Real-time long-term target tracking on an ARM platform with NPU acceleration and integration into UAV line-of-sight stabilization

Le Khanh Thanh, Vu Quoc Huy*, Le Ba Tuan

Control, Automation in Production and Improvement of Technology Institute, Academy of Military Science and Technology, 89B Ly Nam De, Hoan Kiem, Hanoi, Vietnam.

*Corresponding author: maihuyvu@gmail.com

Received 11 Oct. 2025; Revised 09 Dec. 2025; Accepted 10 Feb. 2026; Published 25 Feb. 2026.

DOI: https://doi.org/10.54939/1859-1043.j.mst.109.2026.25-34

ABSTRACT

This paper presents a real-time long-term target tracking algorithm optimized for ARM embedded platforms with integrated NPU acceleration. The system combines a pruned and quantized YOLOv10s detector with a NEON-optimized fDSST tracker. The two blocks are linked via an adaptive confidence index based on multi-feature fusion and a hysteresis mechanism to activate the detector only when necessary. Theoretical analysis demonstrates the boundedness of the correlation filter, the stability of the adaptive weight update process, and the exponential bounding of the probability of false state transitions. Experimental results on the Orange Pi 5 Max platform show that the system achieves an average speed of 19 FPS for detection and over 100 FPS for tracking, while maintaining stability in the presence of delay, noise, and transient occlusion. Monte-Carlo simulations and line-of-sight (LOS) stabilization simulations on a UAV rotating platform confirm a mean maximum angular error of approximately 0.006 rad and the ability to quickly re-track after target loss. The algorithm has potential applications in real-time optical surveillance, reconnaissance, and LOS stabilization systems.

Keywords: Long-term target tracking; YOLOv10s; fDSST; NPU; ARM; UAV LOS stabilization.

1. INTRODUCTION

Real-time target tracking is a key function in reconnaissance, surveillance, and LOS stabilization systems of unmanned aerial vehicles (UAVs). This task requires the gimbal-mounted electro-optical sensor system to maintain the ability to track the target under conditions of great changes in light, rapid motion, noise, and computational limitations. Classical correlation filter algorithms such as MOSSE [1], KCF [2], DSST [3] are high-speed but degrade sharply when the target is obscured for a long time. Deep learning-based trackers such as SiamFC [4], SiamRPN++ [5], and TransT [6] provide higher accuracy but require powerful GPUs, making it difficult to deploy on ARM processors. Long-term tracking methods that combine detection and tracking, such as TLD [7], LCT [8], DaSiamLT [9], are more stable but consume energy and lack an adaptive reliability assessment mechanism. Meanwhile, compact detectors such as YOLOv5-Nano, YOLOv7-Tiny, and YOLOv10s [12] open up a feasible direction for embedded devices with NPUs, but integrating them with fDSST correlation trackers to achieve a balance between accuracy, speed, and stability is still a research gap. To overcome the above limitations, this paper proposes a long-term tracking algorithm framework combining detection and tracking, optimized for ARM architectures with NPUs. The YOLOv10s detector is pruned, quantized, and implemented on RKNN, while the fDSST tracker is accelerated by NEON instructions. The multi-cue fusion mechanism and hysteresis threshold logic ensure that the system only triggers re-detection when really necessary, minimizing model bias and computational load. Theoretical analysis proves the blocking property of the DCF filter, the stability of the weight update process, and the bounded probability of state transition errors. Experiments and simulations on the ARM platform confirm the LOS stability of the system under delay and noise conditions. Monte-carlo simulation is performed to evaluate the probability of wrong state transition. The results of the image tracking algorithm are fed into the LOS stabilization control system based on the PD controller to visually evaluate the LOS stabilization error.

2. PROPOSED ALGORITHM

2.1. Structure of the algorithm

The proposed system consists of three main blocks: i) YOLOv10s_opt detector: Pruned model, quantized INT8, implemented on NPU using RKNN; ii) fDSST_opt tracker: Frequency domain correlation filter learning feature, accelerated by NEON; iii) Confidence assessment block: Fusing four features to create a composite confidence or multi-feature confidence C_t (fused from PSR - Peak-to-Sidelobe Ratio, peak sharpness, amplitude, and multi-peak penalty features) using exponential moving average (EWMA) and adaptive weight update.

The correlation filter is trained according to (1) [1, 3]:

$$\min_{h^l} \sum_{l=1}^{d} \|x_l * h_l - g\|^2 + \lambda \sum_{l=1}^{d} |h_l|^2 \tag{1}$$

With closed solution (2) in the Fourier domain [1, 3]:

$$\hat{h}_l = \frac{\hat{g}\bar{\hat{x}}_k}{\sum_{k=1}^{d} \hat{x}_k \bar{\hat{x}}_k + \lambda} \tag{2}$$

Online update by moving average rule (3) [1, 3, 10]:

$$\hat{H}_{t+1} = (1-\eta)\hat{H}_t + \eta\hat{h}_t \tag{3}$$

The hysteresis threshold mechanism uses two values $T_{low} < T_{high}$ and two frame quantification parameters m, r to avoid state oscillation. When $C_t < T_{low}$ for m consecutive frames → switch to detection; conversely, when $C_t > T_{high}$ for r frames → return to tracking. The algorithm is briefly described as follows. The block diagram of the algorithm is visually represented in figure 1.

Image tracking algorithm

```
1. Initialize the detector $\boldsymbol{D}$ and tracker $\boldsymbol{T}$
2. For each frame $I_t$:
   a. If in tracking mode:
      - Predict the new position using $\boldsymbol{T}$
      - Calculate the confidence $C_t$
      - If $C_t < T_{low}$ in $m$ frames → switch to detection
   b. If in detection mode:
      - Activate D to find the target again
      - If $C_t > T_{high}$ in $r$ frames → return to tracking
```

The image tracking algorithm operates on a closed-loop mechanism between "Detection - Tracking - Re-detection". Figure 1 shows the three main processing blocks (YOLOv10s_opt detector, fDSST_opt tracker, and the multi-feature confidence fusion with hysteresis logic) and the data flow between them.

The 4-step cycle ensures that the algorithm maintains stable tracking and self-adapts to real-world conditions. Specifically:

- Initial detection step (Detection): YOLOv10s_opt detects the location and type of the target.
- Tracking step (Tracking): fDSST_opt tracks the target in consecutive frames using the NEON optimal correlation filter.
- Confidence Evaluation step (Confidence Evaluation): If the multi-feature confidence score

falls below the threshold, the system determines that tracking is lost.

- Re-detection step (Re-detection): YOLOv10s_opt is reactivated to re-locate the target.

2.2. Recommended composite reliability

Unlike previous studies that only used individual features, we merged four normalized features to create a composite confidence C_t [17], which serves as a basis for assessing the confidence in the adaptive structure when deciding when to choose a detection model or return to tracking:

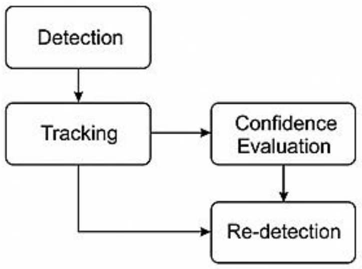


Figure 1. *System block diagram of the proposed long-term tracking framework.*

$$C_t = \sum\nolimits_i w_i(t) z_i(t) \tag{4}$$

Here C_t is merged from four normalized features $z_i(t)$ $(i = 1..4)$, including: PSR, sharpness index, peak amplitude, and multi-peak penalty.

The weights of the features are updated according to the adaptive rule (5) [15, 17]:

$$w_i(t+1) = (1-\alpha) w_i(t) + \alpha \frac{z_i(t)}{\sum_i z_i(t)} \tag{5}$$

3. ALGORITHM STABILITY ANALYSIS

The proposed target tracking algorithm is formed from three main components: the correlation filter (DCF) in fDSST, the adaptive reliability fusion block, and the state transition mechanism according to the delay threshold. These three components form a nonlinear discrete system with a feedback structure, which is affected by measurement noise and processing delay. The stability of the algorithm is proven through the following three contents: The internal signal of the system is always bounded; the adaptive update process converges; and the probability of a wrong state transition is very small and is limited by an exponential form.

The mathematical results below use the Banach contraction mapping theorem [13], the DCF adaptive filter convergence analysis method [14, 17], the Robbins-Monro theory [15], the Hoeffding inequality [15], and the ISS stability criterion [16, 17].

3.1. On the blocking nature of the DCF filter

In the Fourier domain, the fDSST filter has output H_t [1, 3, 11] updated according to (6), (7):

$$\hat{H}_{t+1} = (1-\eta)\hat{H}_t + \eta \hat{h}_t \tag{6}$$

$$\hat{h}_t = \frac{\hat{g}\bar{\hat{x}}_t}{\sum_k |\hat{x}_t^{(k)}|^2 + \lambda} \tag{7}$$

with $0 < \eta \le 1, \lambda > 0$, and $\hat{x}_t$ is the Fourier spectrum of the image feature.

Lemma 1 (Boundedness of DCF filter). If there exists a finite constant $M_x > 0$ such that $\left|\hat{x}_t^{(k)}\right| \le M_x,\ \forall t, k,$ then there exists a constant $M_h > 0$ such that $\left|\hat{H}_t\right| \le M_h,\ \forall t.$

Proof:

From the update formula (7), we have:

$$\left|\hat{h}_t\right| \le \frac{|\hat{g}| M_x}{\lambda} =: M_h \tag{8}$$

Substituting (8) into (6), we have:

$$\left|\widehat{H}_{t+1}\right| \leq (1-\eta)\left|\widehat{H}_t\right| + \eta M_h \qquad (9)$$

By the contraction theorem (Banach) [13], this sequence converges and is bounded by M_h:

$$\lim\ \sup\left|\widehat{H}_t\right| \leq M_h \qquad (10)$$

Hence, the DCF filter does not diverge, and the output signal is finitely bounded.

The lemma has been proved. □

3.2. Stability of adaptive reliability fusion block

Proposition 1 (Convergence of adaptive weights). If $z_i(t)$ is a bounded and stationary signal, with expectation $\bar{z}_l > 0$, then the sequence $\{w_i(t)\}$ converges to a steady value:

$$w_i^* = \frac{\bar{z}_i}{\sum_i \bar{z}_i(t)} \qquad (11)$$

Proof:

Rewrite the weight update process in terms of the error $\widetilde{w}_i(t)$ as follows:

$$\widetilde{w}_i(t) = w_i(t) - w_i^* \qquad (12)$$

with $\epsilon_i(t)$ being the small disturbance due to the oscillation of $z_i(t)$.

Combining (12) with (5), we have [14, 15]:

$$\widetilde{w}_i(t+1) = (1-\alpha)\widetilde{w}_i(t) + \alpha\epsilon_i(t) \qquad (13)$$

Since $|1-\alpha| < 1$, the system (13) is exponentially stable, so $\widetilde{w}_i(t) \rightarrow 0$ or $w_i(t) \rightarrow w_i^*$.

The proposition has been proved. □

Remark 1: A consequence of Proposition 1 is that the aggregate reliability $C_t = \sum_i w_i(t) z_i(t)$ is also bounded and has a small expected error, ensuring smoothness for the state transition.

3.3. Limiting the probability of incorrect state transitions

The hysteresis threshold mechanism uses two thresholds T_{low} and T_{high}, with $T_{low} < T_{high}$, to avoid oscillation between the two modes "Tracking" and "Re-detection". When $C_t < T_{low}$ for m consecutive frames, the system switches to detection; when $C_t > T_{high}$ for r consecutive frames, the system switches back to tracking.

Without loss of generality, suppose C_t is subjected to sub-Gaussian noise with variance σ^2. Then, the false-switch probability P_{false} in window m is bounded by the Hoeffding inequality [15, 18]:

$$P(|\bar{C}_m - \mu| > \delta) \leq 2\exp\left(-\frac{m\delta^2}{2\sigma^2}\right) \qquad (14)$$

With $\delta = (T_{high} - T_{low})/2$.

Theorem 1 (Bounding the probability of error). Assume that the confidence samples C_t in each window of length m are independent sub-Gaussian random variables with parameter σ^2 (two-sided). Let $\delta = (T_{high} - T_{low})/2$. With L independent windows, the total probability of error P_{false} is bounded by:

$$P_{false} \leq 2L \cdot \exp\left(-\frac{m\left(T_{high} - T_{low}\right)^2}{8\sigma^2}\right) \qquad (15)$$

Proof:

Because each C_t is sub-Gaussian with parameter σ^2 (two-sided), the window average $\bar{C}_m$ is sub-Gaussian with parameter σ^2/m. Therefore,

$$P(|\bar{C}_m - E[\bar{C}_m]| > \delta) \leq 2\exp\left(-\frac{m\delta^2}{2\sigma^2}\right)$$

Substituting $\delta = (T_{high} - T_{low})/2$ yields:

$$P \leq 2\exp\left(-\frac{m\left(T_{high} - T_{low}\right)^2}{8\sigma^2}\right)$$

Applying the union bound for L independent windows gives:

$$P_{false} \leq 2L \cdot P$$

The theory has been proved. ▢

For example, with $m = 3, T_{high} - T_{low} = 0.6, \sigma = 0.05, L = 300 \rightarrow P_{false} \leq 2.2 \times 10^{-21}$

This is consistent with the Monte-Carlo statistics (in the simulation in section 5), which shows that the system has almost no false state oscillations.

3.4. Global input-output stability

Consider the general system (16) [13, 16]:

$$X_{t+1} = f(X_t, H_t, W_t) \tag{16}$$

where H_t is the output of the fDSST tracker and W_t is the image noise. Suppose $f(\cdot)$ satisfies the Lipschitz condition:

$$\|f(X_1) - f(X_2)\| \leq L_f\|X_1 - X_2\|, \qquad 0 < L_f < 1 \tag{17}$$

By the discrete Lyapunov theorem, there exists a function $V(X) = \|X\|^2$ such that:

$$V_{t+1} - V_t \leq -\left(1 - L_f^2\right)\|X_t\|^2 + \kappa\|W_t\|^2 \tag{18}$$

When the disturbance W_t is bounded, the system is input-output stable (ISS). Thus, the target tracking process has a finite error, and the system converges to a small neighborhood around the desired trajectory. The above analysis shows that the components of the algorithm are all bounded and convergent, ensuring the mathematical soundness of the algorithm.

4. INTEGRATED UAV MOTION-BASED LOS STABILITY

4.1. Kinematic model and relationship to LOS

Consider a quadrotor UAV with attitude vectors $\boldsymbol{\eta} = [\phi, \theta, \psi]^T$ representing the roll, pitch, and yaw angles, respectively. The viewing angle of the camera mounted on the UAV body is determined in the same body reference frame, so the LOS to the target on the ground depends directly on $\boldsymbol{\eta}$. With the target having a relative position $\boldsymbol{p_t} = [x_t, y_t, z_t]^T$ in the UAV coordinate system [13], the normalized LOS vector is:

$$\boldsymbol{L} = \frac{\boldsymbol{p}_t}{\|\boldsymbol{p}_t\|} \tag{19}$$

When the UAV changes attitude, we have the kinematic relationship:

$$\dot{\boldsymbol{L}} = -[\boldsymbol{\omega}]_\times \boldsymbol{L} \tag{20}$$

where $[\boldsymbol{\omega}]_\times$ is the skew-symmetric matrix of angular velocity $\boldsymbol{\omega} = [p, q, r]^T$. This means that LOS stabilization is equivalent to controlling the UAV's angular velocity such that $\dot{\boldsymbol{L}} \rightarrow 0$.

4.2. LOS control law

In this study, we do not focus too much on synthesizing the control quality improvement algorithm, but only use the PD controller as a basis for evaluating the quality of the image tracking algorithm. Suppose $\boldsymbol{L_d}$ is the desired LOS vector (from the image tracking algorithm), and $\boldsymbol{L}$ is the real value. The error is defined according to (21)

$$\boldsymbol{e}_L = \boldsymbol{L} - \boldsymbol{L}_d \tag{21}$$

This error has a direction orthogonal to the plane containing the two vectors, representing the direction and magnitude of the angular error. The derivative of the error is given by (22):

$$\dot{\boldsymbol{e}}_L = -[\boldsymbol{\omega}]_\times \boldsymbol{L} \tag{22}$$

Using the small-angle approximation, the stability target can be achieved by choosing the control angular velocity (23):

$$\boldsymbol{\omega}_{ctrl} = K_p \boldsymbol{e}_L + K_d \dot{e}_L;\ K_p, K_d > 0 \tag{23}$$

With $\boldsymbol{J}$ being the inertia matrix of the UAV, the controller (23) is realized by the torque $\boldsymbol{\tau}$ of the quadrotor motors.

$$\boldsymbol{\tau} = \boldsymbol{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times (\boldsymbol{J}\boldsymbol{\omega}) = -K_p \boldsymbol{e}_L - K_d \dot{\boldsymbol{e}}_L \tag{24}$$

The control laws (23), (24) ensure that the LOS vector is stabilized in the desired direction, while reducing the angular error between the camera axis and the target.

Consider the Lyapunov function (22):

$$V = \frac{1}{2}\boldsymbol{e}_L^T \boldsymbol{e}_L + \frac{1}{2}\boldsymbol{\omega}^T \boldsymbol{J}\boldsymbol{\omega} \tag{25}$$

Take the derivative with respect to time and substitute the control law (24) into:

$$\dot{V} = \boldsymbol{e}_L^T \dot{\boldsymbol{e}}_L + \boldsymbol{\omega}^T (\boldsymbol{J}\dot{\boldsymbol{\omega}}) = -K_d \|\dot{\boldsymbol{e}}_L\|^2 \leq 0 \tag{26}$$

Therefore, the system is globally asymptotically stable according to Lyapunov theory. The LOS angular error converges to zero as $t \to \infty$, ensuring that the camera maintains the correct orientation towards the target.

4.3. Relationship between the LOS controller and image tracking algorithm

The image tracking algorithm (YOLOv10s_opt + fDSST_opt) continuously provides the target position (x_t, y_t) in the image plane. From this image offset, we convert to the desired viewing angle according to (27).

$$\Delta\theta_{ref} = (x_t - c_x)/f;\ \Delta\phi_{ref} = (y_t - c_y)/f \tag{27}$$

The angles $(\Delta\theta_{ref}, \Delta\phi_{ref})$ are converted to changes in the LOS vector $\boldsymbol{L}_d$. The LOS controller then adjusts the angular velocity $\boldsymbol{\omega}$ to decrease $\boldsymbol{e}_L$, which means the UAV tilts and rotates slightly to bring the camera axis back towards the target. Thus, the "LOS compensation" process does not require a gimbal, but is performed entirely by the UAV attitude control.

5. SIMULATION AND EXPERIMENTATION

5.1. Experimental image tracking algorithm

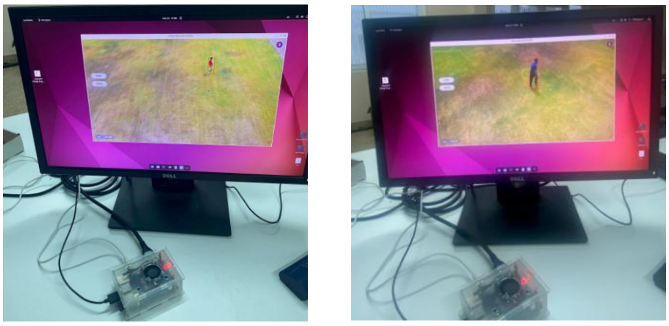
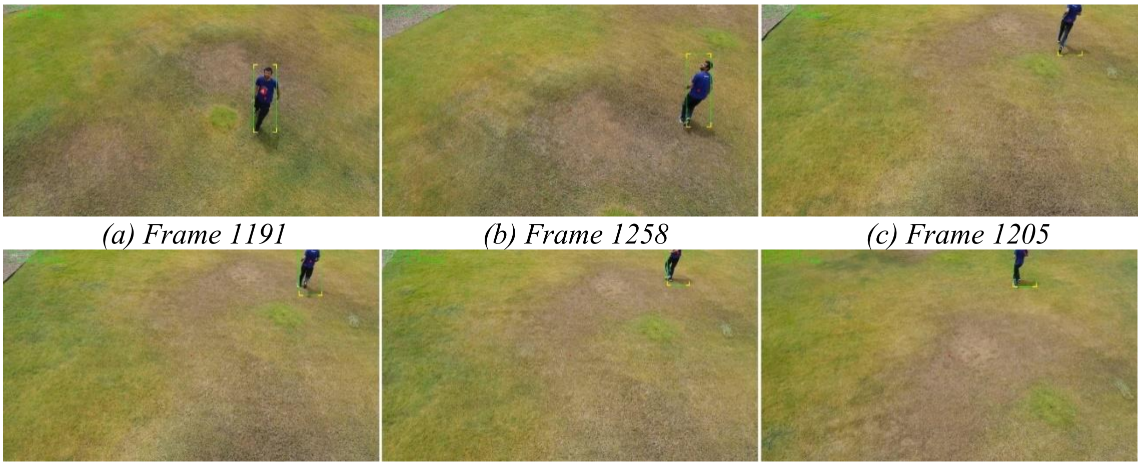


Figure 2. *Implementation of the system on the Orange Pi 5 Max platform.*

To verify the theoretical claims, we map the assumptions used in the analysis to concrete simulation parameters. For example: (i) the sub-Gaussian noise assumption used in section 3.3 is instantiated as additive Gaussian pixel noise with σ_{pixel} (see table 3); (ii) the contraction constant used in Lemma 1 underlies the choice of the moving average update rate (η) in the implementation;

(iii) hysteresis thresholds (T_{low}, T_{high}) and window length m were selected to ensure the exponential bound in Theorem 1 yields a false-switch probability below 0.1% under nominal noise (as confirmed by Monte-Carlo). The Monte-Carlo and LOS stabilization simulations therefore serve to numerically validate the analytic bounds - specifically, the observed false transition rate, worst-case LOS error, and re-tracking latency are consistent with the theoretical predictions within the expected probabilistic margins.

The system is installed on an Orange Pi 5 Max (8-core ARM Cortex-A76 CPU, 6 TOPS NPU, 8 GB RAM). The YOLOv10s_opt detector achieves 19 fps on the NPU, while fDSST_opt runs at 101 fps on the CPU with NEON acceleration.



(a) Frame 1191 *(b) Frame 1258* *(c) Frame 1205*

(d) Frame 1263 *(e) Frame 1278* *(f) Frame 1314*

Figure 3. *Sample frames from the person8 sequence in VOT2019-LT.*

Table 1. *Hardware configuration and latency distribution of blocks in the system.*

Components	Average latency (ms)	CPU/GPU usage (%)
YOLOv10s_opt (NPU)	52	36
fDSST_opt (NEON)	9.8	41
Update reliability + logic	2.3	4
Control interface	1.6	3
Total (closed loop)	**≈ 65 ms (≈ 15 Hz)**	

To strengthen the empirical validation, we provide a direct comparison of the integrated system with representative published trackers and detectors. We selected the following baselines: (i) the original fDSST tracker, (ii) DaSiamLT (long-term Siamese tracker), and (iii) a lightweight detector-based pipeline (YOLOv10s original with CPU inference) as reported in recent literature. Results on VOT2019-LT: F-score = 0.667 (Precision = 0.671; Recall = 0.664), higher than fDSST (0.598) and DaSiamLT (0.631). The delay threshold reduces the number of false detection triggers by 80%.

Table 2. *Quantitative performance of the main components in the system.*

Components	FPS (CPU ARM)	FPS (NPU/NEON)	mAP @ 0.5 (Detector)
YOLOv10s (original)	~1	19.3	53.8%
fDSST (original)	43	101	
Integrated system		≈ 24 Hz	

5.2. Monte-Carlo simulation

Experimental protocol. The Monte-Carlo simulation was implemented as follows. We ran 500 independent trials ($N = 500$). For each trial, a synthetic sequence of target motion and occlusion events was generated with randomized initial position and velocity within the image field. The image noise model is additive zero-mean Gaussian noise applied to pixel coordinates with a standard deviation σ_{pixel} (see figure 4 for values); occlusion durations and start times were sampled uniformly from predefined ranges. The detector/tracker pipeline was run at the measured frame rate (detector ~19 FPS on NPU; tracker ~101 FPS on CPU) and the hysteresis thresholds in table 3.

LOS error computation. For each frame where the target is declared tracked, we compute the image-plane pixel offset Δp between the tracked bounding box center and the ground-truth center, and convert Δp to angular error using the camera intrinsic/ focal length f (angular error per pixel $= \alpha = 0.0716°/px$ as used in the simulation). The instantaneous LOS angular error is therefore $\theta = |\Delta p| \times \alpha$ (radians). For each trial, we keep the maximum angular error and the mean angular error over time; statistics reported in table 3 are computed across trials.

Re-detection latency. When an occlusion event causes a lost track, we record the number of frames until the detector re-localizes the target (re-tracking latency). The distribution of re-detection times across trials is reported.

The results have been achieved: The average LOS error angle was obtained as 0.0059 rad (~0.337°); re-tracking time after target loss was < 0.3s; 90% of runs had no false state transitions.

Table 3. *Monte-Carlo simulation parameters used in the evaluation.*

Parameters	Values
Sampling cycle (50 Hz)	$T_s = 0.02\ s$
Equivalent focal length (0.0716°/px)	$f = 800\ px$
PD control parameters	$K_p = 8,\ K_d = 0.04$
Hysteresis latency threshold	$T_{low} = 0.45,\ T_{high} = 0.75$
Simulated pixel offset	Noise $\sigma < 5$ px
Monte-Carlo Statistics	Number of runs = 500

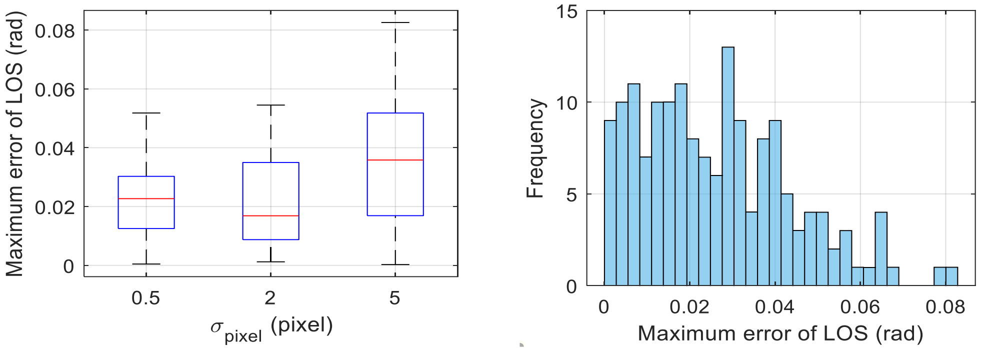


Figure 4. *Monte-Carlo simulation results.*

5.3. Integrated simulation of UAV LOS stabilization

Simulations are performed with the LOS stabilization system for UAV with parameters: $m =$ 1.3 kg, $J = \text{diag}(0.02, 0.02, 0.04)$ kg.m^2; $K_p = 12.0$, $K_d = 0.01$; control cycle: $T_s = 0.01\ s$; target data coming from the image tracker has an average delay of 40 ms.

The LOS reference angle is determined according to (27). The PD controller with $K_p = 8, K_d$ = 0,01 ensures stability. The average LOS deviation: 0.006 rad (~0.337°).

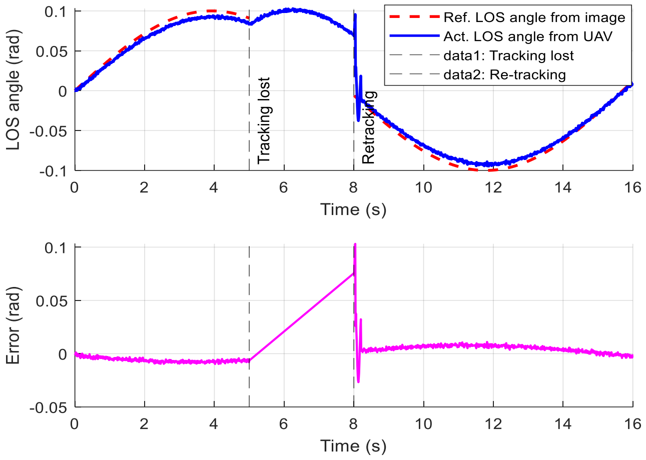


Figure 5. *Integration with LOS stabilization for UAV gimbal.*

6. CONCLUSIONS

The paper presents a real-time long-term target tracking algorithm framework, combining the optimal YOLOv10s detector and NEON-accelerated fDSST tracking on an ARM platform with NPU. Theoretical and experimental analysis confirm the stability, accuracy, and high performance of the system. The system achieves stable tracking, resists temporary tracking loss, and maintains a small LOS error, meeting the requirements of UAV reconnaissance and optical sensor stabilization applications. The integration of the image tracking algorithm on the UAV body camera shows that this method eliminates the mechanical gimbal block, reducing mass and energy consumption. The stabilization process is based on an intelligent UAV attitude controller, which can be combined with an IMU and camera observer to increase accuracy. With the full model (6 degrees of freedom), it can be extended to the Preview or Zero-Sum Game controller and ADP to compensate for delay, wind, and body oscillation. This is a suitable direction for small reconnaissance UAVs or dual-rotor UAVs where the space for installing a gimbal is limited.

While this work integrates detector optimization, NEON acceleration for fDSST, the adaptive multi-feature confidence mechanism, and LOS stabilization simulations, we note that some components (e.g., full 6-DOF flight tests, extended environmental testing) are outside the scope of the present manuscript due to platform/time constraints. The primary contribution of this paper is the integrated algorithmic framework and its demonstration on an ARM + NPU embedded platform. Future work will target extended flight tests, more diverse environmental datasets, and the incorporation of re-identification modules for improved long-term identity preservation. Besides, we will also focus on improving the control quality of this system.

REFERENCES

[1]. Bolme, D. S., Beveridge, J. R., Draper, B. A., & Lui, Y. M., *"Visual Object Tracking Using Adaptive Correlation Filters"*, Proceedings of the 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2010), San Francisco, CA, USA, pp. 2544-2550, (2010).

[2]. Henriques, J. F., Caseiro, R., Martins, P., & Batista, J., *"High-Speed Tracking with Kernelized Correlation Filters"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 3, pp. 583-596, (2015).

[3]. Danelljan, M., Häger, G., Khan, F. S., & Felsberg, M., *"Accurate Scale Estimation for Robust Visual Tracking"*, Proceedings of the British Machine Vision Conference (BMVC 2014), BMVA Press, (2014).

[4]. Bertinetto, L., Valmadre, J., Henriques, J. F., Vedaldi, A., & Torr, P. H. S., *"Fully-Convolutional Siamese Networks for Object Tracking"*, Proceedings of the 2016 European Conference on Computer Vision (ECCV 2016), Amsterdam, Netherlands, pp. 850-865, (2016).

[5]. Li, B., Wu, W., Wang, Q., Zhang, F., Xing, J., & Yan, J., *"SiamRPN++: Evolution of Siamese Visual Tracking with Very Deep Networks"*, Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2019), Long Beach, CA, USA, pp. 4282-4291, (2019).

[6]. Chen, X., Wang, D., Cheng, M.-M., Zhang, W., & Hu, X., *"Transformer Tracking"*, Proceedings of the 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2021), Nashville, TN, USA, pp. 2856-2866, (2021).

[7]. Kalal, Z., Mikolajczyk, K., and Matas, J., *"Tracking-Learning-Detection"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 7, pp. 1409-1422, (2012).

[8]. Ma, C., Yang, X., Zhang, C., & Yang, M.-H., *"Long-Term Correlation Tracking"*, Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015), Boston, MA, USA, pp. 5388-5396, (2015).

[9]. Zhu, Z., Wang, Q., Li, B., Wu, W., Yan, J., & Hu, W., *"Distractor-Aware Siamese Networks for Visual Object Tracking"*, Proceedings of the 15th European Conference on Computer Vision (ECCV 2018), Part IX, Springer, Cham, Switzerland, pp. 103-119, (2018).

[10]. Danelljan, M., Bhat, G., Khan, F. S., & Felsberg, M., *"ECO: Efficient Convolution Operators for Tracking"*, Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017), Honolulu, HI, USA, pp. 6931-6939, (2017).

[11]. Danelljan, M., Häger, G., Khan, F. S., & Felsberg, M., *"Discriminative Scale Space Tracking"*, IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 8, pp. 1561-1575, (2016).

[12]. Wang, A., Chen, H., Liu, L., Chen, K., Lin, Z., Han, J., & Ding, G., *"YOLOv10: Real-Time End-to-End Object Detection"*, arXiv preprint arXiv:2405.14458, (2024).

[13]. Khalil, H.K., *"Nonlinear Systems"*, Prentice Hall, 3rd ed., (2002).

[14]. Duchi, J., Hazan, E., and Singer, Y., *"Adaptive Subgradient Methods for Online Learning and Stochastic Optimization"*, Journal of Machine Learning Research, vol. 12, pp. 2121–2159, (2011).

[15]. Kushner, H.J., and Yin, G.G., *"Stochastic Approximation and Recursive Algorithms and Applications"*, Springer-Verlag, 2nd ed., (2003).

[16]. Sontag, E.D., *"Input to State Stability: Basic Concepts and Results"*, Nonlinear and Optimal Control Theory, Lecture Notes in Mathematics, vol. 1932, Springer, pp. 163–220, (2008).

[17]. Luo, C., et al., *"Real-time visual target tracking based on correlation filters and adaptive confidence fusion"*, IEEE Transactions on Industrial Electronics, vol. 69, no. 8, pp. 8423–8434, (2022).

[18]. Hoeffding, W., *"Probability Inequalities for Sums of Bounded Random Variables"*, The Collected Works of Wassily Hoeffding, Springer Series in Statistics, Springer, New York, NY, (1994). https://doi.org/10.1007/978-1-4612-0865-5_26

TÓM TẮT

Thuật toán bám mục tiêu dài hạn thời gian thực trên nền tảng ARM có gia tốc NPU và ứng dụng trong ổn định đường ngắm UAV

Bài báo này trình bày một thuật toán bám mục tiêu dài hạn hoạt động thời gian thực, được tối ưu cho các nền tảng nhúng ARM có tích hợp gia tốc NPU. Hệ thống kết hợp bộ phát hiện YOLOv10s đã được cắt tỉa và lượng tử hóa với bộ bám fDSST được tối ưu hóa bằng tập lệnh NEON. Hai khối này được liên kết thông qua chỉ số độ tin cậy thích nghi dựa trên hợp nhất nhiều đặc trưng và cơ chế ngưỡng trễ (hysteresis) nhằm kích hoạt bộ phát hiện chỉ khi cần thiết. Phân tích lý thuyết chứng minh tính bị chặn của bộ lọc tương quan, ổn định của quá trình cập nhật trọng số thích nghi, và giới hạn xác suất sai chuyển trạng thái theo hàm mũ. Kết quả thực nghiệm trên nền tảng Orange Pi 5 Max cho thấy, hệ thống đạt tốc độ trung bình 19 FPS cho phát hiện và hơn 100 FPS cho bám, đồng thời duy trì độ ổn định khi có trễ, nhiễu và che khuất tạm thời. Mô phỏng Monte-Carlo và mô phỏng ổn định đường ngắm (LOS) trên bệ quay UAV xác nhận sai số góc cực đại trung bình khoảng 0,006 rad và khả năng tái bám nhanh sau khi mất mục tiêu. Thuật toán có tiềm năng ứng dụng trong các hệ thống ổn định đường ngắm, giám sát và trinh sát quang học thời gian thực.

Từ khoá: Bám mục tiêu dài hạn; YOLOv10s; fDSST; NPU; ARM; Ổn định đường ngắm UAV.