Automation of UAV Navigation Support Based on SIFT-like Methods*

Pylyp Prystavka[1,†], Olha Cholyshkina[2,*,†], Oksana Kovtun[2,†], Dmytro Pryshchepa[1,†]

[1] *The State University "Kyiv Aviation Institute", 1 Lubomyr Huzar Avenue, 03058 Kyiv, Ukraine*

[2] *Taras Shevchenko National University of Kyiv, 60 Volodymyrska Street, 01033 Kyiv, Ukraine*

Abstract

This paper presents an approach to supporting the navigation of an unmanned aerial vehicle (UAV) via an optical channel in cases where satellite navigation signals are unavailable. The proposed algorithmic technology is based on pre-planned flight routes and a reference dataset of landmark images. Landmarks are selected according to their visual characteristics, ensuring high recognition reliability. Image processing employs algorithms for detecting and describing local features, including SIFT, ORB, and neural network architectures such as SuperPoint and LightGlue. The proposed approach was tested under simulated flight conditions using hardware platforms Raspberry Pi 4 Model B and Orange Pi 5 Pro. The results confirm the effectiveness of the proposed method for automating UAV navigation support through optical channels, enabling autonomous UAV localization in GPS-denied environments.

Keywords

UAV, optical navigation, SIFT, ORB, image descriptors, landmarks, density distribution [1]

1. Introduction

The increasing role of unmanned aerial vehicles (UAVs) in both civil and military applications—such as surveillance, reconnaissance, monitoring, and data collection—requires continuous improvement in navigation and positioning technologies. With the advancement of onboard hardware and software, determining the UAV's spatial position becomes critical for effective flight management and mission success.

Accurate localization is essential for tactical operations, autonomous navigation, collision avoidance, and real-time mission execution, particularly in complex terrains or combat scenarios.

Optical channels—cameras [1], video sensors, and other visual systems—are of particular importance as the primary source of information for determining UAV coordinates in environments with limited or unstable GPS access, including urban areas and zones of electronic interference.

Localization methods based on image analysis—particularly those using algorithms for detecting and matching key features, such as SIFT [2]—can achieve high positioning accuracy even under varying scale, illumination, and perspective. These approaches are indispensable for aerial photography [3], terrain monitoring, search and rescue operations, environmental assessment, and precision agriculture.

In military contexts, SIFT-based techniques facilitate reliable object recognition, terrain orientation, and navigation automation during reconnaissance and strike missions, especially under adversarial countermeasures.

*International Workshop on Computational Intelligence, co-located with the IV International Scientific Symposium "Intelligent Solutions" (IntSol-2025), May 01-05, 2025, Kyiv-Uzhhorod, Ukraine

[1*] Corresponding author.

[†] These authors contributed equally.

olha.cholyshkina@knu.ua (O.Cholyshkina); chindakor37@gmail.com (P.Prystavka); Oksana.kovtun@knu.ua (O.Kovtun); dmytroo.bc@gmail.com (D. Pryshchepa).

0000-0002-0360-2459 (P.Prystavka); 0000-0002-0681-0413 (O.Cholyshkina); 0000-0003-0871-5097 (O.Kovtun); 0009-0008-3581-8474 (D. Pryshchepa)

Research in this domain enables the development and enhancement of computer vision algorithms, spatial analysis models, and software tools that improve UAV autonomy and operational capabilities.

Therefore, determining UAV position using optical channels and keypoint detection methods—especially SIFT—holds substantial scientific and practical value across a broad range of applications, from civilian to military domains, and remains a priority area in modern autonomous systems research.

2. Review of existing solutions and literature sources

Modern UAV navigation systems are the subject of active research in both academia and industry. Efforts are underway worldwide to enhance the reliability and accuracy of UAV positioning, especially in environments where GPS usage is limited or impossible [4]. Particular attention is paid to developing alternative navigation solutions for scenarios with obstacles or complete satellite signal loss.

Study [5] presents a multi-level localization module combining GNSS, inertial navigation, and visual depth data. This approach improves the robustness and accuracy of navigation in complex conditions. Similarly, [6] proposes a new GNSS/INS/LiDAR integration scheme that ensures continuous and precise navigation where satellite access is restricted.

An alternative strategy involves the development of low-cost navigation solutions for GPS-denied environments [7], based on new-generation sensors and visual scene analysis methods. Such systems increasingly employ computer vision techniques, including keypoint detection and description algorithms, with SIFT, SURF, and ORB being among the most prominent.

Before the rise of deep learning classifiers, object detection and matching on images relied on rotation- and scale-invariant methods. One of the most recognized was the patented SIFT (Scale-Invariant Feature Transform) algorithm, introduced by David G. Lowe in 1999. SURF (Speeded-Up Robust Features), developed by Herbert Bay in 2006, built upon SIFT's principles. Both algorithms became foundational for various navigation and recognition applications.

An in-depth analysis of local feature-based methods is provided in the survey by B.H. Kukharenko, "Image Analysis Algorithms for Detecting Local Features and Recognizing Objects and Panoramas", which explores both the advantages and limitations of these approaches. Further advancements were made by researchers like T. Lindeberg, who developed the concept of scale-space, and Brown, Hua, and Winder, who investigated discriminative descriptor training and real-time object tracking using contour levels.

Work [8] explores a multi-layer architecture for autonomous UAVs addressing diverse challenges such as adaptive environmental interaction and decision-making. Optical navigation using SIFT facilitates landmark recognition and enhances positioning accuracy in visual odometry tasks.

To mitigate GPS loss, researchers recommend employing additional optical sensors integrated via Kalman filters and their modifications [9–10]. SIFT-based methods in such systems enhance image alignment and reduce localization errors during motion.

Several reviews emphasize the role of artificial intelligence. In [11], an LSTM-based model is proposed to improve visual odometry, using SIFT-derived features as input. Work [12] discusses adaptive learning with step-size regulation for enhanced inertial navigation, while [13] introduces an ensemble deep learning method for GPS spoofing detection as part of a broader navigation system.

SIFT-based approaches remain integral to visual localization algorithm development due to their robustness against geometric and lighting variations. These qualities are vital for real-time UAV operation in complex or dynamic environments.

Notable contributions to the field have been made by Ukrainian researchers [19–21] and by the authors of this study [2, 16–18], who explore visual feature analysis and SIFT-based localization algorithms..

3. Objectives of the study and research questions

This study aims to develop and test an information technology system for automated UAV navigation support based on SIFT-like methods. In scenarios where GPS signals are unavailable, the proposed system enables UAV localization through the identification of pre-defined visual landmarks using onboard surveillance cameras.

To address this challenge, we examine several SIFT-like methods along with neural network architectures such as LightGlue and SuperPoint. A key feature of these algorithms is their ability to return only those descriptor sets and coordinates of keypoints that show mutual similarity. Typically, they automatically match descriptors between two images. However, this can result in false positives when descriptors from reference and test images coincidentally align. To mitigate this effect and properly interpret the matching outcomes, we propose a custom method based on evaluating the spatial density distribution of keypoint coordinates [22].

For experimental validation, a custom software application was developed to operate in two modes: object detection and coordinate estimation. The solution was tested on computational platforms that emulate UAV onboard systems using aerial imagery data.

4. Research materials and methods

Feature detection and description algorithms are fundamental for building visual navigation systems that function without GPS. The typical operation of such algorithms involves several key stages. First, local features are detected—these are distinctive elements in the image that remain invariant under changes in lighting, scale, orientation, and geometric transformation. Next, descriptors are generated—numeric representations of image fragments surrounding each keypoint. These descriptors are then matched between image frames or samples to establish spatial correspondences. Finally, erroneous matches are filtered using criteria such as vector distance thresholds or a minimum required number of matches.

In this study, we selected three approaches for performance comparison in GPS-denied UAV navigation scenarios: SIFT (Scale-Invariant Feature Transform), ORB (Oriented FAST and Rotated BRIEF), and neural architectures LightGlue and SuperPoint. The selection was based on prior benchmarking results [9], showing that ORB offers superior speed, while SIFT delivers high-quality matches in most conditions. In experiments, these algorithms were tested on images subjected to various transformations: brightness changes, scaling, rotation, geometric distortions (e.g., fisheye effect), and noise addition.

LightGlue stands out among neural solutions for its compatibility with different types of descriptors, including SuperPoint, DISK, ALIKED, and even classical SIFT. According to recent studies [10], SuperPoint performs well in real-time tasks but is less robust to scale changes compared to ALIKED, particularly the ALIKED (16, MS) variant designed for scale-sensitive applications.

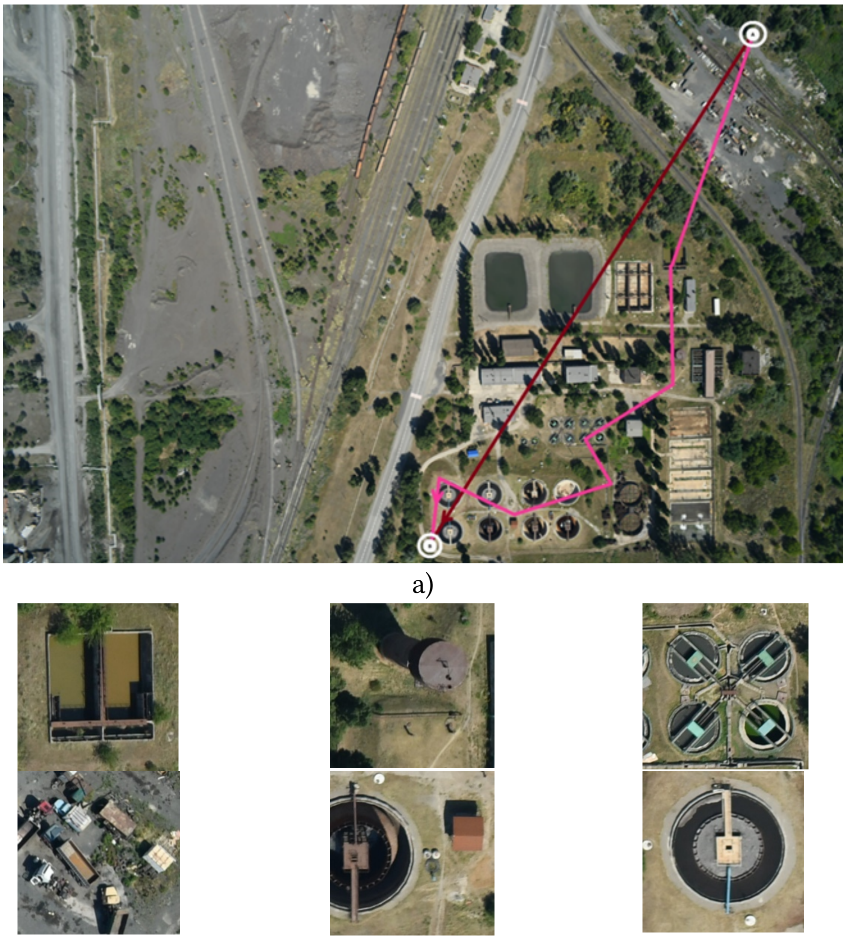
Despite this, SuperPoint was chosen as one of the main algorithms in this work due to its ability to simultaneously detect keypoints and compute descriptors over full-resolution images in real time [12]. Notably, it is self-supervised and does not require large labeled datasets for training.

4.1. Pre-flight preparation

The proposed system includes a pre-flight preparation stage involving route planning and the formation of a reference set of landmark images. Landmarks are considered to be objects with clearly defined visual characteristics—such as shape, contour, or texture—that can be reliably identified in images. These may include buildings with distinctive geometry, open industrial structures, power infrastructure (e.g., substations), hydraulic facilities (e.g., dams), and others.

Flight routes are planned to pass through areas where such landmarks are located. This enhances the accuracy of visual UAV localization, even if the route is longer than a direct alternative. For example, as shown in Fig. 1(a), the pink route “A–B” is longer than the red one but traverses areas with higher visual information density.

Each image in the reference dataset contains a specific landmark captured during the pre-flight phase (Fig. 1(b)). For every image, keypoints are detected and georeferenced to real-world coordinates, which are later used during visual matching and localization in flight.



a)

b)

Figure 1: Flight planning: a) schematic map and planned flight route; b) landmarks identified along the route.

4.3. Formal formulation of the problem

Let us consider a reference dataset of images

$$I=\{i_1,i_2,i_3,\dots,i_k\},k=\overline{1,N} \tag{1}$$

with various characteristic landmark objects, we define a set of local features that are described by their location (keypoint pixel coordinates) and descriptors - unique vectors of fixed length. Each image i_k contains a different number of keypoints and, accordingly, descriptors:

$$d_{k,p_k,}\, p_k=\overline{1,N_k},$$

$$d_{k,p_k,}=\left(\left(d_{k,p_k}\right)_1,\left(d_{k,p_k}\right)_2,\dots,\left(d_{k,p_k}\right)_l\right),l=\overline{1,n}$$

where N – is the number of keypoints for image i_k, n– vector dimention.

The images were aligned to a digital map, meaning that each pixel was assigned real-world geographic coordinates

Therefore, for each descriptor d_{k,p_k}, we obtain coordinates

$$C_{k,p_k}=\{x_{k,p_k},y_{k,p_k}\},k=\overline{1,N},p_k=\overline{1,N_k}, \quad (2)$$

belonging to the set $C_{k,p_k} \epsilon \, Coord$

The following data set is generated (ND):

$$S=\{d_{k,p_k},C_{k,p_k},k=\overline{1,N},p_k=\overline{1,N_k}\}. \quad (3)$$

A tabular representation of this dataset is shown below (Table 1).

Table 1
Data set

Landmark image	Key points descriptors	Coordinates
	d_{11}	C_{11}
	d_{12}	C_{12}
I_1	….	….
	d_{1p1}	C_{1p1}
…	…	…
	d_{N1}	C_{N1}
	d_{N2}	C_{N2}
I_N	d_{N3}	C_{N3}
	…	….
	d_{NPN}	C_{NPN}.

For each input image, the task is to determine whether it contains an object from the reference set *I*. If so, the system must estimate its coordinates.

A key feature of algorithms for detecting, describing, and matching local features is that they return only those descriptors and keypoints that are mutually similar. These algorithms typically establish correspondences between two sets of descriptors. However, random matches between descriptors in reference and test images can lead to false positives.

To address this issue and improve interpretation accuracy, we propose a method based on evaluating the spatial density function of keypoint coordinates [22]

Let the coordinate space be divided Δ_{s_x,s_y} into rectangular regions:

$$x_{k,l},y_{k,l,}l=\overline{1,M},$$

where M – number of identified keypoint matches in the frame, on rectangular areas

$$\Delta_{h_t,h_q}:(x_{min}+ii\cdot s_x,y_{min}+jj\cdot s_y),ii=\overline{0,M_x-1},jj=\overline{0,M_y-1},$$

where M_x, M_y – is the number of rectangular partitions;

$$x_{min}=\min_{l=\overline{1,M}}\{x_{k,l}\};y_{min}=\min_{l=\overline{1,M}}\{y_{k,l}\},$$

which determine the empirical probability of the appearance of a particular point from the array in a specific local partition region Δ_{s_x,s_y}, we will get

$$f_{ii,jj}=\frac{1}{N}\sum_{k=1}^{N} I_k, ii=\overline{0,M_x-1}, jj=\overline{0,M_y-1},$$

where

$$I_k=\begin{cases} 1, if\ ii=\left[\frac{x_l-x_{min}}{s_x}\right] ma\ jj=\left[\frac{y_l-y_{min}}{s_y}\right] \\ 0, otherwise \end{cases}$$

where $[\,]$ - whole part function.

Let Matches be the set containing only those descriptors that match the descriptors of the input image $.I_{flight}$. It is assumed that if the target object is indeed present in the input image, then the keypoints corresponding to the descriptors from the set Matches will lie compactly, that is, within a single histogram partition cell with the maximum frequency value at partition class i^*, j^*:

$$i^*, j^* = \underset{ii,jj}{arg\,max}\{f_{ii,jj}\}.$$

After identifying i^*, j^*, the coordinates of the target landmark can be chosen either as the center of this region or as the arithmetic mean of the coordinates of the descriptors located within it..

5. Testing of the developed technology

To test the developed technology, a custom software application was created. It supports two operational modes: object detection and coordinate estimation.

During flight, if a GPS signal is available, the software enables the extraction of descriptor sets for the target image from the onboard camera, allowing the inclusion of newly detected landmarks into the reference dataset. Landmark identification can be performed automatically or with the assistance of an operator. These landmarks are subsequently used to determine the UAV's position.

If necessary—such as in the event of GPS signal loss—the system initiates the algorithm for UAV geolocation estimation. For each image captured by the onboard camera, preprocessing operations can be applied to increase image processing speed for feature-based methods. These transformations may include resizing, image smoothing with various algorithms (using different hyperparameters), intensity adjustment, and reduction in the number of color channels. Such preprocessing reduces the number of keypoints, thereby decreasing the execution time of feature matching algorithms.

Following preprocessing, keypoints and their descriptors are extracted from the image acquired by the UAV's target payload camera. These descriptors are then matched with those in the reference dataset. A set of matched descriptors is obtained, corresponding to specific keypoints.

Next, the spatial density function of the matched keypoints is evaluated using a frequency histogram. A uniform partitioning $.\Delta_{s_x,s_y}$, is introduced, and the class (cell) with the highest frequency is selected. If this frequency does not exceed a defined threshold, the system proceeds to process the next frame.

Otherwise, for each matched point within the selected class, geographic coordinates are retrieved from the reference dataset. The most probable position of the target landmark is then estimated by analyzing the distribution of these coordinates—specifically, by identifying the center of the histogram bin with the maximum frequency.

The estimated coordinates of the landmark within the camera's field of view can subsequently be used to determine the UAV's position.

Figure 2 illustrates the algorithm implemented in the software application.

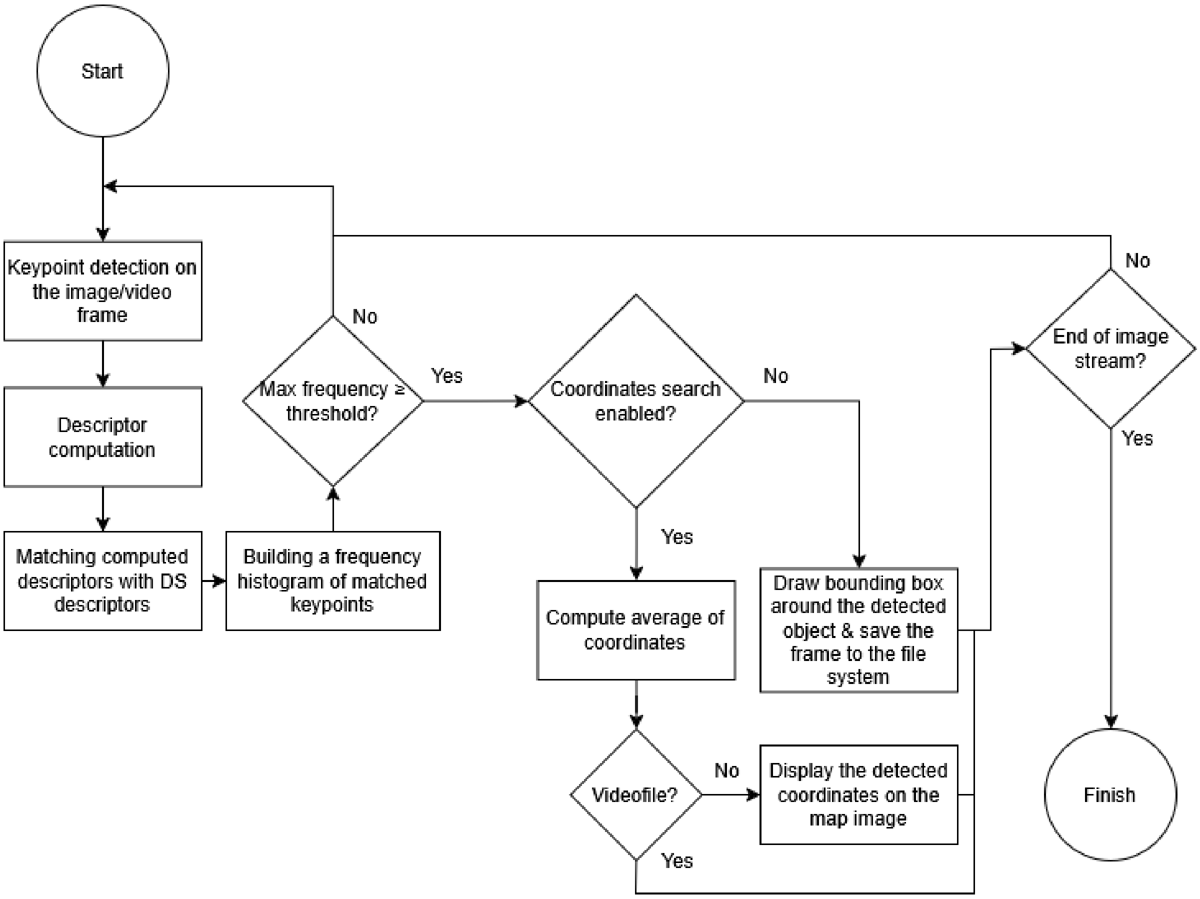

Start
Keypoint detection on the image/video frame
Descriptor computation
Matching computed descriptors with DS descriptors
Building a frequency histogram of matched keypoints
Max frequency ≥ threshold?
No
Yes
Coordinates search enabled?
No
Yes
Compute average of coordinates
Draw bounding box around the detected object & save the frame to the file system
Videofile?
No
Yes
Display the detected coordinates on the map image
End of image stream?
No
Yes
Finish

Figure 2: Block diagram of the application algorithm.

The block diagram shown illustrates the following steps:

1. Start – acquiring a frame of the terrain from the UAV camera. It is assumed that a reference dataset is available on the onboard computer.
2. Keypoint detection in the image/video frame – detecting keypoints in the image. The application can be tested on a target image simulating a map (with a pre-formed reference dataset) or on a video frame (captured by the UAV camera or extracted from a video file).
3. Descriptor computation – generating descriptors for the detected keypoints.
4. Matching descriptors with the reference dataset – searching for matches between descriptors from the input image and those in the reference dataset, and applying filtering rules if necessary.
5. Building a frequency histogram of matched keypoints – constructing a frequency histogram based on the number of matched keypoints falling into coordinate partition cells.
6. Is the maximum frequency ≥ threshold? – checking whether the number of matched keypoints in the most populated partition cell exceeds a predefined threshold. If not, the system proceeds to process the next input image.
7. Coordinate search? – verifying user-defined parameters to determine the next step. If the application is being tested only for object detection, proceed to step 8. Otherwise, continue to step 9.
8. Draw a bounding box around the detected object and save the frame – drawing a frame around the detected object and saving the image with the bounding box to the user's file system.
9. Estimate the average coordinate value – calculating the central coordinate of the histogram cell containing the target object.

10. Display detected coordinates on the map image – marking the estimated UAV position on the map-simulated target image.
11. End – terminating the application when the video stream ends (either from a file or a live feed) or when all images in the simulation have been processed.

Each algorithm used in the software includes several hyperparameters. These were selected based on publicly available technical documentation (SIFT, ORB) or through experimental tuning (SuperPoint). For the SuperPoint method, the recommended maximum number of detected keypoints is 2048, while 1024 is advised for improved performance. A comparison of performance between 1024 and 380 keypoints is shown in Table 2.

Table 2
Number of frames with successfully recognized object

Video file	1024 keypoints	380 keypoints
VideoFile1	24	25
VideoFile 2	25	31
VideoFile 3	35	33
VideoFile 4	9	5
VideoFile 5	24	27
Total	117	121

The comparative analysis of this network showed that the difference in object recognition accuracy between using 1024 and 380 keypoints is not significant. Moreover, using a smaller number of keypoints provides a slight advantage in recognition accuracy. Therefore, the number of keypoints was set to 380.

Table 3
Accuracy of methods in the object recognition task

	Correctly recognized					
Total frames	SIFT	SIFT with smoothing	ORB	ORB with smoothing	SuperPoint, LightGlue	SuperPoint, LightGlue with smoothing
435	288 (66%)	367 (84%)	253 (58%)	243 (56%)	121 (27.8%)	123 (28%)

Based on the collected data, we can evaluate the accuracy of object recognition. The chosen accuracy metric is the number of successful object recognitions in the video stream (Table 3).

6. Comparison of object search methods performance

We compare the performance of object detection methods depending on the specific hardware used. The following tables present averaged testing results of the information technology (IT) system on video footage recorded in various terrain conditions. Table 4 shows the performance of the IT system in object recognition on the microcomputer “Raspberry Pi 4 Model B” using different methods. Methods based on neural networks require a significant amount of processing time (more than 2 minutes) to handle a 1920×1080 video frame on this microcomputer and were therefore excluded from the comparison in Table 4.

Table 4
Average IT performance using “Raspberry Pi 4 Model B” in frames/s

Image processing	SIFT	ORB
No pre-processing	0.33726	4.9228
Smoothing	0.472	5.2366
Reducing linear dimensions by 2 times	1.614	11.139
Smoothing and reduction	1.7206	7.7492

The results of testing on the “Orange Pi 5 Pro” are contained in Table 5.

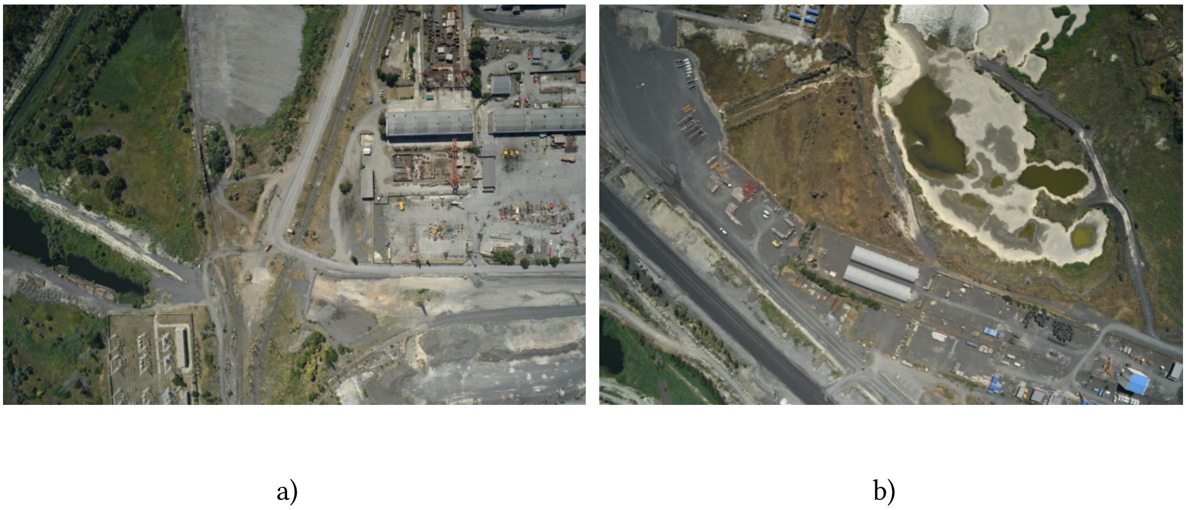
Table 5
Average IT performance using “Orange Pi 5 Pro” in frames/s

Image processing	SIFT	ORB	LightGlue+SuperPoint
No pre-processing	1.1304	9.8176	0.2088
Smoothing	1.3562	10.1886	0.2092
Reducing linear dimensions by 2 times	5.4344	20.9286	0.2134
Smoothing and reduction	5.2432	18.8096	0.2122

The test results indicate that the Orange Pi 5 Pro is more powerful and also features a neural processing unit (NPU). As a result, it outperforms the Raspberry Pi 4 Model B by more than three times in terms of processing speed.

7. Accuracy of finding object coordinates

The accuracy of coordinate estimation was evaluated by simulating flight over large images with a resolution of 8256 × 5504 pixels, representing a conceptual map (Fig. 3).



a) b)

Figure 3: Digital image for flight simulation.

After estimating the coordinates of the landmarks, the relative errors were calculated (Table 6) for seven landmarks in each type of terrain near Horishni Plavni, Ukraine (“Terrain 1” – Fig. 3a, “Terrain 2” – Fig. 3b).

Table 6
Relative error of coordinate estimation

Terrain 1		Terrain 2	
Fig.3 a)		Fig.3 b)	
landmark a.1.	1.3%	landmark b.1.	3%
landmark a.2.	0.4%	landmark b.2.	0.2%
landmark a.3.	0.7%	landmark b.3.	2%
landmark a.4.	0.5%	landmark b.4.	0.03%
landmark a.5.	12.9%	landmark b.5.	0.2%
landmark a.6.	1.41%	landmark b.6.	0.04%
landmark a.7.	9.8%	landmark b.7.	1.03%

It is worth noting that all planned landmarks were successfully detected in both Terrain 1 and Terrain 2.

8. Conclusions

1. A technology for automating UAV position support has been proposed, based on optical channel processing and the identification of landmark objects. The approach includes three methods for comparative evaluation under GPS-denied conditions: SIFT (Scale-Invariant

Feature Transform), ORB (Oriented FAST and Rotated BRIEF), and neural network architectures LightGlue and SuperPoint.

2. The proposed implementation combines these methods with a custom approach based on evaluating the spatial density function of the coordinates of detected landmark keypoints.
3. A comparative analysis of the accuracy and performance of the selected methods was conducted across different types of hardware by simulating UAV flights over large images with a resolution of 8256 × 5504 pixels, representing various terrain conditions. Seven landmarks were evaluated per terrain. The maximum coordinate estimation error reached approximately 10% for two landmarks in one of the terrains. For all other landmarks, the error was approximately 1%.
4. The feasibility of applying landmark detection methods on "Orange Pi 5 Pro" microcomputers was assessed and compared with "Raspberry Pi 4 Model B." The latter is not recommended due to its low computational speed.
5. Test results showed that the processing speed of the proposed methods significantly depends on prior image smoothing. After smoothing, the number of detected keypoints is reduced, which accelerates feature matching. Furthermore, smoothing improves keypoint stability, retaining only the most robust features for processing.
6. It is recommended to use the SIFT method on the "Orange Pi 5 Pro" when recognition accuracy is more important than speed in a particular flight task. SIFT has higher robustness to image scaling and demonstrates good recognition accuracy even after image downscaling. According to test results, SIFT achieved 84% recognition accuracy with image smoothing (66% without), compared to approximately 58% for ORB.
7. It is recommended to use the ORB method on both "Raspberry Pi 4 Model B" and "Orange Pi 5 Pro," as it demonstrated near real-time processing speeds during testing.
8. Future research will focus on testing the proposed technology on the NVIDIA Jetson Nano microcomputer, which is expected to provide near real-time performance. Additional development will aim to enhance UAV navigation support by determining the UAV's position based on the coordinates of identified landmark objects.

Declaration on Generative AI

During the preparation of this work, the authors used GPT4o in order to translate research notes and results from Ukrainian to English. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

References

[1] Buryi P., Pristavka P., Sushko V. Automatic definition the field of view of camera of unmanned aerial vehicle. Science-intensive technologies, 2(30) (2016), pp. 151–155. URL: https://ouci.dntb.gov.ua/en/works/4Lq0DAVl/

[2] Nixon M. S., Aguado A. S. Feature Extraction and Image Processing. Oxford, Auckland, Boston, Johannesburg, Melbourne, New Delhi: Newnes. 2nd edn. (2008). URL: https://www.cl72.org/090imagePLib/books/book2-Nixon,Aguado-feachureExtraction-ImageProcessing-.pdf

[3] Prystavka, P., Dukhnovska, K., Kovtun, O., Cholyshkina, O., Semenov, V. Recognition of aerial photography objects based on data sets with different aggregation of classes. Eastern-European Journal of Enterprise Technologies. 2-121 (2023), pp. 6–13. doi.org/10.15587/1729-4061.2023.272951.

[4] J.Gallo E., Barrientos A. Long-Distance GNSS-Denied Visual Inertial Navigation for Autonomous Fixed-Wing Unmanned Air Vehicles: SO(3) Mainifold Filter Based on Virtual Vision Sensor. Aerospace. 10(8):708 (2023). URL: https://www.mdpi.com/2226-4310/10/8/708

[5] Antonopoulos A, Lagoudakis MG, Partsinevelos P. A ROS Multi-Tier UAV Localization Module Based on GNSS, Inertial and Visual-Depth Data. Drones. 6(6):135 (2023). doi.org/10.3390/drones6060135

[6] Elamin A, Abdelaziz N, El-Rabbany A. A GNSS/INS/LiDAR Integration Scheme for UAV-Based Navigation in GNSS-Challenging Environments. Sensors. 22(24):9908 (2022). URL: https://www.mdpi.com/1424-8220/22/24/9908

[7] Ashraf S, Aggarwal P, Damacharla P, Wang H, Javaid AY, Devabhaktuni V. A low-cost solution 4 for unmanned aerial vehicle navigation in a global positioning system-denied environment. International Journal of Distributed Sensor Networks. 14(6) (2018). doi.org/10.1177/1550147718781750

[8] Bigazzi L, Basso M, Boni E, Innocenti G, Pieraccini M. A Multilevel Architecture for Autonomous UAVs. Drones. 5(3):55 30(2021). doi.org/10.3390/drones5030055

[9] Yu Dam Lee, La Woo Kim, Hyung Keun Lee. A tightly-coupled compressed-state constraint 6 Kais eir fer interated is at ariar: Cona Navagain. zelie Solem navision in. doi.org/10.1049/rsn2.12265

[10] Yang Y, Liu X, Zhang W, Liu X, Guo Y. A Nonlinear Double Model for Multisensor-Integrated 7 Navigation Using the Federated EKF Algorithm for Small UAVs. Sensors. 20(10):2974 (2020). doi.org/10.3390/20102974

[11] Ashraf A. Deraz, Osama Badawy, Mostafa A. Elhosseini, Mostafa Mostafa, Hesham A. Ali, Ali I. 8 -Desouky. Deep learning based on LSTM model for enhanced visual odometry navigation system, Ain Shams Engineering Journal. Volume 14, Issue 8 (2023). doi.org/10.1016/j.asej.2022.102050

[12] B. Or and I. Klein, "Adaptive Step Size Learning With Applications to Velocity Aided Inertial 9 Navigation System," in IEEE Access, vol. 10, pp. 85818-85830 (2022). doi.org/10.1109/ACCESS.2022.3198672

[13] Y. Dang, C. Benzaïd, B. Yang, T. Taleb and Y. Shen, "Deep-Ensemble-Learning-Based GPS 10 Spoofing Detection for Cellular-Connected UAVs," in lEEE Internet of Things Journal, vol. 9, no. 24, pp. 25068-25085, 15 Dec. 15 (2022). doi.org/10.1109/JSPECIAL POINTS.2022.3195320

[14] Iatsyshyn, A., Iatsyshyn, A., Kovach, V., Zinovieva, I., Artemchuk, V., Popov, O., Turevych, A. Application of open and specialized geoinformation systems for computer modelling studying by students and PhD students. Paper presented at the CEUR Workshop Proceedings, 2732 893-908 (2020). URL: https://ceur-ws.org/Vol-2732/20200893.pdf

[15] Hubanova, T., Shchokin, R., Hubanov, O., Antonov, V., Slobodianiuk, P., & Podolyaka, S. Information technologies in improving crime prevention mechanisms in the border regions of southern Ukraine. Journal of Information Technology Management, 13, 75-90 (2021). doi:10.22059/JITM.2021.80738

[16] Prystavka P., Cholyshkina O., Sorokopud T. Experimental Study of Distributions Differential Invariants Based on Spline Image Models / CMiGIN 2022: 2nd International Conference on Conflict Management in Global Information Networks, Kyiv, 2022, pp.163-172. URL: https://ceur-ws.org/Vol-3530/paper16.pdf

[17] Prystavka P., Cholyshkina O. Comparative analysis of differential invariants based on the spline model for various image distortion. Advanced Information System. V.4, № 4 (2020), pp. 70–76. URL: http://ais.khpi.edu.ua/article/view/2522-9052.2020.4.10/218505.

[18] Prystavka P., Chirkov A.V., Sorokopud V.I., Zdota D.V. Simulation testing of the information technology of aircraft navigation by optical channel // Scientific technologies. No. 3(47) (2020), pp. 370–377. URL: https://ouci.dntb.gov.ua/en/works/9GPJwmP7/

[19] Naruoka, M, and Tsuchiya, T, High. High performance naviga-tion system with integration of low precision MEMS INS and general-purpose GPS . Transactions of the Japan Society for Aeronautical and Space Sciences. 50(170):284-292 (2008). doi: 10.2322/tjsass.50.284.

[20] F.M. Zakharin, S.A. Ponomarenko. «Unmanned Aerial Vehicle Integrated Navigation Complex» 2017 IEEE 4th International Conference Actual Problems of Unmanned Aerial Vehicles

Developments (APUAVD). Vol. 3 No. 53 (2017). pp. 75-83. doi: https://doi.org/10.18372/1990-5548.53.12146

[21] Ponomarenko S.O., Zakharin F.M. Features of modeling the process of complex processing of navigation information in on-board complexes of unmanned aerial vehicles. XIX International Conference DYNAMICAL SYSTEM MODELING AND STABILITY INVESTIGATION Kyiv, (2019), pp. 189-192. URL: http://www.dsmsi.univ.kiev.ua/downloads/book_DSMSI-2019%20final.pdf

[22] Prystavka P.O., Cholyshkina O.G. Polynomial splines in alternative navigation problems based on aerial survey data: monograph. Interregional Academy of Personnel Management, 2022. URL: https://maup.com.ua/ua/pro-akademiyu/novini1/usi-novini1/vijshla-z-druku-monografiya-polinomialni-splajni-v-zadachi-alternativnoi-navigacii-za-danimi-aerozjomki.html