Comparative analysis of SIFT and SURF methods for local feature detection in satellite imagery

Artem Riabko*1,*,† and Yuliya Averyanova*1,†*

1 National Aviation University, Liubomyra Huzara Ave., 1, Kyiv, 03058, Ukraine

Abstract

This paper describes the local feature detection as a key component in computer vision and image processing which perform tasks such as object recognition, image matching and mapping. Various algorithms and techniques used for detecting distinctive features in satellite imagery, the most popular of them are Scale-Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF) detectors. The goal is to provide insights into the strengths and limitations of these methods, describing an accurate understanding of their applicability in satellite image processing. The experimental evaluation of local feature detection methods using the MATLAB environment is demonstrated and discussed. The findings aim to guide researchers and practitioners in selecting suitable local feature detection approaches for diverse satellite image analysis applications.

Keywords
computer vision, image processing, satellite imagery, local feature detectors, SIFT, SURF, feature extraction, objects recognition

1. Introduction

In modern world satellite images play an important role in improving our understanding and management of Earth's surface. These images provide a necessary data for survey of various features of the Earth, including changes in land utilization, natural disasters, climate patterns and agricultural practices. Furthermore, satellite imagery is integral to navigation, communication, defense and the exploration of outer space. The variety of satellite images makes them essential for informed decision-making and resource management across a wide array of disciplines [1]. One of the most difficult fields and applications to work on is object identification and detection. Because of the complex nature of raw satellite data sophisticated analysis techniques are required, one of the commonly used and base techniques is local feature detection [2].

This paper conducts evaluation review of the most popular local feature detection methods, focusing on their applicability to satellite images. The detectors Scale-Invariant Feature Transform (SIFT) and Speeded Up Robust Features (SURF) each chosen for its

distinctive characteristics and relevance in satellite image analysis. The evaluation covers a careful exploration of the strengths and limitations of each method with an emphasis on their performance in different scenarios [3].

The paper addresses challenges proper to satellite imagery such as different lighting conditions, complex terrains and atmospheric interferences. Gaining expertise in local feature recognition techniques is essential in a constantly changing environment where data-driven insights are critical. The findings of this review aim to guide researches of this field in selecting the most suitable detector for specific satellite image analysis applications. As technology advances, the ability to accurately detect and analyze local features becomes integral to harnessing data-driven insights and fostering progress across various disciplines [4]. The investigation is carried out within the MATLAB environment using the Image Processing Toolbox.

2. Description of local feature detectors

Local feature detection has become essential in computer vision due to its ability to extract meaningful information from images. It enables computers to recognize complex patterns, track objects and understand visual data in various applications. Its strength lies in its adaptability and robustness, making it a key component for tasks such as image recognition and object tracking [5].

Distinctive patterns and keypoints called "local features" are useful in images. They stay recognizable even when the image is changed in size, rotated or has different lighting. The key to detecting these local features is finding keypoints like corners, edges or small areas that stand out compared to the rest of the image [6].

A feature is an important piece of information for solving a computer-related task in a particular application. For instance, features can be specific details like dots, lines or objects visible in an image. Features can also be detected by a general operation or tool applied to an image to identify important details. When an image goes through transformations such as movement, rotation or scaling along two axes and the recognition of specific points within the image remains consistent, it shows that those marked points have consistent features. These points are considered to be feature points which are distinctive and robust against various image transformations [7, 8].

Main components of local feature detection are definition, description and matching. Definition: identify the interest point. Description: each interest point's local appearance is determined in a way that remains unchanged regardless variations in lighting, translation, scale and in-plane rotation. A descriptor vector is typically obtained for each feature point. Matching: to find common characteristics among images, their descriptors are compared. A set of pairs may be obtained for two images $(X_i, Y_i) \leftrightarrow (X_i', Y_i')$, where (X_i, Y_i) is a feature in one image and (X_i', Y_i') its matching feature in the other image [9, 10].

Computer vision employs various techniques for finding local features, each with unique strengths and limitations. SIFT is popular for identifying keypoints across different sizes, angles and lighting. SURF is an efficient alternative to SIFT, offering comparable robustness. Another popular local feature detector is Binary Robust Independent Elementary Features (BRIEF) prioritizes speed by using predefined binary tests around key points, making it

ideal for real-time applications or other techniques include Harris, Shi-Tomasi, KAZE and BRISK, each designed for specific needs. Various methods are available to those who work in this field, giving a number of alternatives to meet various demands. By considering factors like the efficiency of the computation, the reliability of the method and the requirement for real-time functionality in the specific application, users can select the most appropriate approach from this scope of methods [11].

There are still certain issues that need to be resolved in local feature detection despite improvements in the field. These issues include things like covered or hidden objects, a big amount of information and lighting variations. Investigators are trying to find solutions for these issues in order to improve the accuracy and functionality of local feature detection. They are attempting to enhance the system's capacity to identify complex visual patterns by utilizing machine learning, particularly deep learning [12].

3. The operational algorithms of SIFT and SURF

Paper provides a concise overview of key methods for object detection using local feature: the SIFT and SURF algorithms. By analyzing the algorithms in detail, their complex methods, advantages and applications are aimed to be clarified. Through this exploration, a deeper understanding of these methods is provided to professionals involved in computer vision.

3.1. Scale Invariant Feature Transform

The SIFT is an image-matching algorithm in data science that uses to identify key features in images and compare these characteristics to a new image of the same object. SIFT computes descriptors and identifies critical locations based on their local intensity extrema that capture the local image information around those key points. These descriptors can then be used for tasks like image matching, object recognition and image retrieval. SIFT features, also known as over-edge features or hog features, have the main benefit of being independent of imagine size and orientation. Identifying key image features is essential even in the presence of noise. These features should remain consistent regardless of scale changes. Let's explore these key concepts individually to enhance our understanding [13].

First, the Gaussian Blurring technique is used to reduce the noise in an image. For this purpose, picture is subsequently blurred using a Gaussian convolution. In mathematics, the convolution of the image with the Gaussian operator is referred to as "blurring." Every pixel in a Gaussian blur has a specific phrase or "operator," applied to it. The image appears blurry as a result.

$$L(x, y, \sigma) = G(x, y, \sigma) \, I(x, y), \tag{1}$$

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}}, \tag{2}$$

where G is the Gaussian Blur operator and I is an image, x, y are the location coordinates and σ is the "scale" parameter.

This approach helped in image processing and successfully removed the noise from the images and highlight the important features of the image. Then it is needed to ensure that these features are scale-dependent, so scale space is created for this purpose. Scale space is

a collection of images having different scales generated from a single image which follows by a sequence of further convolutions with increasing standard deviation.

As a discrete approximation of this continuous space is necessary, a technique called Difference of Gaussians (DoG) will be used. The Difference of Gaussian method is a feature improvement technique that includes excluding a blurry original picture version from a less blurry original image version. DoG eliminates each image from the previous image in a similar scale to generate a new collection of images for each octave. Finding the keypoints is the next stage after creating another set of images.

The goal is to locate the images' local maxima and minima. To locate the local maxima and minima, every pixel in the image is examined and compared with its neighboring pixels. A discrete maximum, in this context, is defined as a pixel whose gray value is larger than those of all its 26 neighboring pixels, while a discrete minimum is defined analogously. Here as "neighbors" the eight adjacent pixels in the same picture are count, the corresponding two pixels in the adjacent pictures in the same octave and finally their neighbors in the same picture.

Many keypoints are produced from the keypoints created in the previous stage. Some of them lack contrast or are located close to an edge. In order to determine the extrema with more accuracy, the Taylor series expansion of the scale space is used: if the level of intensity at this extremum is less than a threshold value of 0.03 it is discarded. Edges must also be eliminated as DoG responds more strongly to them. They used a 2x2 Hessian matrix to compute the principal curvature. Now both the contrast test and the edge test have performed to reject the unstable keypoints [14].

To make the rotation invariant, now each keypoint an orientation value is given. Depending on the scale a neighborhood is drawn around the keypoint position and the gradient's magnitude and direction are determined there. A 360-degree orientation histogram with 36 bins is produced. The orientation indicates the direction of the pixel while the magnitude indicates its intensity. Now that the pixel magnitude and orientation data have been obtained and a histogram is generated. There would come a peak in this histogram. The direction of the keypoint will be the bin where we observe the peak.

The last step is to create a distinct fingerprint or so called "descriptor," for this keypoint by utilizing the nearest pixels' orientations and magnitudes. Also, the descriptors will be partially invariant to the illumination or brightness of the images. First take a 16×16 neighborhood around the keypoint. This 16×16 block is further divided into 4×4 sub-blocks and for each of these sub-blocks, the histogram using magnitude and orientation is generated. Each arrow corresponds to the 8 bins with the length of each arrow indicating its magnitude. As a result, it is a total of 128 bin values for every keypoint [15].

3.2. Speeded-Up Robust Features

The SURF algorithm is a popular method for detecting and describing keypoints in images. It is designed to be faster while maintaining robustness to changes in scale, rotation and illumination.

The algorithm begins by computing the integral image of the input image. The integral image is used for fast summation of pixel values within any rectangular region of the image.

The average intensity inside one specific image may also be determined using it. They provide efficient calculation of box-type convolution filters [16].

The method for detecting interest points is based on a simple Hessian matrix approximation. SURF focuses on the Hessian matrix's determinant to choose both the location and the scale, compared to applying separate measures for each. For adapt to any scale, the image is filtered by a Gaussian, so a point X (x, y) the Hessian matrix H (x, σ) in x at scale σ is defined as:

$$H(x, \sigma) = \begin{bmatrix} L_{xx}(x, \sigma) & L_{xy}(x, \sigma) \\ L_{yx}(x, \sigma) & L_{yy}(x, \sigma) \end{bmatrix}, \tag{3}$$

where $L_{xx}(x, \sigma)$ is the convolution of the Gaussian second order derivative $\frac{\partial^2}{\partial x^2} \partial(\sigma)$ with the image I in point X and the same for $L_{yx}(x, \sigma)$ and $L_{yy}(x, \sigma)$.

SURF improves on SIFT's use of Laplacian of Gaussian (LoG) approximations by employing box filters for both convolution and second-order derivatives. Box filters resemble approximate Gaussian derivatives and can be efficiently evaluated using integral images, regardless of the image size. This efficient approximation contributes to the speed of SURF. The Hessian is computed as follows

$$\det(H_{approx}) = D_{xx}D_{yy} - (0.9D_{xy})^2, \tag{4}$$

where D_{xx}, D_{yy}, D_{xy} is a convolution filters.

Scale spaces are commonly created by using image pyramids. The process involves smoothing the images with a Gaussian filter and then downsampling them to form a higher level of the pyramid as it was in SIFT. By utilizing box filters and integral images, SURF can apply filters of various sizes to the original image at the same speed, without the need for iterative processing. This allows for analysis of the scale space by increasing the filter size (9×9 → 15×15 → 21×21 → 27×27, and so on) rather than repeatedly reducing the image size. With each new octave, the filter size is doubled while the sampling intervals for extracting points of interest are adjusted accordingly.

On the stage of feature description when identifying interest points, SURF aims to be unaffected by rotation by determining a consistent orientation for them. To do this, SURF calculates the Haar-wavelet responses in both the x and y-direction within a circular neighborhood with a radius of 6S around the keypoint, where S represents the scale at which the keypoint was found. The vertical and horizontal wavelet responses within a specific area of scanning are added up. Then the scanning orientation is adjusted by adding π/3 and the calculations are repeated until the orientation that gives the highest total value is identified. This identified orientation is considered as a primary orientation of the feature descriptor.

The descriptor is extracted by first creating a square region around the key point with the same orientation. The window size is set to 20 times the scale parameter. Then the area is divided into smaller 4×4 square sections. Simple features are calculated at 5×5 sample points within each section. The horizontal Haar wavelet response is referred to as dx and the vertical Haar-wavelet response as dy. The responses from wavelets dx and dy for each subregion are combined to create the initial entries for the feature vector. Additionally, to capture information on the polarity of intensity changes, the sum of the absolute values of dx and dy responses, denoted as $|dx|$ and $|dy|$ respectively, is computed. As a result, each

subregion is represented by a descriptor vector V with four dimensions, representing the underlying intensity structure $V = (\sum dx, \sum dy, \sum |dx|, \sum |dy|)$. This leads to a 64-dimensional descriptor vector encompassing all 4x4 subregions [17, 18].

4. Methods comparison and evaluation

This section will provide an experimental evaluation of SUFR and SIFT local feature detection methods in the MATLAB environment.

Around 50 experiments were conducted within assessment, it was testing a variety of satellite images with wide range of resolutions and objects located on them. Despite the fact that the images contained in each experiment varied greatly in content, also as environmental conditions that may have impacted them, common findings related to the performance features of SURF and SIFT results tendency were almost the same in all experiments. Based on this data and in order to reflect experimental results this section will be focusing on the one satellite image [19]. The performance of the SURF and SIFT methods can be effectively illustrated by presenting the findings from a representative satellite image of Athens' ship port, Figure 1.

Figure 1: Athens' ship port satellite image.

The satellite image of Athens' ship port with resolution 5463x7475 was chosen due to its wide range and intensity of objects located on it. The port area likely covers a diverse number of objects such as ships, docks, buildings and vehicles, providing a rich dataset for analysis. The complexity of the scene within the port brings closer to the real-world scenarios of objects detection.

The main goal of this experiment was to detect cruise liner showed on Figure 2 in Athens' ship port.

Figure 2: Image of cruise liner that need to be detected.

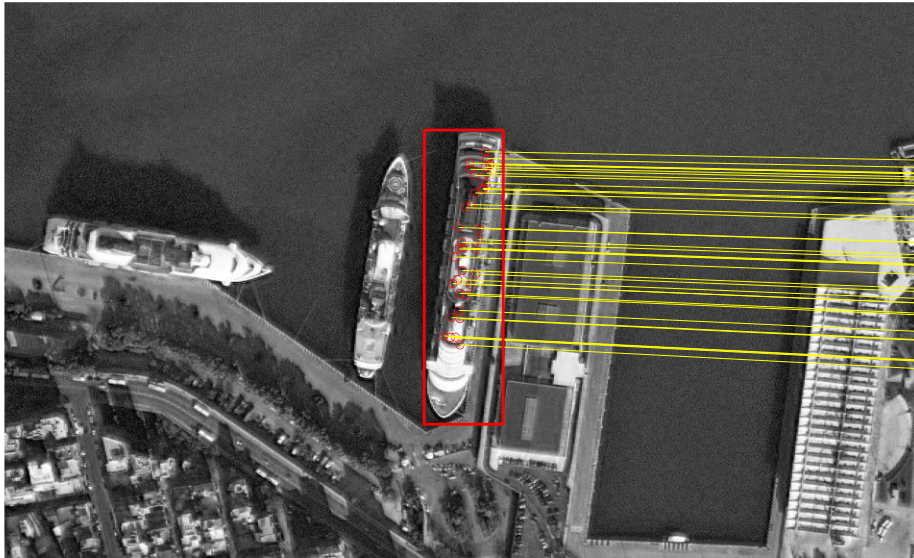The results of detection are presented on the Figure 3.

Figure 3: Detected cruise liner.

The number of detected key points of both port and ship images was determined to evaluate effectiveness of SURF and SIFT methods. This numerical study allows us to understand better how local-feature detectors work when confronted with disorganized environment such as harbor scenes where ships are docked against them. By looking at how many key points are detected using different approaches it's easy to see ability to accurately identify relevant features. In the analysis it was assumed that the satellite image remained unchanged [20]. Number of key points on Athens' ship port satellite image using SURF and SIFT methods showed on Figure 4. The evaluation revealed that the SIFT method consistently outperformed the SURF method in terms of keypoints detection. Obviously SIFT is computationally more intensive than SURF, despite its higher computational complexity, SIFT's effectiveness in detecting and matching key points prevails any potential disadvantages in terms of processing time [21, 22, 23]. In order to investigate scale variance of both methods objects were systematically rotated the at different angles and analyzed the corresponding variations in the number of detected key points (Figure 5). It was gained conception into the algorithms' sensitivity to rotational changes in the scene. The analysis revealed a periodic dependence of the number of key points detected on the cruise liner image based on both the detection method performed and the rotation angle of the object.
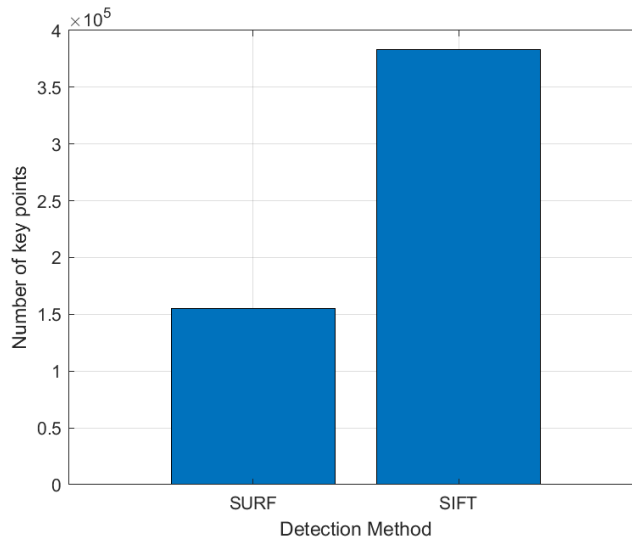
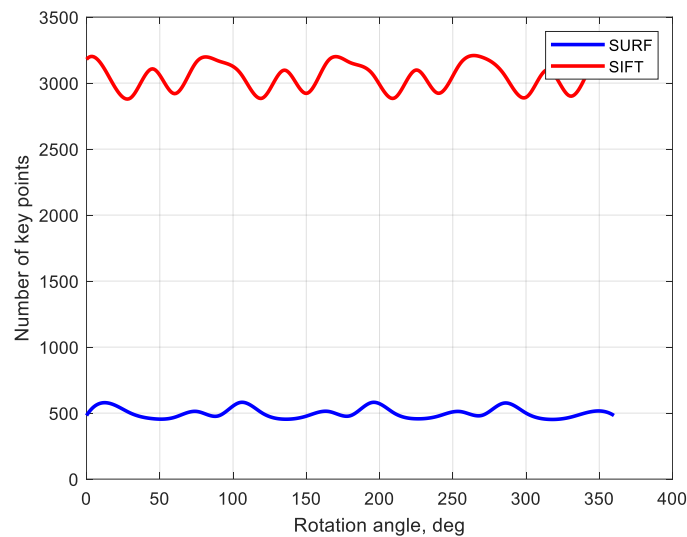**Figure 4:** Number of key points of Athens' ship port satellite image.

Figure 5: Number of key points of cruise liner image depending on detection method and rotation angle.

Notably, this periodic dependence exhibited a consistent pattern, with peaks and troughs occurring at regular intervals of every 90 degrees of rotation (Figure 6). The observed periodicity every 90 degrees suggests a strong correlation between the geometric properties of the cruise liner image and the behavior of the key point detection algorithms. At certain orientations, such as when the cruise liner is aligned parallel or perpendicular to the image axes, the features of the object may be more prominent and easily detected by the algorithms [24, 25]. The algorithms' effectiveness can be evaluated by counting the number of matching points so the dependence of rotation angle and the number of matched points between corresponding key points in the two images was also investigated.
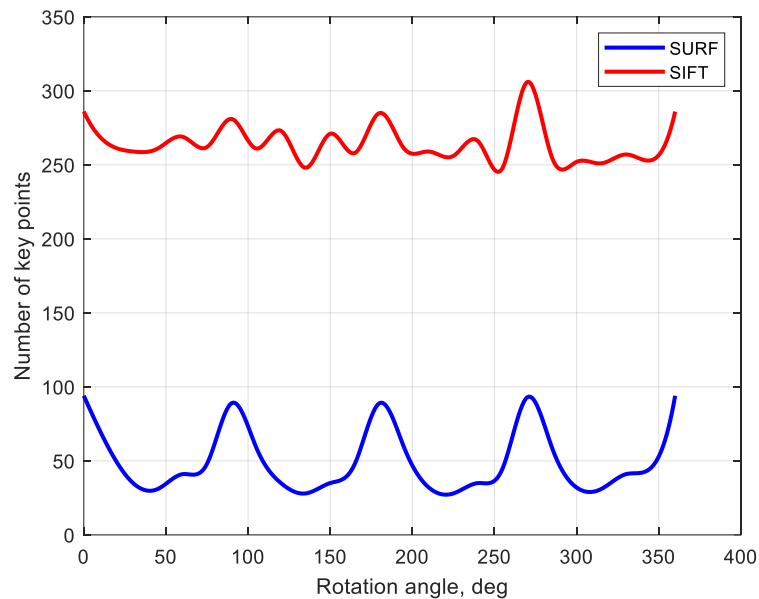
Figure 6: Number of matched points depending on detection method and rotation angle.

SURF was discovered to show periodic dependency with differences in the number of matched points seen at various rotation degrees. On the other hand, there was no apparent pattern in SIFT's reaction to rotation angles. A peak in matched points was noted at a precise rotational angle of about 270 degrees. This suggests that even in non-traditional orientations SIFT is capable of identifying features.

5. Conclusions

Examining how well the SURF and SIFT techniques performed for local feature detection on satellite pictures gave important information about how they behaved in various viewing scenarios. Although SIFT can capture more key points, SURF might be the better choice for situations requiring extensive feature extraction and analysis due to its greater processing efficiency. In general, SURF processes data more quickly than SIFT. On the other hand, SIFT typically finds more key points in satellite images.

For tasks where real-time processing and computational efficiency are critical, SURF may be preferred due to its faster processing times. On the other hand, SIFT's strength lies in its robustness and ability to detect a higher number of key points even in complex and challenging imaging conditions. As a result, the choice between SURF and SIFT depends on factors such as the particular objectives of the study, the complexity of the satellite images, the available computational resources and the preferred alternatives between processing speed and feature detection capabilities. By considering these factors, researchers and practitioners can select the satellite image analysis technique that best suits their needs.

References

[1] Y. Averyanova, V. Larin, N. Kuzmenko, I. Ostroumov, M. Zaliskyi, O. Solomentsev, O. Sushchenko, Y. Bezkorovainyi, Turbulence detection and classification algorithm using data from AWR, in: Proceedings of IEEE 2nd Ukrainian Microwave Week (UkrMW), Kyiv, Ukraine, 2022, pp. 518–522. doi: 10.1109/UkrMW58013.2022.10037172.

[2] T. Lindeberg, Scale selection, Computer Vision: A Reference Guide, Springer, 2014, pp. 701–713. doi: 10.1007/978-3-030-63416-2_242.

[3] E. Ristani, F. Solera, R. Zou, R. Cucchiara, C. Tomasi, Performance measures and a data set for multi-target, multi-camera tracking, Lecture Notes in Computer Science, vol. 9914, 2016, pp. 17–35. doi: 10.48550/arXiv.1609.01775.

[4] A. Riabko Methods of satellite images segmentation analysis, in: Proceedings of 7th IEEE International Conference on Methods and Systems of Navigation and Motion Control (MSNMC), Kyiv, Ukraine, 2023, pp. 163–167. doi: 10.1109/MSNMC61017.2023.10329167.

[5] Z. Zhao, BALF: Simple and Efficient Blur Aware Local Feature Detector, IEEE/CVF Winter Conference on Applications of Computer Vision (WACV), Waikoloa, USA, 2024, pp.3350–3360. doi: 10.1109/WACV57701.2024.00333.

[6] J. Shi, C. Tomasi, Good features to track, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Seattle, USA ,1994, pp. 593–600.

[7] D. G. Lowe, Distinctive image features from scale-invariant keypoints, International Journal of Computer Vision 60 (2) (2004) 91–110.

[8] H. Zhang, E. Fromont, S. Lefevre, B. Avignon, Guided attentive feature fusion for multispectral pedestrian detection, in: Proceedings of IEEE Winter Conference on Applications of Computer Vision, Waikoloa, USA, 2021, pp. 72–80. doi: 10.1109/WACV48630.2021.00012.

[9] H. Hu, J. Gu, Z. Zhang, J. Dai, Y. Wei, Relation networks for object detection, in: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, art. no. 8578476, 2018, pp. 3588–3597. doi: 10.48550/arXiv.1711.11575.

[10] G. Bokman, F. Kahl, A case for using rotation invariant features in state of the art feature matchers, in: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, New Orleans, USA, 2022, pp. 5106–5115. doi: 10.48550/arXiv.2204.10144.

[11] D. Detone, T. Malisiewicz, A. Rabinovich, SuperPoint: Self-supervised interest point detection and description, in: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, Salt Lake City, USA, 2018, pp. 337–349. doi: 10.48550/arXiv.1712.07629.

[12] Y. Jin, D. Mishkin, A. Mishchuk, J. Matas, P. Fua, K. M. Yi, E. Trulls, Image matching across wide baselines: from paper to practice, International Journal of Computer Vision 129 (2) (2021) 517–547. doi: 10.48550/arXiv.2003.01587.

[13] A. Singh, SIFT Algorithm: How to use SIFT for image matching in Python, 2024. URL: https://www.analyticsvidhya.com/blog/2019/10/detailed-guide-powerful-sift-technique-image-matching-python.

[14] D. Tyagi, Introduction to SIFT, 2019. URL: https://medium.com/@deepanshut041 /introduction-to-sift-scale-invariant-feature-transform-65d7f3a72d40.

[15] D.G. Lowe, Object recognition from local scale-invariant features, in: Proceedings of the International Conference on Computer Vision, Kerkyra, Greece, 1999, pp. 1150–1157. doi: 10.1109/ICCV.1999.790410.

[16] E. Oyallon, J. Rabin, An analysis and implementation of the SURF method, and its comparison to SIFT, Image Processing On Line, 2015. doi: 10.5201/ipol.2015.69.

[17] H. Bay, A. Ess, T. Tuytelaars, L. van Gool, SURF: Speeded up robust features, Computer Vision and Image Understanding 110 (3) (2008) 346–359. doi: 10.1016/j.cviu.2007.09.014.

[18] H. Bay, T. Tuytelaars, L. van Gool, SURF: Speeded up robust features, in: Proceedings of the ninth European Conference on Computer Vision, 2006. doi: 10.1007/11744023_32.

[19] K. A. Elorabi, A. Zekry, W.A. Mohamed, Optimizing SIFT algorithm parameters for better matching UAV and satellite images, Journal of Physics: Conference Series 2616 (1). doi: 10.1088/1742-6596/2616/1/012044.

[20] X. Zhao, H. Li, P. Wang, L. Jing, An image registration method using deep residual network features for multisource high-resolution remote sensing images, Remote Sensing 13(17), 3425. doi:10.3390/rs13173425.

[21] K. Dergachov, O. Havrylenko, V. Pavlikov, S. Zhyla, E. Tserne, V. Volosyuk, et al., GPS usage analysis for angular orientation practical tasks solving, in: Proceedings of IEEE 9th International Conference on Problems of Infocommunications, Science and Technology (PIC S&T), Kharkiv, Ukraine, 2022, pp. 187–192. doi: 10.1109/PICST57299.2022.10238629.

[22] M. Zaliskyi, O. Solomentsev, V. Larin, Y. Averyanova, N. Kuzmenko, I. Ostroumov, O. Sushchenko, Y. Bezkorovainyi, Model building for diagnostic variables during aviation equipment maintenance, in: Proceedings of the 17th International Conference on Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 2022, pp. 160–164. doi: 10.1109/CSIT56902.2022.10000556.

[23] N. Kuzmenko, I. Ostroumov, Y. Bezkorovainyi, Y. Averyanova, V. Larin, O. Sushchenko, M. Zaliskyi, O. Solomentsev, Airplane flight phase identification using maximum posterior probability method, in: Proceedings of the 3rd International Conference on System Analysis & Intelligent Computing (SAIC), Kyiv, Ukraine, 2022, pp. 1–5. doi: 10.1109/SAIC57818.2022.9922913.

[24] O. Solomentsev, M. Zaliskyi, O. Sushchenko, Y. Bezkorovainyi, Y. Averyanova, I. Ostroumov, V. Larin, N. Kuzmenko, Data processing through the lifecycle of aviation radio equipment. in: Proceedings of the 17th International Conference on Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 2022, pp. 146–151. doi: 10.1109/CSIT56902.2022.10000844.

[25] V. Megha, K.K. Rajkumar, Automatic satellite image stitching based on speeded up robust feature, in: Proceedings of 1st IEEE International Conference on Artificial Intelligence and Machine Vision, Gandhinagar, India, 2021, pp. 1–6. doi: 10.1109/AIMV53313.2021.9670954.