UDC 004.93
DOI https://doi.org/10.32782/2663-5941/2026.3.1/09

Goyushova U.M.
https://orcid.org/0000-0002-3646-4937
Institute of Mathematics

STRUCTURAL VISION-BASED UAV NAVIGATION USING HESS MATRIX AND B-SPLINE MODELING

This article presents a structural approach to the vision-based navigation of unmanned aerial vehicles (UAVs) based on the detection, parametric modeling, and comparison of curvilinear objects extracted from aerial images. In contrast to conventional image-matching techniques that rely directly on pixel-level intensity values or raw visual appearance, the proposed method represents objects through their geometric and structural characteristics. Such a representation reduces the influence of scale variation, translation, partial viewpoint changes, and other distortions that often degrade the reliability of direct image comparison in real navigation scenarios. As a result, the method is more robust and better adapted to UAV operation in environments where the appearance of observed objects may change significantly.

A central component of the proposed approach is the selection of control points for B-spline approximation using the Hessian matrix. Hessian-based analysis makes it possible to identify the most informative contour points by evaluating the local second-order structure of the image and emphasizing regions with significant curvature changes. This allows the extraction of essential structural elements of curvilinear objects while suppressing redundant or less informative details. The selected points are then used to construct a B-spline model that provides a smooth, compact, and stable parametric representation of object shape.

The combination of Hessian-based point selection and B-spline modeling forms a mathematically grounded framework for the reliable description, comparison, and identification of curvilinear objects. A distinctive feature of the proposed method is that comparison is performed not on the original images themselves, but on spline curves and their characteristic parameters. This significantly reduces computational complexity and increases interpretability. Owing to its structural nature, the method is particularly suitable for UAV navigation tasks in which lightweight, robust, and invariant object representation is required for matching current observations with previously stored reference data and for supporting accurate localization in visually complex scenes.

Keywords: *UAV, vision-based navigation, viewpoint invariance, Hessian matrix, B-spline, curve detection, OpenCV.*

Formulation of the problem. In vision-based UAV navigation, one of the key tasks is the comparison of visual objects extracted from the current frame with previously stored reference images or structural models. For this purpose, geometric features of roads, rivers, boundary lines, and other elongated objects can be used. Since such objects provide more stable visual landmarks along the route, extracting their contour and curvature characteristics helps improve localization accuracy. Especially in GNSS-denied environments, representation methods based on structural features are considered both computationally efficient and suitable for real-time comparison [1].

Analysis of recent research and publications. Detecting curvilinear structures in digital images is one of the important theoretical and applied problems in digital image processing and computer vision. Since curves represent object boundaries, contours, trajectories, and shape characteristics, their reliable extraction is essential for object recognition, shape analysis, medical image processing, cartographic data extraction, industrial inspection, and vision-based navigation systems. The separation of linear and curvilinear structures plays a particularly important role in the extraction of objects such as roads, shorelines, canals, and riverbeds from images [2].

Classical approaches to edge and contour detection in images are primarily based on first-order derivative gradient methods. In this context, the edge detector proposed by Canny is considered one of the most widely used methods. This approach defines precise criteria for edge point extraction and aims to optimize edge detection with respect to detection accuracy, localization, and the reduc-

© Goyushova U.M., 2026
Стаття поширюється на умовах ліцензії відкритого доступу CC BY 4.0

tion of false responses. However, gradient-based operators mainly detect sharp intensity changes and therefore identify the presence of an edge; the local behavior of a curve, its degree of curvature along a direction, and the second-order structural variation of the image are not fully represented by these methods [3].

For more accurate detection of curvilinear structures, methods based on second-order derivatives occupy an important place in the literature. In this regard, the Hessian matrix provides richer information about local curvature, directional variation, and extremum properties. One of the classical studies devoted to the extraction of curvilinear structures is the work of Steger, which demonstrated that the detection of such structures is one of the basic low-level operations in computer vision and has numerous applications [4]. At the same time, Frangi and co-authors proposed an effective approach for enhancing vessel-like structures through multiscale analysis based on the eigenvalues of the Hessian matrix, confirming the practical value of second-order local structural analysis [5].

Contour points extracted from an image are often scattered and unstable because of noise, illumination changes, pixel discretization, and edge detection errors. For this reason, it is more appropriate to represent the shape of an object not by a rigid curve passing through all points, but by a smoother and more stable parametric model. In this context, spline methods, especially B-splines, are regarded as efficient tools both theoretically and practically. De Boor systematized spline theory from a computational point of view and justified the representation of splines as weighted sums of B-splines [6]. Dierckx, in turn, described in detail the algorithmic and mathematical foundations of curve and surface fitting with splines [7]. These properties explain the advantages of B-spline approximation in terms of local controllability, smoothness, and compact parametric representation [8].

Task statement. Thus, although classical edge detection methods can be employed at the initial stage for the reliable extraction of curves from images, approaches based on second-order derivatives, and especially on Hessian analysis, are more suitable for a fuller description of structural characteristics. Constructing a B-spline approximation from the selected informative points makes it possible to represent the object shape in a more compact, smooth, and comparison-ready form. Therefore, the detection of curves, their modeling by splines, and their subsequent comparison based on structural parameters can be regarded as a scientifically grounded approach for the identification of visual objects.

Outline of the main material of the study

1.1. Method for detecting curves from images

Curve detection in images is carried out after the preprocessing stage by means of structural analysis based on second-order derivatives. During preprocessing, the image is denoised, converted into grayscale, binarized when necessary, and used to extract edge information [9]. The purpose of this stage is to isolate the informative parts of the image and to form more stable input data for the subsequent analysis.

In order to detect curves more accurately, the local geometric properties of the image are analyzed by means of the Hessian matrix [10]. For the image intensity function $I(x,y)$, the Hessian matrix at the point (x,y) is defined as follows:

$$H(x,y)=\begin{bmatrix} I_{xx}(x,y) & I_{xy}(x,y) \\ I_{yx}(x,y) & I_{yy}(x,y) \end{bmatrix},$$

where I_{xx} and I_{yy} denote the second-order derivatives along the x and y directions, respectively, while $I_{xy}=I_{yx}$ denotes the mixed derivative.

The presence or absence of local curvature at a given point is determined on the basis of the eigenvalues of the Hessian matrix. The eigenvalues are found as the roots of the following characteristic equation:

$$\det(H-\lambda\mathbf{I})=0,$$

where H is the Hessian matrix, λ is the eigenvalue, and $\mathbf{I}$ is the identity matrix.

By expanding this expression, the following quadratic equation is obtained:

$$\lambda^2-(I_{xx}-I_{yy})\lambda+(I_{xx}I_{yy}-I_{xy}^2)=0.$$

Accordingly, the eigenvalues are computed as

$$\Delta=(I_{xx}+I_{yy})^2-4(I_{xx}I_{yy}-I_{xy}^2)$$

$$\lambda_{1,2}=\frac{(I_{xx}+I_{yy})\pm\sqrt{\Delta}}{2}$$

The local geometric character of the point is determined from the signs and relative magnitudes of the eigenvalues. In order for curvature to be considered present at the point (x,y), one of the eigenvalues must be significantly larger than the other, that is, the following condition must hold:

$$|\lambda_1|\gg|\lambda_2|.$$

This condition indicates the presence of dominant curvature in one direction and implies that the point lies on a curve.

In addition, the points are classified according to the signs of the eigenvalues as follows:

- If $\lambda_1 > 0, \quad \lambda_2 > 0$, the point is a local minimum;
- If $\lambda_1 < 0, \quad \lambda_2 < 0$, the point is a local maximum;
- If $\lambda_1 \cdot \lambda_2 < 0$, the point is a saddle point;
- If $\lambda_1 \approx 0, \quad \lambda_2 \approx 0$, the point belongs to a flat region with no curvature.

This classification serves two main purposes: it removes pixels belonging to flat background regions and it identifies the probable direction of the curve together with the zones of strong local variation. Thus, the informative points selected by the Hessian-based analysis are used in the subsequent approximation stage as control points.

1.2. Approximation of splines from images

The contour or road points extracted from an image may be inaccurate and scattered because of illumination changes, noise, shadows, texture, and edge-detection errors under real conditions. Therefore, it is more appropriate to represent the object shape by a smoothed parametric model rather than by a rigid curve passing through all detected points. For this purpose, B-spline approximation is employed.

The general form of a B-spline curve is written as a linear combination of control points and basis functions:

$$Q(u) = \sum_{i=0}^{n} P_i N_{i,k}(u),$$

where P_i are the control points, $N_{i,k}(u)$ is the B-spline basis function of order k, and u is the parameter.

$$N_{i,k}(u) = \frac{u - t_i}{t_{i+k} - t_i} \cdot N_{i,k-1}(u) + \frac{t_{i+k+1} - u}{t_{i+k+1} - t_{i+1}} \cdot N_{i+1,k-1}(u),$$

$$t_i = 0, 1, .., n+k \, .$$

This representation demonstrates the main advantage of the B-spline method: the basis functions are local over the parameter interval. Consequently, each control point affects only a specific portion of the curve. Therefore, when a local error or discontinuity occurs in one part of the contour, the entire curve is not distorted globally.

The accuracy of B-spline approximation depends directly on the correct selection of control points. If the control points do not adequately reflect the main structure of the contour, the resulting curve may deviate from the actual object shape. For this reason, the control points are formed from the informative points selected by the Hessian matrix. Since points characterized by local curvature and extremum-like behavior represent the main structural properties of the contour, using such points as control points improves both the stability of the approximation and the reliability of the comparison stage.

1.3. Comparison of images by comparing curves

During video-based navigation, the current position of the platform is determined by comparing the observed objects with previously known references [11, 12]. For this purpose, curve-like objects extracted from reference images are modeled by B-splines and their characteristic parameters are stored. The same processing sequence is then applied to the currently observed image in order to construct the corresponding spline. As a result, the problem is formulated not as a comparison of raw images, but as a comparison of spline curves.

In the comparison stage, the control points of the curve are used as the principal features. If r_j denotes the selected features of the observed curve and $r_p(t_j')$ denotes the features of the reference curve, then, in the simplest case, the similarity criterion can be written as

$$d_p^2 = \sum_{j=1}^{M} \left\| r_j - r_p(t_j') \right\|^2 .$$

The reference curve with the minimum difference is accepted as the matching object.

This corresponds to the simplest comparison case, in which scaling, rotation, and translation are not taken into account.

In the more general case, R rotation, $b = (b_x, b_y)$ translation, and β scaling are applied to the reference points. If the transformed reference points are written as $r_{p,T}(t_j')$, then the correspondence is determined by minimizing the following residual:

$$d_p^2 = \sum_{j=1}^{M} \left\| r_j - r_{p,T}(t_j') \right\|^2 .$$

Thus, the comparison process is brought into a form that is structurally compact, mathematically grounded, and applicable to navigation tasks.

1.4. Practical implementation and discussion

The practical validation of the proposed approach was carried out in Python using the functions provided by the OpenCV library [13]. As the initial experimental data in Figure 1, a ground surface image obtained from Google Earth was selected, and a sequence of processing stages was applied to this image. The processing pipeline consisted of image preparation, conversion to grayscale, binarization, edge detection, contour extraction, and, at the final stage, spline-based curve approximation.

At the initial stage, the color image was converted into grayscale. After converting grayscale result shows in Figure 2.

In addition to reducing computational complexity, this conversion made it possible to analyze intensity changes more clearly during the subsequent edge-detection stage. The image was then transformed into a black-and-white binary form. As a result of binarization, the distinction between the object and the background became more explicit, which provided a more suitable representation for contour and boundary extraction.

Fig. 1. A ground surface image obtained from Google Earth

Fig. 2. Grayscale result of Earth surface image

At the next stage, the Canny algorithm was applied to detect edges within the image. The obtained result showed that the main boundary lines of the object became more clearly separated from the background in Figure 3. This stage plays an important role in forming the initial contour representation of curve-like structures. The line structures obtained after edge detection are used as the input data for the subsequent contour extraction step.

Contour extraction was performed using the findContours() function of the OpenCV library. As a result of this operation, continuous curve lines describing the objects in the image were identified. Each detected contour is represented as a sequence of points located along the boundary. This makes it possible to obtain a direct discrete geometric description of the object shape. Such a representation forms the primary data basis for the subsequent spline approximation stage.

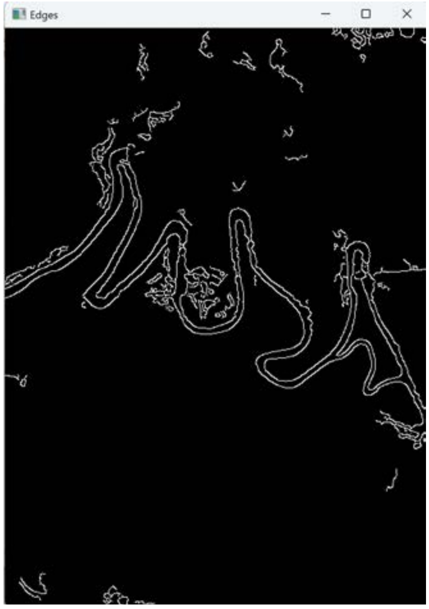

Fig. 3. The result obtained in binary format by applying the Canny algorithm

To construct a smooth curve from the extracted contour, the splprep() and splev() functions of the SciPy library were used. At this stage, a smoothed parametric representation of the curve was obtained on the basis of the contour points. As a result of B-spline approximation, the object boundary was represented not only as a discrete sequence of points, but also as a more stable and structurally compact curve model. The obtained spline representation reduced local irregularities in the contour and revealed the main trajectory more clearly in Figure 4.

The practical results show that the correct sequential application of the preprocessing stages directly affects the quality of the subsequent curve detection and approximation stages. Conversion to grayscale and binarization simplify edge detection, the Canny operator separates the main boundary lines, the findContours() function represents these boundaries as discrete contours, and the splprep() and splev() functions replace these contours with a smooth curve model. Thus, the processing chain ensures the gradual transformation of the raw image into a structurally more informative representation.

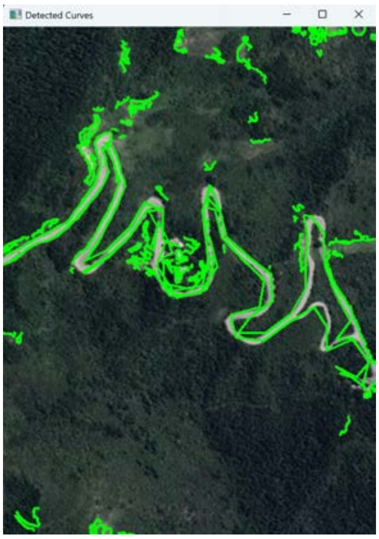

Fig. 4. A representation of the approximated curve in the image via OPENCV

From the standpoint of discussion, the obtained results demonstrate that smoothing discrete contours by means of splines makes the object shape more suitable for comparison. Compared with direct pixel-level or raw contour-point comparison, the spline representation provides a more compact parametric model and therefore creates a basis for simplifying computations in the subsequent identification stage. This can be regarded as an important advantage, especially in vision-based navigation problems where curve-like objects are compared with reference patterns.

At the same time, since the quality of the contour depends on the outcome of the initial preprocessing, the construction of the final curve is also determined by the accuracy of these stages. If the boundary of the object is extracted incompletely or is distorted by noise during edge detection, the sequence of contour points will also become unstable, and this will affect the quality of spline approximation. For this reason, preprocessing, edge detection, and contour extraction should be considered the main stages that determine the reliability of the resulting spline representation.

Thus, the program implementation demonstrates that the extraction of curve-like structures from ground surface images and their approximation by a smooth curve can be practically realized using OpenCV and SciPy tools. The obtained results indicate that the method is structurally workable and creates an appropriate mathematical basis for subsequent curve-based comparison and identification stages.

Conclusions. This study presents an approach for the identification of visual objects based on the detection of curvilinear structures in images and their parametric representation. The main advantage of the proposed method is that objects are described not at the direct pixel level, but at the level of structural features. This makes it possible to reduce the influence of scale, position, and viewpoint differences during comparison, that is, to achieve a more invariant representation of the image.

A key aspect of the approach is the selection of control points for B-spline approximation using the Hessian matrix. The use of Hessian-based analysis makes it possible to take local curvature properties into account and to identify the most informative points of the contour. As a result, the B-spline model represents the main shape of the object in a more accurate, smooth, and stable manner.

Thus, the determination of informative points by means of the Hessian matrix, together with the B-spline representation constructed from these points, provides an appropriate mathematical basis for more reliable comparison and identification of curvilinear objects.

Bibliography

1. Yue-hua S., Yuan-Ii C. Image Feature Extraction for Vision-Based UAV Navigation. In 2018 Chinese Automation Congress (CAC), Xi'an, China, -2018, pp.1130-1134, Doi: 10.1109/CAC.2018.8623264.

2 Parker J.R. Alghorithms for Image Processing and Computer Vision, second edition, Wiley Publishing,Inc, pp.21-251.

3 Canny J. A Computational Approach to Edge Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(6), 679-698, 1986.

4 Steger C. An Unbiased Detector of Curvilinear Structures. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(2), 113-125, 1998.

5 Frangi A. F., Niessen W. J., Vincken K. L. et al. Multiscale Vessel Enhancement Filtering. In: Wells, W. M., Colchester, A., Delp, S. (eds.), MICCAI '98, Lecture Notes in Computer Science, vol. 1496, pp. 130-137, Springer, 1998.

6 de Boor, C. A Practical Guide to Splines. Springer, revised edition, 2001.

7 Dierckx P. Curve and Surface Fitting with Splines. Oxford University Press / Clarendon Press, 1993.

8 Van, T.D., Tegoeh, T. A direct method to solve optimal knots of B-spline curves: An application for non-uniform B-spline curves fitting. Singapore: PLOS one, - 2017. Vol.12, № 3, -p.1-24.

9 Reinhard K. Concise Computer Vision, An Introduction into Theory and Algorithms. New Delhi: Springer-Verlag London, first edition. 2014. 429 p.

10 Kong Y., Liu L., Chen H., Kacprzyk J., Tao D. Overcoming Catastrophic Forgetting in Continual Learning by Exploring Eigenvalues of Hessian Matrix. in IEEE Transactions on Neural Networks and Learning Systems, vol. 35, no. 11, pp. 16196-16210, Nov. 2024, doi: 10.1109/TNNLS.2023.3292359.

11 Sabziev E.N. Determining the location of an unmanned arial vehicle based on video camera images. Baku: Advanced Information Systems, 2021. vol.5. No1. p.136-139.

12 Goyushova U. Determining the location of a moving device based on video images. *Informatics and Control Problems.* 2024. 44(2). 51-57. Doi: 10.54381/icp.2024.2.06

13 Kaehler A., Bradski G. Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library. O'Reilly Media. 2017.

Гоюшова У.М. СТРУКТУРНО-ОРІЄНТОВАНА НАВІГАЦІЯ БПЛА НА ОСНОВІ КОМП'ЮТЕРНОГО ЗОРУ ІЗ ВИКОРИСТАННЯМ МАТРИЦІ ГЕССЕ ТА B-СПЛАЙН-МОДЕЛЮВАННЯ

У статті представлено структурний підхід до навігації безпілотних літальних апаратів (БПЛА) на основі комп'ютерного зору, який ґрунтується на виявленні, параметричному моделюванні та порівнянні криволінійних об'єктів, виділених з аерофотознімків. На відміну від традиційних методів зіставлення зображень, що безпосередньо використовують інтенсивність пікселів або первинний візуальний вигляд, запропонований метод представляє об'єкти через їхні геометричні та структурні характеристики. Такий підхід зменшує вплив змін масштабу, зсуву, часткових змін точки спостереження та інших спотворень, які часто знижують надійність прямого порівняння зображень у реальних умовах навігації. У результаті метод є більш стійким і краще адаптованим до роботи БПЛА в середовищах, де зовнішній вигляд об'єктів може суттєво змінюватися.

Центральним елементом запропонованого підходу є вибір контрольних точок для апроксимації B-сплайном із використанням матриці Гессе. Аналіз на основі матриці Гессе дає змогу визначити найбільш інформативні точки контуру шляхом оцінювання локальної структури другого порядку зображення та виділення ділянок зі значними змінами кривини. Це забезпечує вилучення ключових структурних елементів криволінійних об'єктів при одночасному пригніченні надлишкових або менш інформативних деталей. Відібрані точки використовуються для побудови B-сплайн-моделі, яка забезпечує гладке, компактне та стабільне параметричне представлення форми об'єкта.

Поєднання вибору точок на основі матриці Гессе та B-сплайн-моделювання формує математично обґрунтовану основу для надійного опису, порівняння та ідентифікації криволінійних об'єктів. Відмінною рисою запропонованого методу є те, що порівняння виконується не між вихідними зображеннями, а між сплайновими кривими та їх характерними параметрами. Це суттєво зменшує обчислювальну складність і підвищує інтерпретованість результатів. Завдяки своїй структурній природі метод особливо придатний для задач навігації БПЛА, де потрібне легке, стійке та інваріантне представлення об'єктів для зіставлення поточних спостережень із раніше збереженими еталонними даними та для забезпечення точної локалізації у візуально складних сценах.

Ключові слова: *БПЛА, навігація на основі комп'ютерного зору, інваріантність до точки спостереження, матриця Гессе, B-сплайн, виявлення кривих, OpenCV.*

Дата першого надходження статті до видання: 16.03.2026
Дата прийняття статті до друку після рецензування: 20.04.2026
Дата публікації (оприлюднення) статті: 19.05.2026