© Universiti Tun Hussein Onn Malaysia Publisher's Office

IJIE

http://penerbit.uthm.edu.my/ojs/index.php/ijie
ISSN : 2229-838X e-ISSN : 2600-7916

The International Journal of Integrated Engineering

Identification of Lines Based on Form Factor and Geometrical Descriptors

Maha Abbas Hutaihit[1], Haider Makki Hammed[1*], Oksana Shauchuk[2]

[1]Department of Communication Engineering, Collage of Engineering,
University of Diyala, IRAQ

[2]Belarusian State University of Informatics and Radioelectronics,
Minsk, BELARUS

*Corresponding Author

DOI: https://doi.org/10.30880/ijie.2022.14.06.023
Received 12 December 2021; Accepted 23 June 2022; Available online 10 November 2022

Abstract: The problems of releasing and identifying features of key elements and their identification in images are shown. The basic algorithms and methods for forming descriptors for key elements in the image are considered. The main disadvantages of existing methods in aerospace images are the loss of stability and the presence of a large number of false key elements when changing recording to the conditions (for example, brightness and contrast of the received images). To eliminate these drawbacks, it is proposed to use selected lines as key elements, and their geometrical characteristics to form a descriptor. To analyze the proposed algorithm and the SIFT method, fragments of aerospace images changed in brightness and contrast in a graphic editor from -50 to +50 percent of the original were used, which made it possible to evaluate the operation of these algorithms in conditions close to the real survey. It is shown that the proposed algorithm is more stable than the SIFT method with increasing contrast and decreasing the brightness of the test aerospace image. Also, the proposed algorithm is characterized by a smaller number of detected false key elements compared to the SIFT method.

Keywords: Aerospace images, local orientations, contour, line fragments

1. Introduction

In Earth Remote Sensing Systems (ERSS), the key tasks are stitching the image to produce a photograph of the area, as well as parameterization and identification of objects in the received images. To solve the tasks, it is necessary to detect and describe key objects of images according to various criteria. Key objects must have the following properties [1]:

- Uniqueness concerning a certain neighbourhood and the image as a whole.
- Invariance for affine transformations.
- Noise stability.
- Efficiency in parameterization and identification of objects in the total of analysed images.

The key objects in the image can be a point (pixel), line, or segment of the image. Most identification and parameterization methods use points as key objects. For detection key points in the image different methods are used, such as detectors Moravec [2], Harris [2-3], FAST [4], MSER [5], methods LSD [6] and Hough [7]. The characteristics of detected key objects for the described tasks are described using descriptors.

In general, descriptor – this is a set of different signs, uniquely identifying a key object. Some descriptors, such as SIFT [8], SURF [9] and ORB [10], can also search for special points, and the formation of their signs.

*Corresponding author: haidermakki300@yahoo.com

The most common descriptors are based on the calculation of the brightness gradient of a point and its surroundings, allowing it to identify areas, edges, and angles. These methods are SIFT, SURF, BRIEF [11], ORB, CLON [12-13], DAISY [14], and others. The selection of the descriptor is carried out under the task and the type of image. When choosing a descriptor for solving the above tasks of processing aerial images [15-16], the following conditions must be taken into account:

- Sharp change of illumination of the subject (images are formed at different times of the day, for example, day and night, as well as at different times of the year, for example, winter, summer);
- Similarity, subject (sea, desert, mountains);
- Use for image formation on board equipment ERSS with different characteristics.

In these conditions, the use of descriptors, based on the calculation of the brightness gradient is impossible, therefore, there is a need to use dedicated lines as key objects, but as descriptors – methods, based on the calculation of the geometrical parameters of the detected lines of such images.

2. Research Method

2.1 SIFT

Method SIFT (Scale Invariant Feature Transform) is a method for detecting and calculating descriptors of key points. As a detection method, SIFT uses the most famous method, based on the use of the Gauss pyramid, built for the image, and further calculation of the difference of its Gaussians. Key points, detected using the method SIFT, are the local extrema of the difference of the Gaussians. Next is the refinement of the key point, using extreme polynomial calculations Taylor and matrix Hessian. To find the descriptor of a given key point in this method, it is necessary to calculate its direction. The direction is calculated based on the directions of the brightness gradients of the points, located in a given neighborhood, according to the formulas:

$$m(x,y)=\sqrt{(L(x+1,y)-L(x-1,y)+L(x,y+1)-L(x,y-1))^2}, \tag{1}$$

$$\theta(x,y)=\tan^{-1}\frac{(L(x,y+1)-L(x,y-1))}{(L(x+1,y)-L(x-1,y))} \tag{2}$$

where $m(x,y)$ - gradient value, $\theta(x,y)$ - direction. Based on the calculation of the brightness gradients of each point in the neighborhood of the key, a weighted histogram of gradients is constructed. In a histogram of 36 components, which are uniformly coated with an interval of 360 degrees, the maximum value of the component is selected m (max). The key point is assigned all directions, greater than or equal to 0.8 *m (max) (80 percent). Usually uses a neighborhood radius of 5 pixels.

The neighborhood is usually used to calculate the main descriptor 16x16 pixels. This neighborhood is divided into 4 squares with size 4x4 pixel, for each of which a local histogram of gradients is constructed, using the same algorithm, as in calculating the direction of the key point. However, local histograms do not contain 36, but 8 components, evenly distributed over the interval in 360 degrees as shown in Fig 1.

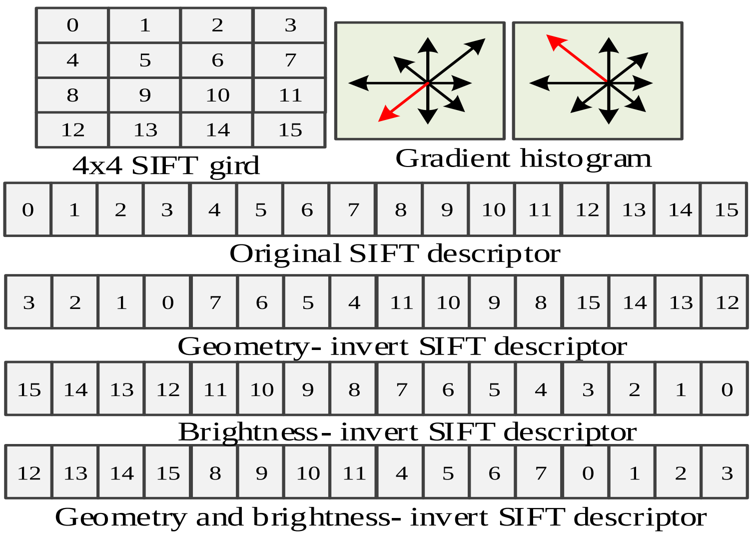

Fig. 1 - SIFT descriptor

As a result, a descriptor is generated for each key point, consisting of the direction vector of the key point and the vector of 128 components ($4 \times 4 \times 8$). Most often it is a vector from 128 components called a keypoint descriptor. Descriptor SIFT scale-invariant, moving an image of an object on a scene, rotation of an object, or camera.

The disadvantage of this method is a high computational complexity [13], some keypoints, and their descriptors are deleted as a result of the filtering. Also, worth noting, as described above, SIFT based on the calculation of the brightness gradient of the keypoint, which leads to the false points and ineffectiveness of this method for solving the described problems for aerospace image ERSS.

2.2 SURF

Method SURF, just like SIFT, is a method for detecting keypoints and constructing their descriptors. SURF detects keypoints using a matrix Hessian, determinants which reaches its maximum at the point of maximum change in brightness gradient. Since the Hessian is not scale-invariant, SURF uses multiscale filters to find it. Also, for each point, the direction of the brightness gradient is calculated, using filter Haar, and scale, equal to the scale factor of the matrix Hessian. Descriptor SURF it represents the vector 64 or 128 numbers for each keypoint. This descriptor shows the gradient fluctuations around the keypoints and is calculated using the filter Haar for a given neighborhood. The size of the neighborhood for calculation is determined by the scale of the matrix Hessian.

Method SURF not invariant to affine transformations and not stable when brightness changes, however applicable when rotating and zooming.

The disadvantage of this method, that it does not work with objects, in which it is easy to identify the keypoints, but considers the image as a whole, which affects the detection of keypoints in images with a fractal structure or a change of season.

2.3 BRIEF

BRIEF (Binary Robust Independent Elementary Features) – one of the fastest and easiest detectors that require little storage for descriptors in [11] is shown, that its computational complexity is only 256 operations. This descriptor is a binary vector, consisting of 0 and 1, before the calculation of which the smoothing operation is performed.

Each descriptor value is calculated by comparing the brightness of the pixel pairs, located in the neighborhood $s \times s$ keypoints p, using the following function:

$$. \ \tau(p, x, y) := \begin{cases} 1, \text{if } I(p, x) < I(p, y) \\ 0, \text{other case} \end{cases} \qquad (1)$$

where $I(p,x), I(p,y)$ – smooth pixel intensity p.

Comparison BRIEF descriptors did by simple distance calculation Hamming. In [11] shown, that under the same experimental conditions on some test images, the accuracy of detection using BRIEF almost 1.5 times higher, then using SURF- descriptors. However, descriptor BRIEF is not invariant to rotation, it is also not effective for fractal images and is not stable to a sharp change in the lighting of an object.

2.4 ORB

ORB (Oriented FAST and Rotated BRIEF) – method, a combination of a keypoints detector FAST and binary descriptors BRIEF, accordingly, this method is an improved alternative to a descriptor BRIEF [10].

2.5 GLOH

GLOH (Gradient location-orientation histogram) – modified descriptor SIFT, designed to increase the stability and accuracy of keypoints descriptions. The technology for calculating this descriptor is identical to the descriptor SIFT, a distinctive feature is the use of a polar grid to break the neighborhood of a keypoint. The neighborhood is divided as follows: 3 radial block with radii 6, 11, and 15 pixels are allocated, each of which are divided into 8 sectors. The result is a 272 bin vector, which carries over 128- dimensional space, using principal component analysis (PCA) [13].

2.6 DAISY

This descriptor has similar characteristics with the most common descriptors SIFT and GLOH but shows a higher speed [14]. SIFT and GLON descriptors are used DAISY. As in GLOH, this descriptor uses a circular neighborhood of the keypoints, however, full circles are used instead of sectors. For each neighborhood, a sequence of processes is performed, similar to descriptor SIFT, however, instead of a weighted histogram of the gradients, the convolution of the original image with the derivatives of the Gaussian filter is used, taken in 8 directions.

This descriptor is designed to work on a dense array of pixels, those when many pixels are keyed in these conditions DAISY has the same properties, as SIFT with GLON, but 66 times faster than the SIFT descriptor [17].

3. Mathematical Model and Proposed Algorithm

The proposed algorithm of descriptor formation based on the construction of a histogram of local orientations of fragments of the selected line, or HLOLF (Histogram of the local orientation of line fragments), uses contours or center lines of objects as key elements [18-21] as shown in Fig 2. The processing line has two endpoints, has no sharp angles, and should be pre-normalized in thickness. For each selected line, the algorithm forms descriptors by constructing histograms of local orientations of their fragments [21]. An image is an input to $I = \left\| i(y, x) \right\|_{\left(y=\overline{0,Y\text{-}1},\ x=\overline{0,X\text{-}1}\right)}$, where $i(y, x) = 0....255$ – brightness value of the pixel in the image, (Y, X) – image dimensions vertical and horizontal.

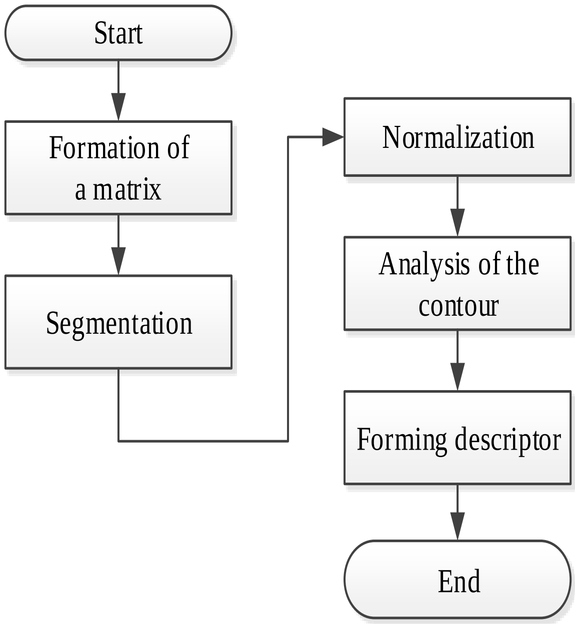

Fig. 2 - HLOLF algorithm

The algorithm is consisting of the following steps:

Step 1. Formation of a matrix $I_B = \left\| i_B(y, x) \right\|_{\left(y=\overline{0,Y\text{-}1},\ x=\overline{0,X\text{-}1}\right)}$ for input image I using the contour Canny filter, where $i_B(y, x) = 1$ for pixels, belongs to contour, $i_B(y, x) = 0$ for background pixel.

Step 2. Segmentation of contour lines. Each contour pixel $i_B(y, x) = 1$ the contour number is assigned, to which it belongs. As a result, a contoured matrix is formed $S = \left\| s(n) \right\|_{n=\overline{1,N}}$ and matrix of the number of endpoints in each contour $K = \left\| k(n) \right\|_{n=\overline{0,N}}$, where $s(n)$ – contour pixel coordinates n contour, presented in the form of matrices $X(n) = \left\| x(n,c) \right\|_{\left(c=\overline{0,C\text{-}1}\right)}$, $Y(n) = \left\| y(n,c) \right\|_{\left(c=\overline{0,C\text{-}1}\right)}$, $k(n)$ – number of endpoints for n contour, N – number of contours found, C – number of pixels in n contour.

Step 3. Normalization of selected contours by thickness. Normalization of contours by using the method of normalizing contour lines in thickness [22]. In the process of normalization from the contour coordinate matrices $X(n)$, $Y(n)$ pixels are deleted, which visually and physically make the line thicker. As a result, contours are formed $s(n)$ one pixel thick.

Step 4. Analysis of the contour. First, the analysis of the endpoints of the contours $k(n)$ if $k(n) = 2$ – a decision is made, that contour $s(n)$ is a line and the Form Factor is calculated f [23, 24].

Step 5. Formation of the descriptor. And this step is consisting of the following steps:

Step 5.1. Calculation of local orientations of each pixel of a line, except end pixel.

The local orientation of each pixel in the line, excluding endpoints, determined by matching the neighborhood of a line point with a size 3×3 pixel of one of 12 masks as shown in Fig 3.

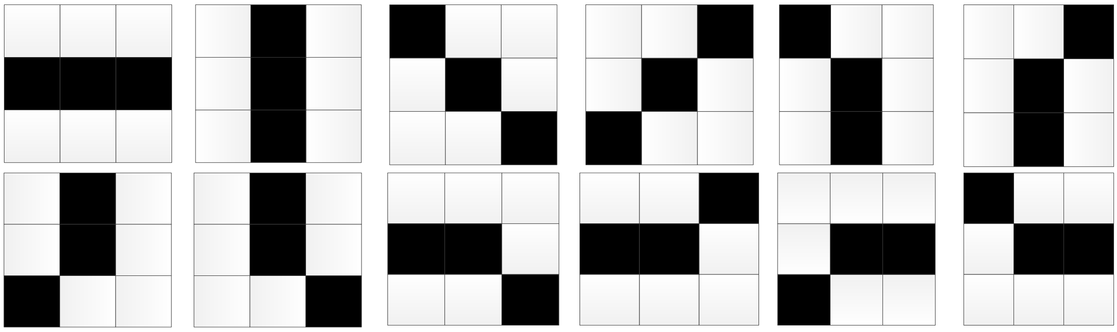

Fig. 3 - Masks for determining local pixel orientation: 1-12 respectively

Step 5.2. Calculating the local orientation of the endpoints. To determine the local orientation of the endpoints, the orientation of the selected line is calculated using the following function:

$$O_L = \frac{y_2 - y_1}{x_2 - x_1} \tag{5}$$

where (y_1, x_1) *and* (y_2, x_2) – endpoint coordinates of the line.

The local orientation of the endpoints to one of the 12 histogram values is set in accordance with the calculated line orientation, based on the following condition:

$$\begin{aligned} &\left|O_L\right| < 0{,}5 \rightarrow mask\ 1, \\ &\left|O_L\right| > 2 \rightarrow mask\ 2, \\ &0{,}5 \le O_L \le 2 \rightarrow mask\ 3, \\ &-2 \le O_L \le -0{,}5 \rightarrow mask\ 4. \end{aligned}, \tag{6}$$

Step 5.3. The descriptor histogram is formed by counting the number of matches for each of the masks and consists of 12 elements, respectively.

Step 6. End of the algorithm.

As a result of the algorithm, a descriptor is formed for each selected line, consisting of a vector, which is a histogram of the possible local orientations of each key line pixel, 12 elements long.

4. Line Identification Based On Form Factor and Geometric Descriptor

Line identification through two steps:

Step 1. Estimation of the difference in the value of the Form Factor line Δf, using the following condition:

$$\Delta f \ge 0.1 \tag{7}$$

where $\Delta f = |f_1 - f_2|$ – Form Factor difference, f_1 and f_2 – the values of the form factors of the selected lines of the first and second images, respectively, calculated at the stage of detection.

Step 2. If the deviation satisfies the given condition Δf (the lines are similar), then the histograms of local orientations are evaluated as shown in Fig 4. If these histograms coincide, the lines are considered appropriate.

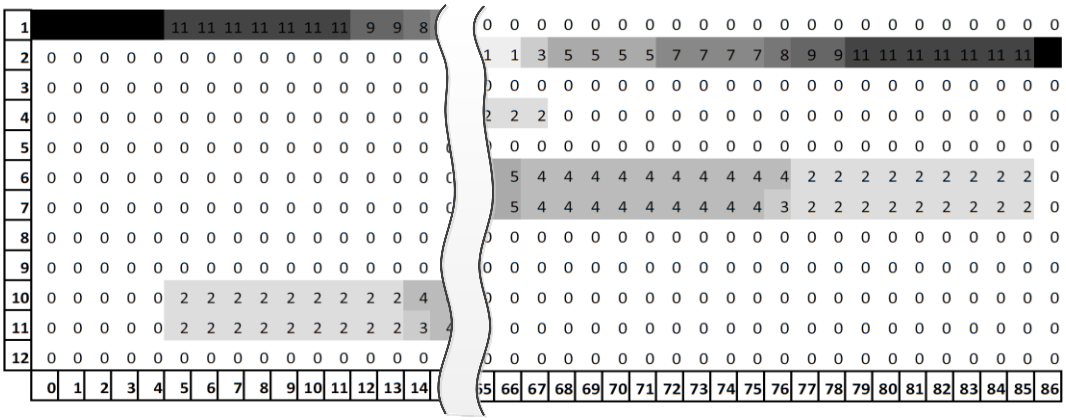

Fig. 4 - Characteristics of changing the histogram of local line orientations with a length of 15 pixels

5. Evaluation of The Effectiveness of the Geometrical Line Descriptor and Their Identification

The proposed algorithm is written in C++. The experiment was tested by the computer with the following technical characteristics: processor Intel(R) Core(TM) i5-2320 CPU @ 3,0 GHz; RAM – 4 GB; Windows 7. For analysis, an aerospace image of size 720×720 pixels as shown in Fig. 5 images were changed in brightness as shown in Fig5, and contrast as shown in Fig 6, using a graphical editor.

Evaluation of the stability of detection of key elements (HLOLF – lines, SIFT – points) calculated, using the following expression:

$$S = \frac{k_i}{k_0} \times 100\% , \tag{8}$$

where k_0 – the number of selected key elements in the original image; k_i – number of selected key elements in the modified image; $i = \overline{-50...50}$ – contrast or brightness value relative to the original test image.

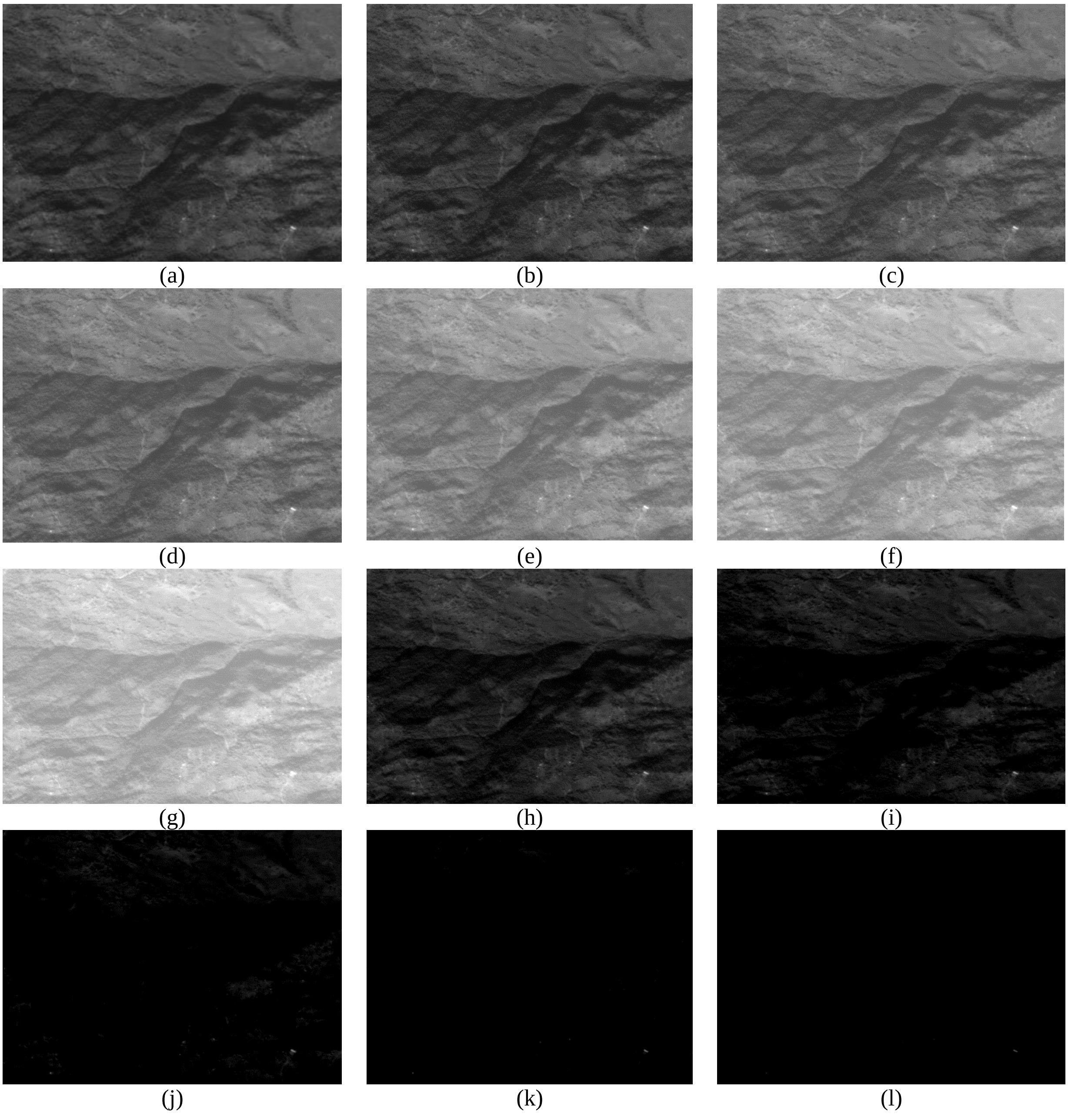

Fig. 5 - Changing in brightness, (a) original image; (b) 0; (c) 10; (d) 20; (e) 30; (f) 40; (g) 50; (h) -10; (i) -20; (j) -30; (k) -40; (l) -50

For evaluation of the stability of line identification is calculated using the expression:

$$S_I = \frac{d_i}{k_i} \times 100\,\%, \qquad (9)$$

where d_i – the number of identified lines in the modified image relative to the original image; k_i – number of selected lines in the modified image; $i = \overline{-50...50}$ – contrast or brightness value relative to the original test image. Evaluation of the stability of detection of key elements by HLOLF and SIFT method in the image when changing brightness and contrast as shown in Fig 6.

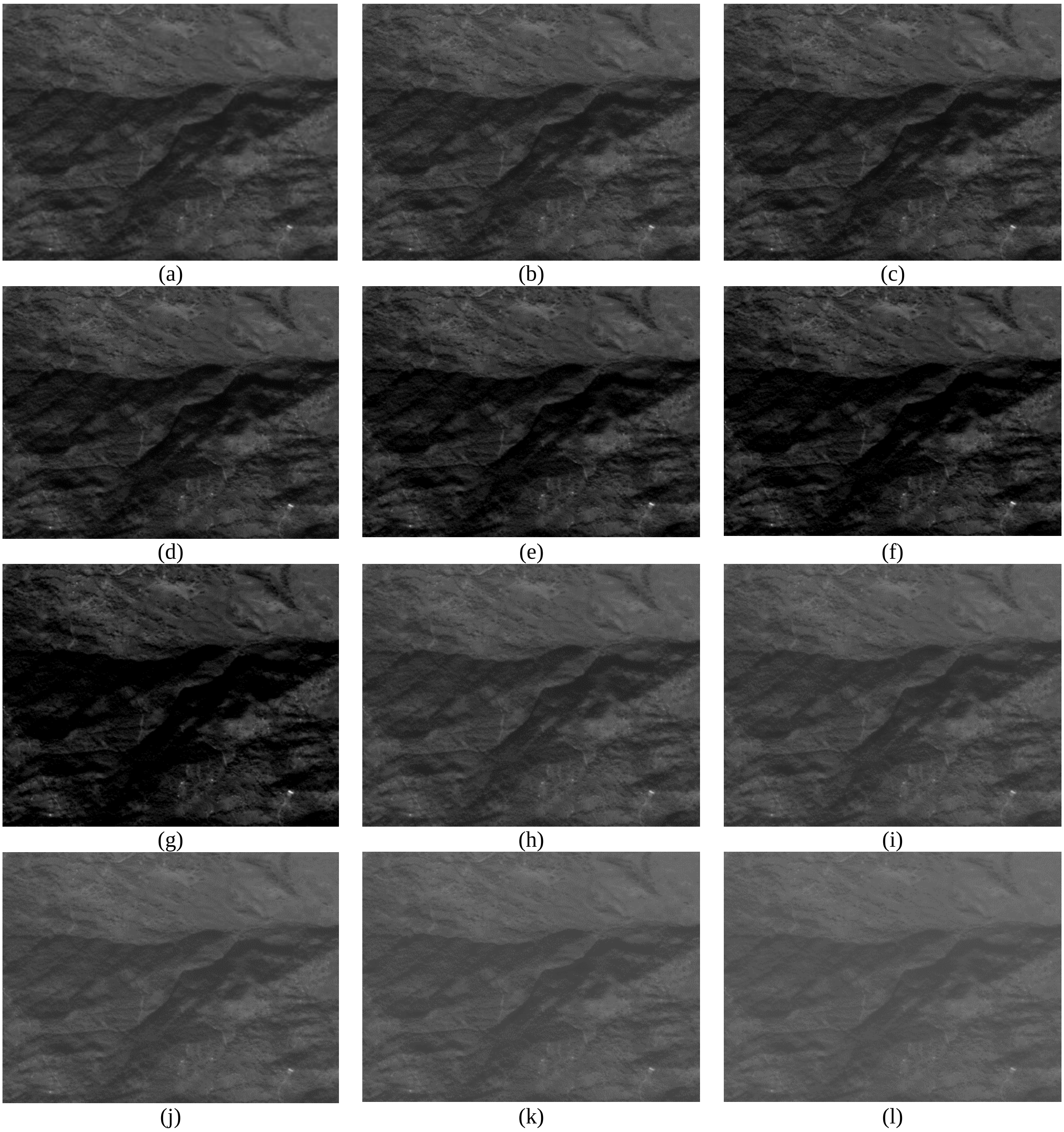

Fig. 6 - Changing in contrast, (a) original image; (b) 0; (c) 10; (d) 20; (e) 30; (f) 40; (g) 50; (h) -10; (i) -20; (j) -30; (k) -40; (l) -50

From Fig. 7 it can be seen that HLOLF algorithm is more stable than the SIFT method with increasing contrast and decreasing brightness.

The average value of the stability of the detection of key elements for the HLOLF algorithm is 90% with a change in contrast and 81% with a change in brightness, for the SIFT method - 120% and 89%, respectively. A value greater than 100% for the SIFT method when contrast is changed, indicates the presence of a large number of false points, a small number of which are also observed for the HLOLF algorithm with increasing contrast.

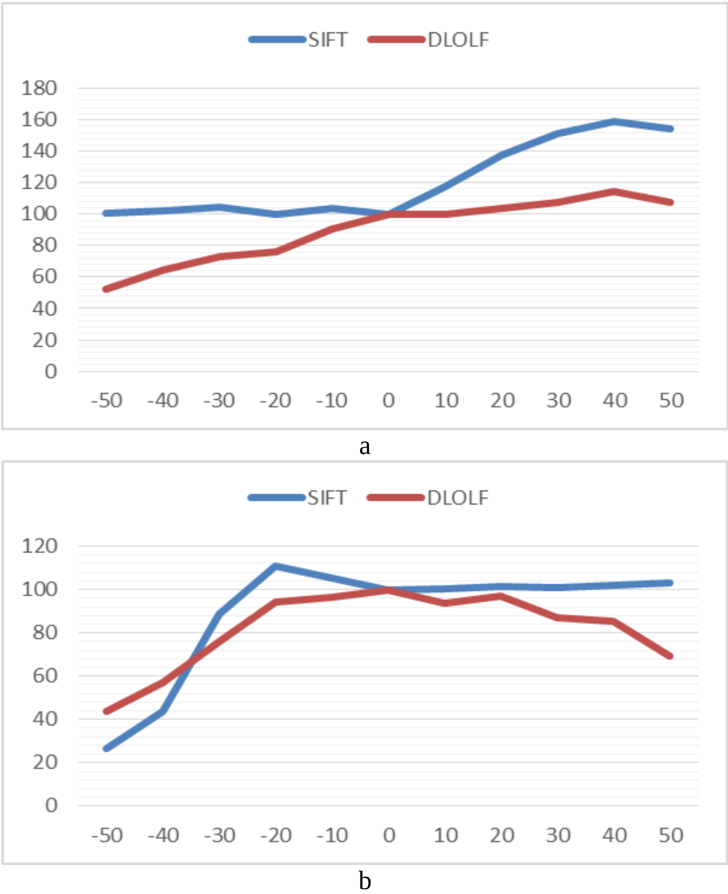

Fig. 7 - Graph stability detection of key elements when: a) chaining the contrast; b) changing the brightness

In Fig. 8 represented the stability Evaluation of the identification line of the original test image on the modified. From Figure 8. It is seen that the proposed algorithm for identifying lines is more stable when the brightness is changed than when changing the image contrast. Also worth noting, that when changing the contrast from −50 to −20 value of the stability S_I exceed 100 % – this is due to, that the same line on the modified image can correspond to several lines on the original. Such a detection error is associated with a small line size (to 11 pixels) and the coincidence of the geometric characteristics of different lines. Such lines need a closer look: lowering the threshold form factor, additional comparison of line orientations, lowering the descriptor difference threshold.

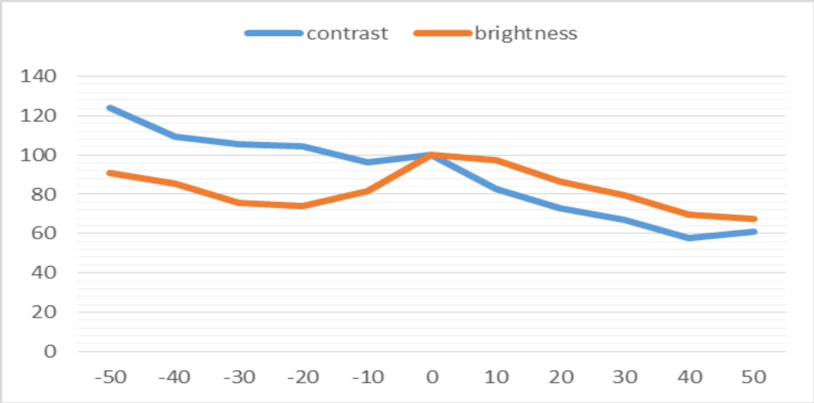

Fig. 8 - Line identification stability

6. Conclusion

An algorithm for the formation of a descriptor based on a histogram of the local orientations of the fragments of the selected line. Produced the stability of the detection of key elements by HLOLF and SIFT method when changing the brightness and contrast of the image. It is shown that the proposed algorithm detects a few number of false points, compare with the SIFT method, for which their number can reach 50%. Evaluation of the stability of identification of the lines of the original image is showing that with a decrease in contrast, the identification error increases.

Acknowledgment

The authors fully acknowledged University of Diyala and Belarusian State University of Informatics and Radioelectronics, for supporting this work.

References

[1] Rodehorst, V., & Koschan, A. (2006). Comparison and evaluation of feature point detectors, 5th International Symposium Turkish-German Joint Geodetic Days TGJGD'06. Berlin, paper 3906.
[2] Papageorgiou, C., & Poggio, T. (2000). A trainable system for object detection. International Journal of Computer Vision, 38, 15–33.
[3] Harris, C., & Stephens, M. (1988). A Combined Corner and Edge Detector, Proceedings of the Fourth Alvey Vision Conference. UK, 2, 147-151.
[4] Rosten, E., Porter, R., & Drummond, T. (2010). Faster and better: a machine learning approach to corner detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 32, 105-119.
[5] Matas, J., Chum, O., Urba, M., & Pajdla, T. (2004). Robust wide baseline stereo from maximally stable extremal regions. Image and Vision Computing, 22, 761-767.
[6] Grompone von Gioi, R., Jakubowicz J., Morel J. M., Randall G. L.(2010). A Fast Line Segment Detector with a False Detection Control. IEEE Transactions on Pattern. Analysis and Machine Intelligence, 32, 722-732.
[7] Duda, O., & Hart, E. (1972). Use of the Hough Transformation to Detect Lines and Curves in Picture. Communication of the ACM, 15, 229-246.
[8] Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. Intern. J. of Computer Vision, 60, 91-110.
[9] Bay, H., Ess, A., Tuytelaars, T., & Gool, L. (2008). SURF: Speeded up robust features, Computer Vision and Image Understanding, 110, 346–359.
[10] Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2010). ORB: an efficient alternative to SIFT or SURF, IEEE International Conference on Computer Vision, 2564-2571.
[11] Calonder, M., Lepetit, V., Strecha, C., & Fua, P. (2010). BRIEF: Binary Robust Independent Elementary Features," ECCV'10 Proceedings of the 11th European conference on Computer vision, Berlin, 6314, 778-792.
[12] Kalal, Z., Mikolajczyk, K., & Matas, J. (2010). Forward-backward error: automatic detection of tracking failures, International Conference on Pattern Recognition, 1-4.
[13] Jolliffe, I. T. (2002). Principal Component Analysis (2nd ed.). Springer, NY: McGraw-Hill.
[14] Tola, E., Lepetit, V., & Fua, P. (2010). daisy: an efficient dense descriptor applied to wide baseline stereo, IEEE transactions on pattern analysis and machine intelligence, 32, 815-830.
[15] McKeown, D., Denlinger, J. L. (1988). Cooperative methods for road tracking in aerial imagery, The Computer Society Conference on Computer Vision and Pattern Recognition, 662-672.
[16] Liu, G., Sun, X., Fu, K., & Wang, H. (2013). Aircraft recognition in high-resolution satellite images using coarse-to-fine shape prior. IEEE Geoscience and Remote Sensing Letters, 10, 573-577.

[17] Pentland, A., & Choudhury, T. (2000). Face Recognition for Smart Environments. IEEE Computer Vision, 33, 50-55.
[18] Liu, G., Sun, X., Fu, K., & Wang, H. (2013). Interactive geospatial object extraction in high resolution remote sensing images using shape-based global minimization active contour model. Pattern Recognition Letters, 34, 1186–1195.
[19] Ahmadi, S., Zoej, M., Ebadi, H., & Abrishami, H., et al. (2010). Automatic urban building boundary extraction from high resolution aerial images using an innovative model of active contours. International Journal of Applied Earth Observation and Geoinformation, 12, 150–157.
[20] Jing, Y., An, J., & Liu, Z. (2011). A novel edge detection algorithm based on global minimization active contour model for oil slick infrared aerial image. Transactions on Geoscience and Remote Sensing, 49, 2005-2013.
[21] Dalal, N. & Triggs, B. (2005). Histograms of oriented gradients for human detection, IEEE Computer Society Conference on Computer Vision and Pattern Recognition, San Diego, United States, 886-893.
[22] Shauchuk, A., & Tsviatkou, V. (2016). Method of Normalization of the Contour Line in Thickness Based On Binary Masks, Al-Sadeq International Conference on Multidisciplinary in IT and Communication Science and Applications, Baghdad, 44, 1-6.
[23] Baradzina, O., & Tsviatkou, V. (2015) Localization of isolated straight lines on the images using the form-factor. Saint-Petersburg state electrotechnical university LETI, 1, 41-45.
[24] Alzakki, H. M., Tsviatkou, V. Yu. (2015) Contour processing of texture Images, Proceedings of Second Engineering Scientific Conference, University of Diyala, Diyala, Iraq, 2015. Iraq, 453–461.