# Robust and lightweight UAV visual localization in GNSS-denied environments using a variational autoencoder

Phan Huy Anh*, Ngo Van Quan, Bui Thi Thanh Tam, Cao Van Toan

Institute of Information Technology and Electronics, Academy of Military Science and Technology, 17 Hoang Sam, Nghia Do, Hanoi, Vietnam.
*Corresponding author: huyanhfanvdt@gmail.com
*Received 02 Mar. 2026; Revised 27 Apr. 2026; Accepted 15 Jun. 2026; Published 25 Jun. 2026.*
**DOI:** https://doi.org/10.54939/1859-1043.j.mst.112.2026.56-63

## ABSTRACT

*This paper proposes a robust and lightweight visual localization framework for UAVs in GNSS-denied environments. Utilizing a Variational Autoencoder (VAE) trained on full RGB imagery, the system extracts rich features compressed into an optimized 256-dimensional latent space to accommodate onboard constraints. These features are matched using an unnormalized $L_2$ Euclidean distance, while a Linear Kalman Filter (LKF) smooths the trajectory. Experiments demonstrate this model outperforms baselines, achieving a raw RMSE of 0.087 m, which improves to 0.065 m with the LKF. This approach ensures stable, highly accurate real-time navigation.*

**Keywords:** UAV; Visual localization; Variational autoencoder.

## 1. INTRODUCTION

Unmanned Aerial Vehicles (UAVs) have become indispensable for various operations, including search and rescue, environmental monitoring, and infrastructure inspection. However, their reliance on Global Navigation Satellite Systems (GNSS) poses significant risks in environments where signals are degraded or denied, such as urban canyons and indoor spaces. To mitigate these hazards, visual-based map-matching techniques have emerged as a reliable alternative, enabling robust pose estimation by comparing real-time onboard camera images against pre-built maps [1].

For resource-constrained UAV platforms, visual localization pipelines must strictly balance speed, accuracy, and storage efficiency. Traditional feature-based methods struggle to meet these requirements. For instance, Scale-Invariant Feature Transform (SIFT) provides high matching accuracy but incurs prohibitive computational costs, rendering it impractical for real-time edge processing. Conversely, lightweight alternatives like Oriented FAST and Rotated BRIEF (ORB) offer significant speedups but lack robustness under dynamic illumination or severe viewpoint changes [2].

Consequently, deep learning approaches, such as Autoencoders (AEs) and NetVLAD, have been widely adopted to extract robust semantic descriptors [3, 4]. For instance, Bianchi and Barfoot [5] demonstrated that encoding pre-collected satellite maps into low-dimensional latent spaces enables rapid inner-product kernel matching, drastically reducing inference time compared to traditional mutual information techniques. To further address rotational misalignment and matching inaccuracies, recent hybrid architectures combine AEs for coarse database retrieval with SIFT (AE-SIFT) for fine-grained homography estimation [6]. While these AE-SIFT pipelines enhance localization accuracy under misaligned conditions, their reliance on traditional feature matching in the final stage reintroduces computational bottlenecks, hindering real-time UAV operations. Furthermore, standard AE models still produce relatively high-dimensional latent vectors (e.g., 1000 dimensions or more), imposing considerable memory footprints unsuitable for embedded hardware. Moreover, modern visual place recognition (VPR) systems and standard AEs predominantly employ normalized metrics, such as cosine similarity or inner-product kernels, for

feature matching [5]. Recent studies demonstrate that applying such metrics in unconstrained embedding spaces discards crucial vector magnitudes, yielding arbitrary similarities and suboptimal matching results in visually noisy domains [7]. While Variational Autoencoders (VAEs) map inputs into continuous Gaussian distributions - better handling uncertainty than standard discrete AEs - existing implementations often discard magnitude information during matching, failing to leverage the full probabilistic benefits of their latent spaces.

To address these shortcomings, this paper proposes a novel, lightweight and robust visual localization architecture tailored for GNSS-denied UAVs. Our system integrates a lightweight VAE with an unnormalized $L_2$-metric matching strategy. The main contributions include:

- A decoupled VAE architecture achieving deep compression into a 256-dimensional latent representation, significantly reducing the memory footprint while preserving fidelity.
- Theoretical and empirical validation for utilizing unnormalized Euclidean distance ( $L_2$ ) to capture the true probabilistic nature of visual features.

The remainder of this paper is organized as follows: Section 2 details the proposed methodology, Section 3 presents experimental evaluations, and Section 4 concludes the paper.

## 2. METHODOLOGY

The proposed Unmanned Aerial Vehicle (UAV) visual localization framework operates through a two-phase pipeline: an offline phase and an online phase (Figure 1). During the offline phase, a reference satellite map is cropped into $256 \times 256$ RGB images to train a Variational Autoencoder (VAE) on a ground station. Once trained, the encoder parameters and the reference feature database are deployed onboard the UAV's companion computer.

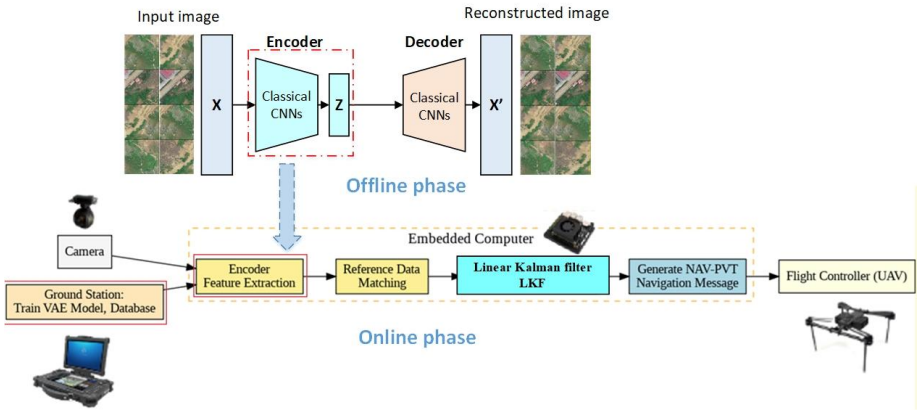


***Figure 1.*** *VAE based UAV localization system.*

During the online execution phase, real-time camera frames are processed to extract and encode features within the Encoder Feature Extraction block. These features are subsequently matched against the reference database within the Reference Data Matching block to estimate the UAV's global position. The estimated position is then smoothed by the Linear Kalman Filter block and fed into the Generate NAV-PVT Navigation Message block. This final module synthesizes standard positioning telemetry - mimicking the output format of a physical GNSS receiver - which is then utilized to directly command the UAV's flight controller.

Unlike traditional feature-based matching algorithms or previous autoencoder applications that rely heavily on grayscale imagery [5] - which is a destructive transformation that collapses different materials with similar luminance into identical intensity values - this methodology leverages the full 3-channel RGB tensor. Preserving this RGB data allows the network to learn disentangled chromatic and morphological features, thereby preventing local feature collapse in

visually ambiguous terrains. By capturing these richer spatial, structural and textural contexts, this integration significantly enhances the overall accuracy, stability and robustness of the localization process under varying environmental conditions.

At the core of the feature extraction pipeline is a Variational Autoencoder (VAE), as depicted in Figure 1. The encoder compresses high-resolution RGB satellite reference images into a low-dimensional latent space. The network consists of a sequence of 2D convolutional layers, each followed by Batch Normalization and LeakyReLU activation functions. As the spatial dimensions are progressively downsampled, the number of feature channels is doubled (from 32 up to 1024) to capture complex, multi-scale hierarchical features. The utilization of RGB images over grayscale is a critical enhancement. Grayscale conversion discards significant chromatic information that helps distinguish terrain types (e.g., grass versus asphalt) and subtle structural boundaries, especially under varying illumination. Preserving the 3-channel RGB input forces the autoencoder to learn more distinct and resilient feature representations, thereby mitigating localization failures caused by shadows or seasonal changes. Finally, compressing the latent space to 256 dimensions effectively captures the intrinsic data dimension of the aerial imagery. This bottleneck acts as a semantic low-pass filter that discards high-frequency noise while mitigating the curse of dimensionality and preventing posterior collapse commonly observed in overly wide latent spaces.

**) Training variational autoencoder (VAE) with RGB images*

Unlike the baseline method [5] that relies on single-channel Grayscale imagery, our VAE is explicitly trained utilizing full 3-channel RGB inputs. The trained Encoder parameters and the reference database are subsequently uploaded to the onboard embedded computer. The autoencoder is trained offline using a custom loss function designed to balance precise image reconstruction with a well-regularized latent space. The total loss function, $L_{total}$, is defined as a weighted sum of the Mean Squared Error (MSE) and the Kullback-Leibler Divergence (KLD):

$$L_{total} = L_{MSE} + \beta \cdot L_{KLD} \tag{1}$$

Where the scaling hyperparameter $\beta$ is set to 0.00025 to balance regularization and visual reconstruction; $L_{MSE}$ measures the pixel-wise reconstruction fidelity between the original RGB input $x$ and the decoded output $\hat{x}$; $L_{KLD}$ represents the Kullback-Leibler Divergence regularizing the latent distribution toward a standard normal distribution $\mathcal{N}(0, I)$:

$$L_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{x}_i)^2 \tag{2}$$

$$L_{KLD} = -\frac{1}{2}\sum_{j=1}^{J}\left(1 + \log(\sigma_j^2) - \mu_j^2 - \sigma_j^2\right) \tag{3}$$

where $N$ represents the total number of elements in the input tensor; $J$ is the latent dimension size, and $\mu_j$ and $\sigma_j^2$ are the mean and variance of the $j$-th latent variable predicted by the encoder.

During the online localization phase, to guarantee absolute deterministic stability, the stochastic reparameterization trick is omitted, and the system exclusively extracts the deterministic mean vector $\mu \in \mathbb{R}^J$ as the unique image descriptor.

**) Reference Data Matching and outlier filtering*

During flight, the real-time RGB frame is encoded into $\mu_{query}$ and compared against a pre-computed reference database $\mu_{ref,i}$. Since the magnitude of a Gaussian-regularized latent vector

reflects its probability density, standard normalized metrics like cosine similarity can yield suboptimal matching in noisy domains [7]. Therefore, the system evaluates similarity using the unnormalized Euclidean distance ( $L_2$ ):

$$d_i = \| \mu_{query} - \mu_{ref,i} \|_2 \tag{4}$$

where $d_i$ is the unnormalized Euclidean distance in the latent space between the real-time query feature vector $\mu_{query}$ and the $i$ -th reference feature vector $\mu_{ref,i}$ from the database.

To convert this distance into a similarity score $s_i$ (where higher is better), an inverse formulation is applied with a small constant $\epsilon = 10^{-9}$ to prevent division by zero:

$$s_i = \frac{1}{d_i + \epsilon} \tag{5}$$

Relying on a single best match is highly susceptible to grid-spacing errors and outliers. Instead, the algorithm retrieves the top $K$ reference images with the smallest distances. To filter out weak candidate matches (outliers), the system computes the standard deviation of the similarities, $\sigma_s$ , and zeros out any similarity score that falls significantly below the maximum similarity:

$$s_i = \begin{cases} s_i, & \text{if } s_i \geq \max(s) - \sigma_s \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where the standard deviation $\sigma_s$ is computed as the square root of the mean of the quared differences between each individual similarity score $s_i$ and the average similarity score $\bar{s}$ across the top $K$ retrieved matches:

$$\sigma_s = \sqrt{\frac{1}{K} \sum_{i=1}^{K} (s_i - \bar{s})^2} \tag{7}$$

The remaining valid similarities are normalized to sum to 1, creating weights $w_i$ . The final estimated coordinates are computed as the weighted average of the reference positions [5]:

$$Lat_{est} = \sum_i w_i \cdot Lat_{ref,i} \tag{8}$$

$$Lon_{est} = \sum_i w_i \cdot Lon_{ref,i} \tag{9}$$

**) Trajectory smoothing via a Linear Kalman filter (LKF)*

To facilitate the linear kinematic modeling in the subsequent Kalman Filter phase, these spherical coordinates are immediately projected into the Universal Transverse Mercator (UTM) planar coordinate system (Easting and Northing, measured in meters). Because visual localization can occasionally yield noisy discrete position jumps, a Linear Kalman filter (LKF) is coupled with the vision pipeline to smooth the trajectory and estimate vehicle dynamics.The system utilizes a Constant Velocity (CV) model. The state vector $X_k$ is defined as the 2D position and velocity:

$$X_k = \left[ x_k, y_k, v_{x,k}, v_{y,k} \right]^T \tag{10}$$

The state is projected forward based on the time delta $\Delta t$ between frames:

$$X_{k|k-1} = F \cdot X_{k-1|k-1} \tag{11}$$

$$P_{k|k-1} = F P_{k-1|k-1} F^T + Q \tag{12}$$

where $X_{k|k-1}$ is the predicted next state based on the transition matrix $F$; $P_{k|k-1}$ is the predicted system error covariance; $Q$ represents the process noise covariance matrix; $F$ is the state transition matrix:

$$F = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13}$$

When a valid localization coordinate $[x_{map}, y_{map}]^T$ (projected from the estimated $Lat_{est}, Lon_{est}$) is successfully generated by the reference matching block, it acts as the measurement $Z_k = [x_{map}, y_{map}]^T$. The observation model maps the state $X_k$ to the measurement space using matrix $H$:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{14}$$

The Kalman gain ($G$) and updated state are computed as:

$$G = P_{k|k-1} H^T S^{-1} \tag{16}$$

where, $S = HP_{k|k-1}H^T + R$ with $R$ is the measurement noise covariance.

To further stabilize the UAV's heading and velocity output, the raw LKF velocity components $(v_x, v_y)$ are passed through a moving average window. The final smoothed heading is computed dynamically from these smoothed velocities $\left(\theta = \arctan 2(v_{x,smooth}, v_{y,smooth})\right)$, ensuring the navigation outputs remain highly robust against transient visual noise.

Following the LKF processing cycle, the smoothed planar positions $(x_{k|k}, y_{k|k})$ extracted from the updated state vector $X_{k|k}$ are inversely transformed back into geographic spherical coordinates $(Lat_{filtered}, Lon_{filtered})$. These filtered coordinates, alongside the moving-average smoothed velocities and derived heading, are subsequently fed into the Generate NAV-PVT Navigation Message block. This module formats the data strictly into standard GNSS telemetry protocols (e.g., UBX NAV-PVT). By continuously transmitting these serialized messages via a serial interface (UART), the embedded companion computer effectively spoofs a high-precision physical GNSS receiver. Consequently, the UAV's flight controller seamlessly ingests this visual-based telemetry, enabling continuous, autonomous waypoint navigation in GNSS-denied environments without requiring any core firmware modifications.

## 3. RESULTS AND DISCUSSION

To thoroughly evaluate the real-world applicability of the proposed RGB-based Variational Autoencoder (RGB VAE) framework, comprehensive experiments were conducted using a meticulously documented, custom-collected dataset. During the data acquisition phase, the UAV maintained an average flight altitude of 100 meters Above Ground Level (AGL) with a downward-facing camera field of view of 90 degrees, covering a flight trajectory of approximately 300 meters. The generated reference map was cropped into overlapping 256×256-pixel images, yielding 3,069 training samples. The VAE model was trained using the Adam optimizer for 150 epochs with a batch size of 64. Finally, to validate real-time capability, the complete online localization pipeline

was deployed on an Nvidia Jetson Orin NX embedded computer, achieving stable inference speeds comparable to the baseline [5].

Crucially, to assess the framework's generalization capabilities, the testing dataset was acquired at a later temporal stage during which the terrain had undergone significant environmental changes. Specifically, previously vegetated areas had been converted into bare land, resulting in the loss of numerous structural features. This temporal gap rigorously challenged the model's robustness against substantial topological variations. Finally, to accurately evaluate the metric localization quality and calculate the cross-track Root Mean Square Error (RMSE), all ground truth and estimated coordinates were projected into the UTM coordinate system. This spatial conversion explicitly accounts for the deviations in meters presented in the trajectory comparison.

To rigorously test the robustness and compressibility of the feature representations, we evaluated our proposed framework against the baseline autoencoder architecture established by Bianchi and Barfoot, denoted as Grayscale AE [5], across various latent dimensions. The 256-dimensional model is referred to as the optimized model. The evaluation specifically focuses on the raw cross-track error to independently assess the pure vision-based matching accuracy prior to any Linear Kalman Filter (LKF) smoothing (denoted as RGB VAE). Furthermore, the complete pipeline incorporating the smoothing filter is designated as RGB VAE + LKF. Table 1 summarizes the statistical analysis of the cross-track localization errors across the different configurations.

***Table 1.*** *Statistical analysis of cross-track errors across VAE configurations.*

| Models | Latent dimension | Mean error (m) | RMSE (m) | Max error (m) | Median error (m) |
|---|---|---|---|---|---|
| Grayscale AE [5] | 1000 | 0.582 | 0.637 | 0.827 | 0.664 |
| RGB VAE | 1000 | 0.109 | 0.141 | 0.827 | 0.083 |
| RGB VAE + LKF | 1000 | 0.108 | 0.121 | 0.231 | 0.102 |
| Grayscale AE [5] | 256 | 0.446 | 0.518 | 0.827 | 0.494 |
| RGB VAE | 256 | 0.064 | 0.087 | 0.386 | 0.077 |
| RGB VAE + LKF | 256 | 0.052 | 0.065 | 0.245 | 0.046 |
| Grayscale AE [5] | 128 | 0.597 | 0.647 | 0.827 | 0.681 |
| RGB VAE | 128 | 0.209 | 0.292 | 0.827 | 0.137 |
| RGB VAE + LKF | 128 | 0.199 | 0.281 | 0.824 | 0.128 |

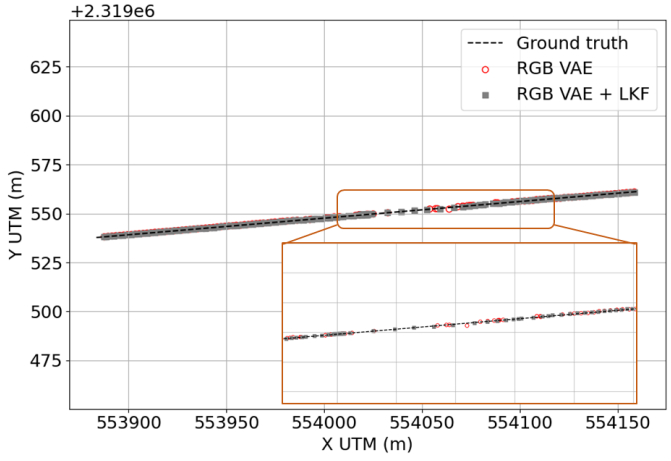


***Figure 2.*** *Trajectory comparison between ground truth, RGB VAE, and RGB VAE + LKF.*

Table 1 clearly highlights the performance advantage of the proposed framework over the baseline approach. Notably, when compressed to 256 dimensions, the grayscale AE model's RMSE degrades to 0.518 m. In contrast, the Proposed RGB VAE model maintains highly robust features,

yielding a significantly lower Mean Error (0.064 m) and RMSE (0.087 m). Furthermore, when applying the kinematic smoothing technique, the proposed RGB VAE + LKF architecture achieves the best overall performance with a Mean Error of 0.052 m and an RMSE of 0.065 m, effectively demonstrating the superiority of our holistic design.

The trajectory plot visually confirms the effectiveness of the LKF implementation, as described in Figure 2. The filtered data closely hugs the ground truth line, successfully smoothing out the minor deviations present in the raw visual matching data, while concurrently providing continuous velocity vector estimates. This dynamic tracking allows the Flight Controller to seamlessly interpolate coordinates between dropped frames or during temporary camera occlusions.

Table 2 details the technical metrics regarding storage capacity and training efficiency across the evaluated models. The results indicate that while model and database sizes scale proportionally with the latent dimension, the input modality (RGB versus Grayscale) introduces no significant difference in storage footprint. Notably, despite the higher complexity of processing 3-channel data, the training time for the proposed RGB VAE remains comparable to the baseline model. This demonstrates that transitioning to the RGB domain significantly enhances localization accuracy without imposing additional computational burdens or memory overhead on the embedded hardware.

***Table 2.*** *Model storage and training metrics.*

| Models | Latent dimension | Model size (MB) | Database size (MB) | Training time (minutes) |
|---|---|---|---|---|
| Grayscale AE [5] | 1000 | 149 | 11.8 | 36.40 |
| RGB VAE | 1000 | 149 | 11.8 | 37.10 |
| Grayscale AE [5] | 256 | 56 | 3.17 | 35.67 |
| RGB VAE | 256 | 56 | 3.17 | 37.03 |
| Grayscale AE [5] | 128 | 40 | 1.67 | 35.50 |
| RGB VAE | 128 | 40 | 1.67 | 36.84 |

Tables 1 and 2 justify the 256-dimensional latent vector as the optimal configuration, striking an ideal balance between localization accuracy and storage efficiency. Extreme compression to 128 dimensions minimizes memory but severely degrades accuracy (RMSE 0.281 m). Conversely, the 1000-dimensional baseline yields negligible precision gains while nearly tripling the storage requirement (149 MB), making it inefficient for edge deployment. Ultimately, the 256-dimensional architecture reduces the onboard database size by 73% (to 3.17 MB) compared to the 1000-dimensional model, ensuring stable, resource-efficient real-time navigation without compromising accuracy.

## 4. CONCLUSIONS AND FUTURE WORKS

This paper presented a robust vision-based UAV localization framework utilizing an RGB-based VAE for GNSS-denied environments. By processing full-color imagery, the model extracts richer features, significantly improving image matching reliability. The framework demonstrates that the 256-dimensional latent space provides an optimal trade-off, enabling a 73% reduction in database size without degrading localization precision, achieving a superior raw cross-track RMSE of 0.087 m. By executing intensive training offline, the UAV only loads a highly compressed database, ensuring that online inference and $L_2$ distance matching are fast and memory-efficient. Future research will focus on tightly coupling the Kalman filter architecture with Inertial Measurement Unit (IMU) data to ensure fail-safe autonomous navigation during temporary camera occlusions, and evaluating the framework across diverse seasonal and lighting conditions.

## REFERENCES

[1]. E. P. Herrera-Granda, J. C. Torres-Cantero, A. Rosales, and D. H. Peluffo-Ordóñez. *"A Comparison of Monocular Visual SLAM and Visual Odometry Methods Applied to 3D Reconstruction"*. Appl. Sci., vol. 13, no. 15, p. 8837, (2023).

[2]. X. Yu, C. Wang, X. Li, and J. Zhang. *"A Robust Learned Feature-Based Visual Odometry"*. IEEE Trans. Instrum. Meas., vol. 72, pp. 1-12, (2023).

[3]. I. Moskalenko, A. Kornilova, and G. Ferrer. *"Visual place recognition for aerial imagery: A survey"*. Rob. Auton. Syst., vol. 183, p. 104837, (2024).

[4]. Y. Wang et al. *"Multi-Modal Aerial-Ground Cross-View Place Recognition with Neural ODEs"*. Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR), (2025).

[5]. M. Bianchi and T. D. Barfoot. *"UAV localization using autoencoded satellite images"*. IEEE Robot. Autom. Lett., vol. 6, no. 2, pp. 1761-1768, (2021).

[6]. N. V. Quan, P. H. Anh, B. T. T. Tam, and N. C. Thanh. *"Efficient UAV localization using combined autoencoder and SIFT"*. Tạp chí Nghiên cứu KH&CN quân sự, (2024) (in Vietnamese).

[7]. H. Steck, C. Ekanadham, and N. Kallus. *"Is Cosine-Similarity of Embeddings Really About Similarity?"*. Companion Proceedings of the ACM Web Conference 2024, pp. 887-890, (2024).

## TÓM TẮT

### Phương pháp định vị bằng hình ảnh bền vững và nhẹ cho UAV sử dụng Autoencoder biến phân trong môi trường thiếu tín hiệu định vị vệ tinh toàn cầu

*Bài báo này đề xuất một phương pháp định vị bằng hình ảnh bền vững và nhẹ cho UAV trong môi trường thiếu tín hiệu định vị vệ tinh toàn cầu. Việc sử dụng mạng Autoencoder biến phân (VAE) được huấn luyện trên tập ảnh RGB đầy đủ, hệ thống trích xuất nhiều đặc trưng và nén vào một không gian ẩn 256 chiều tối ưu nhằm đáp ứng các hạn chế về phần cứng trên khoang. Các đặc trưng này được so khớp thông qua khoảng cách Euclidean $L_2$ không chuẩn hóa, đồng thời một bộ lọc Kalman tuyến tính (LKF) được sử dụng để làm mượt quỹ đạo bay. Thực nghiệm chứng minh mô hình này vượt trội so với các cấu hình cơ sở, đạt sai số bình phương trung bình (RMSE) quỹ đạo đối với dữ liệu chưa lọc là 0,087 m và đạt 0,065 m sau khi qua bộ lọc LKF. Phương pháp đề xuất này tối ưu bộ nhớ đảm bảo khả năng định vị dẫn đường theo thời gian thực ổn định và đạt độ chính xác cao.*

**Từ khoá:** UAV; Định vị bằng hình ảnh; Autoencoder biến phân.