weighted estimate

$$\mathrm{S_t} = \frac{\sum_{\mathrm{k}\in\mathcal{N}_\mathrm{t}} \mathrm{w_{t,k}C_k}}{\sum_{\mathrm{k}\in\mathcal{N}_\mathrm{t}} \mathrm{w_{t,k}}} \quad \textbf{(6)}$$

The temporal factor favors nearby frames, while the residual factor limits the influence of samples far from the local median. Because the exponential and influence terms are positive for finite residuals, the denominator is nonzero and the normalized estimate is well defined. The filter approaches Gaussian smoothing during normal motion and retains more valid evidence when isolated estimation errors occur.

3.4. Frame-Centering Compensation

The stabilization correction is

$$\mathrm{U_t} = \mathrm{S_t} - \mathrm{C_t} \quad \textbf{(7)}$$

After stabilization, the target center is predicted as $\mathbf{q}_t = \mathbf{p}_t + \mathbf{U}_t$. A bounded centering correction is then generated from the remaining offset:

$$\mathrm{q_t} = \mathrm{p_t} + \mathrm{U_t}, \mathrm{V_t} = \mathrm{clip}\{\alpha(\mathrm{p}^* - \quad \textbf{(8)}$$

$$\mathrm{q_t}), -\mathrm{v_{max}}, \mathrm{v_{max}}\}$$

Here, $\mathrm{clip}\{x, a, b\} = min\{max\{x, a\}, b\}$ is applied component by component, $\alpha \in (0,1]$ is the centering gain, and $\mathbf{v}_{max}$ is the vector of maximum frame-level centering displacements. A larger α accelerates recentering but transfers more tracking-box variation to the scene. The pilot uses $\alpha = 0.50$ and $v_{max,x} = v_{max,y} = 120$ px, which provides gradual correction while remaining compatible with the total displacement limit.

3.5. Correction Fusion and Boundary Protection

The stabilization and centering terms are fused and clipped component-wise to a total displacement limit:

$$d_t = \mathrm{clip}\{U_t + V_t, -d_{max}, d_{max}\}. \quad (9)$$

The corresponding affine transform applied to homogeneous image coordinates is

$$T_t = \begin{bmatrix} 1 & 0 & d_{x,t} \\ 0 & 1 & d_{y,t} \end{bmatrix} \quad (10)$$

If a fixed-size output is required without empty borders, the translated width and height must remain positive. The minimum crop scale is therefore

$$\rho_t = \max\{1, \frac{W}{W-2|d_{x,t}|}, \frac{H}{H-2|d_{y,t}|}\} \quad (11)$$

The crop must satisfy $W - 2|d_{x,t}| > 0$ and $H - 2|d_{y,t}| > 0$. Boundary protection prioritizes the target region and the principal scene content. When both constraints cannot be satisfied, the controller reduces the centering gain or temporarily lowers the stabilization displacement. In deployment, Eq. (11) can be used to crop or scale the output and eliminate empty borders.

3.6. Algorithm Steps and Complexity

For each frame, the algorithm reads the target box, computes its center, estimates inter-frame translation, updates the cumulative path and neighborhood buffer, calculates robust weights and the smoothed path, derives the centering correction and fused displacement, and warps the frame. The main costs are the Fourier transform O(WH log(WH)) and neighborhood weighting O(h), where W and H are the estimation dimensions. A ring buffer of length 2h+1 makes memory independent of total video duration, enabling online implementation.

4. Pilot Evaluation

4.1. Data Source and Preprocessing

Because storage and computation were deliberately limited, a continuous 898-frame prefix of sequence `bike1` was extracted from a public WebDataset mirror of UAV123. UAV123 was introduced by Mueller et al. and contains low-altitude UAV tracking videos with per-frame bounding boxes [22]. The subset retains the native 1280×720 resolution and one-to-one frame-label correspondence. Its total folder size is approximately 87.64 MiB, making it practical for ordinary computers and teaching environments. Table 2 summarizes the data.

Table 2. Lightweight UAV123 Subset Used in the Pilot

Item	Setting
Source dataset	UAV123
Sequence	bike1
Frame range	000001-000898
Included frames	898
Original sequence length	3,085 frames
Resolution	1280×720
Annotation	x, y, width, height per frame
Initial template	Frame 000001
Subset size	approximately 87.64 MiB

This subset is a workflow-validation artifact and not a replacement for the full UAV123 benchmark. To support a general benchmark claim, the experiment should be repeated on all 123 UAV123 sequences and UAV20L, and extended to DTB70 to test different flight conditions.

4.2. Compared Variants and Parameters

Four variants were compared: the original sequence, fixed Gaussian path smoothing, centering compensation alone, and neighborhood-robust smoothing with centering. Fixed Gaussian smoothing represents a conventional path-filter baseline. Centering alone

isolates the contribution of the target-position term. The neighborhood stabilization ablation isolates the effect of robust path weights. Table 3 lists the parameters.

Table 3. Pilot Parameters

Parameter	Meaning	Value
Motion-estimation resolution	Phase-correlation input	640×360
Neighborhood half-width h	One-sided temporal radius	7 frames
Temporal scale σ	Gaussian temporal decay	4.0 frames
Residual threshold τ	Huber-type down-weighting threshold	18 px
Stabilization limit	Maximum frame-level stabilization shift	28 px
Centering gain α	Target-recentering rate	0.50
Centering limit v_max	Maximum frame-level centering shift	120 px
Total limit d_max	Maximum fused shift	148 px

4.3. Metrics

Center-error RMS (RMSE_c) measures the distance between the compensated target center and the image center. The 95th percentile center error p95_c captures large-offset frames. Safety-region coverage is the proportion of frames whose target center lies inside a central square whose side is 25% of the shorter image dimension. Frame retention R_ret is computed from the translated width and height. Motion-stress recovery error RMSE_m compares the processed trajectory with the known noise-free trajectory.

4.4. Real-Sequence Results

Table 4 reports the geometric results. The original sequence has RMSE_c=164.58 px, p95 center error=246.92 px, and safety-region coverage=65.26%. Fixed Gaussian stabilization changes the background path but provides limited recentering. Centering alone reduces RMSE_c to 82.80 px (49.69% reduction). The fused method reaches 82.58 px (49.82% reduction), brings all pilot frames into the defined central safety region, and maintains 90.36% frame retention. The similarity between centering-only and fused results indicates that the stabilization displacement is small on this gimbal-stabilized UAV123 sequence, which is consistent with its stable source geometry.

Table 4. Pilot Results on the Real UAV123 Subsequence

Method	Center RMSE (px)	p95 error (px)	Safety-region coverage (%)	Frame retention (%)	Mean absolute shift (px)
Original	164.58	246.92	65.26	100.00	0.00
Fixed Gaussian stabilization	163.93	244.36	64.59	99.69	1.54
Centering compensation only	82.80	123.46	100.00	90.38	49.05
Neighborhood stabilization only	164.36	246.70	65.03	99.84	0.81
Proposed fused method	82.58	123.24	100.00	90.36	49.15

Figure 2 reports cumulative coverage at representative target-center error thresholds. The centering-related methods substantially increase the fraction of frames below 120 px.

The fused curve nearly coincides with centering alone, indicating that the normalized stabilization increment does not disrupt the centering trend. The approximately 90% retention rate makes the content cost visible. In deployment, the safety-region size and maximum shift should be chosen according to the required field of view.

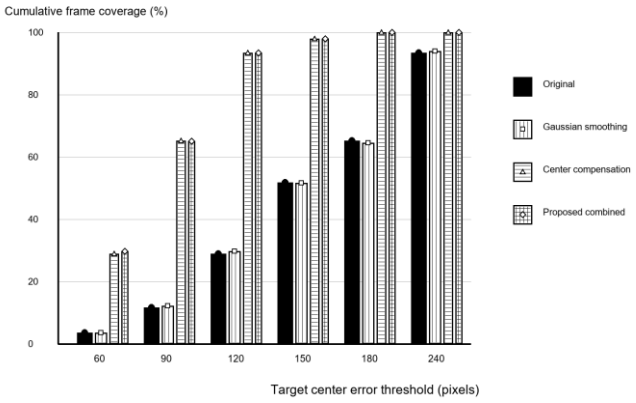


Figure 2. Cumulative Coverage at Representative Target-Center Error Thresholds; Methods Are Distinguished by Bar Width, Fill Pattern, Marker Shape, and Marker Size.

4.5. Injected Motion-Noise Stress Test

To test suppression of sporadic estimation errors, Gaussian noise with standard deviations of 2, 4, and 8 px was added to the real target trajectory. Sparse spikes approximately five times the noise level were inserted in 5% of frames. Each condition was repeated with ten random seeds, and the mean trajectory-recovery error was recorded. Table 5 presents the results. At 2, 4, and 8 px noise, the raw RMSEs are 4.10, 8.19, and 16.39 px. Fixed Gaussian smoothing reduces these values to 2.16, 2.92, and 4.86 px. Neighborhood-robust smoothing further reduces them to 1.72, 2.56, and 4.22 px, corresponding to reductions of 57.95%, 68.70%, and 74.26% relative to the raw input. The advantage grows with noise amplitude because residual down-weighting reduces the influence of sparse spikes (As shown in Figure 3).

Table 5. Trajectory Recovery under Injected Motion Noise

Injected noise	Raw RMSE (px)	Gaussian RMSE (px)	Neighborhood RMSE (px)	Reduction vs raw (%)
2 px	4.10	2.16	1.72	57.95
4 px	8.19	2.92	2.56	68.70
8 px	16.39	4.86	4.22	74.26

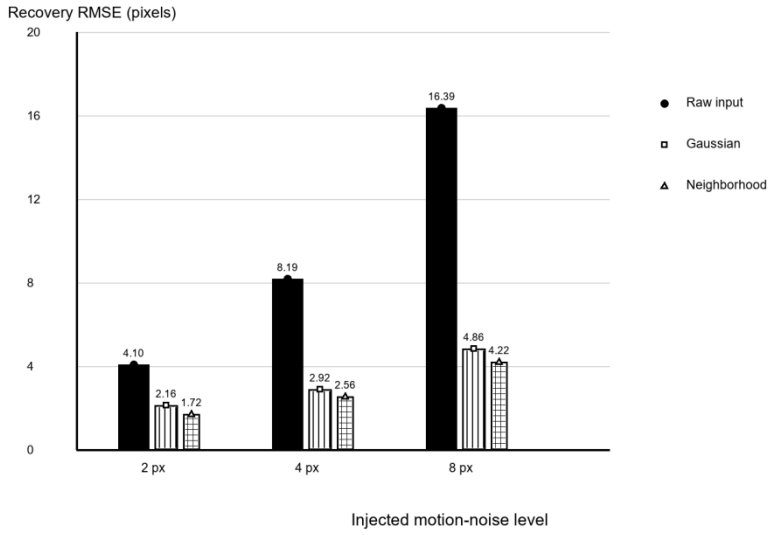


Figure 3. Trajectory-recovery Error under Injected Motion Noise; Methods Are Distinguished by Bar Width, Fill Pattern, Marker Shape, and Marker Size.

4.6. Computational Cost and Ablation

On CPU, phase-correlation motion estimation for 898 frames at 640×360 took 33.72 s, corresponding to 26.63 frames/s. This figure includes motion estimation and trajectory computation but excludes tracker inference, image warping, and video encoding; it is therefore not an end-to-end FPS measurement. The ablation shows that neighborhood stabilization alone barely changes center error, whereas centering alone produces most of the center-error reduction. The fused method improves slightly over centering alone. This division is intentional: centering controls target observability, while neighborhood stabilization suppresses camera jitter and abnormal path measurements.

4.7. Positioning Relative to Tracking Models

The pilot is not compared directly with published accuracy of OSTrack, ODTrack, or AVTrack. Those comparisons require matched training data, hardware, pretrained weights, and benchmark protocols. The present method is a geometric post-processing layer. Its tracking value should therefore be tested by running each tracker with and without compensation under identical inputs. Table 6 defines the full evaluation plan.

Table 6. Full Validation Plan

Dimension	Settings	Reported metrics
General trackers	OSTrack, ODTrack, AVTrack, SiamDMU	Precision, Success AUC, FPS
Applied baselines	YOLO+KCF, lightweight ViT tracker	Center error, AUC, onboard resource use
Stabilization baselines	None, fixed Gaussian, L1 path smoothing	Trajectory RMSE, residual jitter
Proposed variants	Neighborhood stabilization, centering, fused	All above and frame retention
Datasets	UAV123, UAV20L, DTB70	Overall and attribute-wise results
Hardware	Desktop CPU/GPU and Jetson-class edge device	End-to-end latency, power, memory

5. Discussion

5.1. Engineering Value

The principal engineering value is that tracker output becomes a soft constraint on stabilization without requiring a new tracker or a motorized gimbal. Existing detection and tracking pipelines only need a trajectory buffer, displacement limiter, and image-warping module at the output stage. Keeping the target near the center benefits manual interpretation, automatic recognition, and long-term tracking. Neighborhood weighting reduces the effect of isolated phase-correlation failures on video continuity.

5.2. Parameter Selection and Applicability

The neighborhood half-width h controls the smoothing horizon. A small h retains high-frequency jitter; a large h may classify intentional turning as jitter and introduce lag. The pilot uses h=7 over 898 frames, which covers short spikes without heavily altering slow drift. The centering gain α and center limit v_max jointly determine recentering speed and frame retention. Applications that prioritize field of view should reduce α. Applications that prioritize keeping the target central can increase α and pair it with digital zoom or a larger search area.

5.3. Limitations

Three limitations remain. The full experiment does not yet cover all UAV123 sequences, UAV20L, or DTB70, so one sequence cannot represent all aircraft, terrain, and weather conditions. The real-sequence test uses annotated boxes as a target-state proxy and does not yet report Precision or Success AUC changes for a fixed tracker. Motion estimation uses a translation model and is less capable under significant roll, pitch, or scale change. Sparse texture and large-area occlusion can also corrupt phase correlation.

5.4. Full Validation Plan

For a broader performance evaluation, three additional tests should be completed. First, run OSTrack, ODTrack, AVTrack, and reproducible applied baselines on the complete UAV123, UAV20L, and DTB70 datasets with common initial boxes and evaluation scripts. Second, evaluate every tracker under four conditions: original video, fixed-path smoothing, neighborhood stabilization, and the fused method. Report Precision, Success AUC, center error, safety-region coverage, frame retention, and FPS. Third, measure total latency, power, and memory on a Jetson-class device and add night, strong-wind, low-texture, and camera-occlusion scenarios.

6. Conclusions and Recommendations

6.1. Implementation Recommendations

Deployment can follow an estimate-compensate-crop structure. When the platform is stable, the centering gain should be reduced to avoid unnecessary translation. When the target approaches a boundary or tracker confidence decreases, the relative priority of centering can increase while the single-frame displacement remains limited. If the box becomes abnormal for a short interval, centering should pause while the camera path is maintained. After complete tracking failure, the most recent valid transform should be retained until re-detection. Both original and stabilized streams should be archived to support later inspection.

6.2. Conclusions and Outlook

This study proposed a neighborhood-constrained stabilization and frame-centering compensation method for UAV target tracking. Global motion is estimated by phase correlation, a robust camera path is constructed from neighborhood medians, temporal weights, and residual down-weighting, and a bounded centering term is fused with the stabilization shift. On an 898-frame public subset, the method reduced center-error RMS by 49.82%, brought all pilot frames into the defined central safety region, and retained 90.36% of frame content. Injected-noise tests reduced trajectory-recovery error by 57.95% to 74.26%. These results support the method as a geometric post-processing layer for existing trackers. Future work will expand the benchmark, test affine and homography motion models, and validate onboard implementation and the trade-off between target centering, intentional camera motion, and scene preservation.

Data Availability Statement: The UAV123 source data were released by King Abdullah University of Science and Technology. The pilot uses an 898-frame prefix of sequence `bike1`, extracted from a public WebDataset mirror. Provenance, frame range, annotations, and checksums accompany the subset. The author will provide the subset and test scripts subject to the original dataset terms.

References

1. B. Ye, H. Chang, B. Ma, S. Shan, and X. Chen, "Joint feature learning and relation modeling for tracking: a one-stream framework," in *European Conference on Computer Vision (ECCV)*, 2022.
2. Y. Zheng, B. Zhong, Q. Liang, Z. Mo, S. Zhang, and X. Li, "ODTrack: online dense temporal token learning for visual tracking," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 7, pp. 7588–7596, 2024.
3. Y. Wu, Y. Li, M. Liu, X. Wang, X. Yang, H. Ye, et al., "Learning an adaptive and view-invariant vision transformer for real-time UAV tracking," *IEEE Trans. Circuits Syst. Video Technol.*, 2025.
4. J. Liu, H. Wang, C. Ma, Y. Su, and X. Yang, "SiamDMU: Siamese dual mask update network for visual object tracking," *IEEE Trans. Emerg. Topics Comput. Intell.*, 2024.
5. M. Grundmann, V. Kwatra, and I. Essa, "Auto-directed video stabilization with robust L1 optimal camera paths," in *IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2011.
6. S. Jeon, I. Yoon, J. Jang, S. Yang, J. Kim, and J. Paik, "Robust video stabilization using particle keypoint update and l1-optimized camera path," *Sensors*, vol. 17, no. 2, p. 337, 2017.
7. Y. G. Lee, "A novel camera path planning algorithm for real-time video stabilization," *EURASIP J. Image Video Process.*, 2019.
8. M. R. Souza and H. Pedrini, "Digital video stabilization: algorithms and evaluation," in *SIBGRAPI Conf. Graphics, Patterns Images*, 2019.
9. M. R. Souza, H. A. Maia, and H. Pedrini, "Rethinking two-dimensional camera motion estimation assessment for digital video stabilization: a camera motion field-based metric," *Neurocomputing*, p. 126768, 2023.
10. Q. Huang, J. Liu, C. Jiang, and Y. Wang, "DMCVS: decomposed motion compensation-based video stabilization," *IET Image Process.*, 2024.
11. L. Yang, Z. Liu, G. Wang, and X. Wang, "Image-based visual servo control for ground target tracking using a fixed-wing UAV with pan-tilt camera," in *Int. Conf. Unmanned Aircraft Syst. (ICUAS)*, 2020.
12. L. W. Krakow, C. M. Eaton, and E. K. P. Chong, "Simultaneous non-myopic optimization of UAV guidance and camera gimbal control for target tracking," in *IEEE Conf. Control Technol. Appl. (CCTA)*, 2018.
13. P. Nousi, I. Mademlis, I. Karakostas, A. Tefas, and I. Pitas, "Embedded UAV real-time visual object detection and tracking," in *IEEE Int. Conf. Real-time Comput. Robot. (RCAR)*, 2019.
14. T. Keawboontan and M. Thammawichai, "Toward real-time UAV multi-target tracking using joint detection and tracking," *IEEE Access*, 2023.
15. Z. Liang, J. Wang, G. Xiao, and L. Zeng, "FAANet: feature-aligned attention network for real-time multiple object tracking in UAV videos," *Chin. Opt. Lett.*, vol. 20, p. 081101, 2022.
16. M. Zhu, H. Zhang, J. Zhang, and L. Zhuo, "Multi-level prediction Siamese network for real-time UAV visual tracking," *Image Vis. Comput.*, p. 104002, 2020.
17. Y. Lu and L. Sun, "UAV target tracking algorithm based on kernel correlation filter," in *Chinese Control Decis. Conf.*, 2023.
18. P. Zhang, J. Zhao, D. Wang, H. Lu, and X. Ruan, "Visible-thermal UAV tracking: a large-scale benchmark and new baseline," in *IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2022.
19. H. Yu, K. Su, and L. Wang, "A lightweight UAV target tracking method against occlusion and viewpoint changes," *Automation Appl.*, 2026.
20. D. Yu, H. Wang, S. Liu, J. Li, S. Gao, and Y. Feng, "Research on lightweight monocular UAV target tracking based on KCF," *Commun. Inf. Technol.*, 2026.
21. M. Zhao, B. Zheng, H. Sun, M. Ma, C. Song, and J. Bai, "UAV aerial target tracking combining attention and anti-occlusion strategies," *Chin. J. Liquid Crystals Displays*, 2026.
22. M. Mueller, N. Smith, and B. Ghanem, "A benchmark and simulator for UAV tracking," in *Computer Vision – ECCV 2016*, Cham: Springer, 2016, pp. 445–461.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of Publisher and/or the editor(s). Publisher and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.