

# Computer methods of image processing of volcanoes

Igor Urmanov  
 Laboratory ITS  
 CC FEB RAS  
 Khabarovsk, Russia  
 urmanov@ccfebras.ru

Aleksandr Kamaev  
 Laboratory NMMPh  
 CC FEB RAS  
 Khabarovsk, Russia  
 kamaev\_an@mail.ru

Aleksei Sorokin  
 Laboratory ITS  
 CC FEB RAS  
 Khabarovsk, Russia  
 alsor@febras.net

**Abstract**—Video surveillance of volcanoes is often complicated by bad weather conditions. Many images are not informative, since they do not show the observed object, that makes access to high-quality images difficult for experts volcanologists and leads to excessive consumption of disk memory. In this paper, an algorithm for calculating the visibility estimate of volcanoes based on the comparison of parametric edges in images with the reference edges of the observed volcano is considered. The resulting estimate can be used to filter non-informative images.

**Keywords**— edges extraction, parametric edges, volcano, edges matching, video surveillance, visibility analysis

## I. INTRODUCTION

One way to track the state of volcanoes is video surveillance through stationary cameras located at a certain distance from the objects observed. Investigation of the data obtained by the observation system [1] showed that a significant part of the archive is filled with images on which the observed object is not visible due to unfavorable weather conditions. In this regard, the actual task is to assess the visibility of the volcano in order to organize automatic filtering of non-informative images. This will simplify access to surveillance data for experts - volcanologists, as well as reduce the storage volume needed to store the image archive.

The task of determining the visibility of a volcano can be considered in the context of computer vision problems such as determining atmospheric visibility [2,3], detecting changes [4,5] or searching for features typical of the volcano [6]. The inhomogeneity of the atmosphere at large distances, the large variability of the appearance of the volcano as a function of season and lighting, and the varying rate of change in weather conditions make it difficult to use the first two methods. The search for features peculiar to the volcano, on the contrary, makes it possible to assess the visibility of the volcano in a variety of situations. In [6], such features are the edges pixels. Since distance maps are used to compare them, false matching occur in the intersection zones of edges as well as multiple mappings of the same edges point. This can lead to both overestimation and underestimation of visibility. In this paper, we consider an algorithm based on the comparison of parametrically defined edges, while discrete edges are used to align the compared images with each other.

## II. DISCRETE EDGES

Let the images received from the surveillance camera have the dimensions  $w \times h$  pixels, then each pixel of this image has the number  $yw + x$ , where  $x = 0, 1, \dots, w - 1$ ,  $y = 0, 1, \dots, h - 1$  — pixel coordinates on the image. Using the edge detection algorithm, for example [7-9], for the image under investigation, we obtain the set  $G$  consisting of the numbers of the edge points. The difference between the two

sets of edge points  $G_1$  and  $G_2$ , obtained from two images, will be estimated as

$$\mu(G_1, G_2) = \frac{1}{|G_1|} \sum_{n \in G_1} \min \left( \frac{\min_{k \in G_2} d(n,k)}{d_{cut}}, 1.0 \right) \quad (1)$$

where  $d(n, k)$  is the distance between pixels with numbers  $n$  and  $k$ , and  $d_{cut}$  is the maximum distance between pixels that can be matched. The minimum distance in (1) is searched using the distance map [10] to the edge points  $G_2$ .

The edges on images obtained from one camera can be strongly biased relative to each other due to the influence of external factors, which affects the evaluation. To combine the edges, we introduce a set of offset edge points

$$G^\delta = \left\{ n - \delta_y w - \delta_x \mid \begin{array}{l} (n \in G) \wedge (\delta_x \leq n_x < w + \delta_x) \\ \wedge (\delta_y \leq n_y < h + \delta_y) \end{array} \right\}$$

where  $n_x = n \bmod w$ ,  $n_y = \lfloor n/w \rfloor$ ,  $\delta = (\delta_x, \delta_y) \in \mathbb{Z}^2$ . Using a set of offset silhouette points, we find the shift:

$$\delta(G_1, G_2) = \arg \min_{\delta} \mu(G_1^\delta, G_2) \quad (2)$$

To accelerate the calculation of the shift (2), the search is carried out in two stages: the brute-force search found approximate solutions, which are refined in the second stage by the method of gradient descent. Initial images are reduced in  $s = \min(w, h) / 100$  times. For smaller images, the parameter  $d_{cut} \approx 3s$  is selected. The matrix  $\mathbf{M} = (m_{ij})$ ,  $i, j = -\lfloor s^{-1} \delta_{max} \rfloor, -\lfloor s^{-1} \delta_{max} \rfloor + 1, \dots, \lfloor s^{-1} \delta_{max} \rfloor$  is calculated, where  $\delta_{max}$  is the maximum possible shift between two images in pixels, and

$$m_{ij} = \mu(G_{S_1}^{(i,j)}, G_{S_2})$$

where  $G_{S_1}, G_{S_2}$  — is the set of edge points on images reduced in  $s$  times. The initial values of the shifts are the shifts  $\lfloor s\delta \rfloor$ , where

$$\delta \in \left\{ (x, y) \mid \begin{array}{l} (\forall |k - x| \leq 1 \wedge |l - y| \leq 1, m_{xy} \leq m_{kl}) \\ \wedge \frac{\min m_{ij}}{m_{xy}} > 0.8 \end{array} \right\}$$

Starting at  $[s\delta]$ , an iterative search for the minimum is made on full-size images. Let  $\delta_i$  be the shift value at the current iteration ( $\delta_0 = s\delta$ ), then shift at the next iteration

$$\delta_{i+1} = \arg \min_{\substack{k=\delta_x, \delta_x \pm 1, \\ l=\delta_y, \delta_y \pm 1}} \mu(G_1^{(\delta_x, \delta_y)}, G_2).$$

Iteration  $k$  is the last if  $\delta_k = \delta_{k-1}$ . As a solution of (2), a shift is chosen from the shifts obtained for all initial approximations  $[s\delta]$ , giving a minimal estimate of the difference. When calculating the shifts on full-size images,  $d_{cut} \approx \max(5, s)$ .

### III. PARAMETRIC EDGES

#### A. Construction of parametric edges

Using parametrically defined edges allows you to more accurately compare them. It is convenient to represent the edges in the form of broken lines that can branch out at fracture points. Parametric edges can be constructed on the basis of thin (having a thickness of one pixel) discrete edges using the following algorithm:

1. Search for intersection points - Fig. 1a., branch points with branch lengths greater than 1 pixel. The search is performed in depth, branches 1 pixel long are deleted. On edges without branches, one arbitrary point is also considered as a branch point.
2. Selection of end points - Fig. 1a. The final points are the points that are the most far from the nearest intersection.
3. Recursive construction of line segments - Fig. 1b. The construction of line segments approximating a sequence of pixels between the intersection points and end points with an accuracy of  $d_{max}$  pixels

An example of the constructed parametric edges is shown in Fig. 1c.

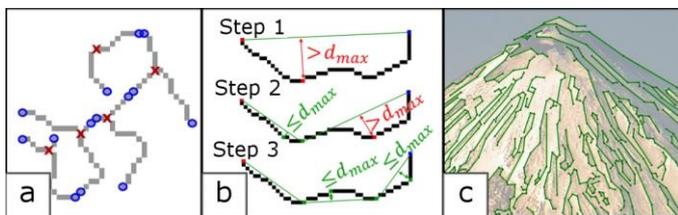


Fig. 1. Building parametric edges: a. – Cross points (red crosses) and ending points (blue circles); b. – recursive line building; c. – example of edges built.

#### B. Comparison of parametric edges

We denote by  $S$  and  $S'$  the sets of all segments on the first and second images. Then for segments  $s \in S$  and  $s' \in S'$  the comparison operation is defined - Fig. 2a. Segments  $s$  and  $s'$  are only to be compared if the angle between them is  $\alpha(s, s') < \alpha_{max}$ ,  $\alpha_{max} \in [0.4; 0.7]$ . The closeness of the

segments is determined by the mean width  $w(s, s') = \Sigma(s, s')l(s, s')^{-1}$ ,  $w(s, s') < R$ , where  $l(s, s')$  is the length of the common part of the segments  $s$  and  $s'$ ,  $\Sigma(s, s')$  is the area of the figure formed by common parts of these segments - Fig. 2a, and  $R$  is the maximum allowed distance between matched segments. In the case where  $w(s, s') \geq R$  or the common zone is absent ( $l(s, s') \leq 0$ ), the segments are also not comparable.

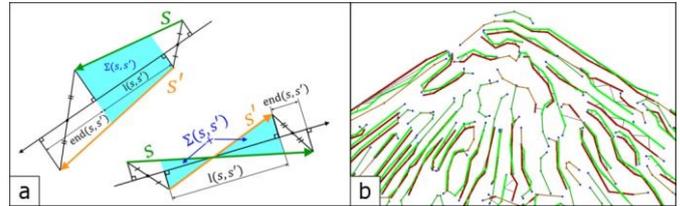


Fig. 2. a. – the different configurations of the segments  $s$  and  $s'$ , being compared,  $l(s, s') \leq 0$  is the length of the common part,  $end(s, s')$  is the difference of the endings of the segments,  $\Sigma(s, s')$  is the area of the figure, formed by segments; b - Comparison of edges. The edges of different images are shown in different shades, the thick lines denote the matched segments, the segment links by thin black lines connecting the centers of the segments.

Starting from the pair of segments  $s$  and  $s'$  to be compared, tracing along the edges is performed according to the following rules:

1. Tracing is performed in the direction of the segments (arrows in Fig. 2a), then all directions are inverted and the trace is repeated
2. If  $end(s, s') > d_{max}$  and  $s$  ends earlier, then the next pair for comparison is the pair  $s \rightarrow next$  and  $s'$ .
3. If  $end(s, s') > d_{max}$  and  $s'$  ends earlier, then the next pair for comparison is the pair  $s$  and  $s' \rightarrow next$ .
4. If in paragraphs 2 and 3 there are several options for continuation of the line, the one having a smaller angle with a non-continuing segment is chosen.
5. If  $end(s, s') \leq d_{max}$ , or 2 and 3 did not lead to continuation at  $end(s, s') \leq R$ , then the next pair is selected  $s \rightarrow next$ ,  $s' \rightarrow next$ . If there are several variants of continuation, then all directions that do not contradict each other with the least mean width are selected.
6. Each pair that was compared during the trace is remembered.
7. Tracing ends when there are no options to continue, or a pair of segments that have already been stored are selected.
8. For segments that have only partially entered the trace, the part that was not entered is remembered.

The result of the  $j$ -th trace operation is denoted by  $P_j = (s_{ji}, s'_{ji})$ ,  $i = 1, 2, \dots, n_j$ , where  $n_j$  is the number of matched pairs, and  $s_{ji} \in S$  and  $s'_{ji} \in S'$  are matched segments. The vector of the matched segments has the estimate  $Q_j$ :

$$Q_j = \bar{l}_j \begin{cases} 1, & \bar{w}_j \leq d_{max} \\ 3t_j^2 - 2t_j^3, & \bar{w}_j > d_{max} \end{cases}, \quad (3)$$

$$\bar{l}_j = \sum_{i=1}^{n_j} l(s_{ji}, s'_{ji}), \quad t_j = \frac{R - \bar{w}_j}{R - d_{max}}, \quad \bar{w}_j = \frac{1}{\bar{l}_j} \sum_{i=1}^{n_j} \Sigma(s_{ji}, s'_{ji}).$$

Let's write down the algorithm for matching all edges on two images:

1. Assign:  $j = 1, V = \emptyset, G = \emptyset$ .
2. For all pairs  $(s, s')$ ,  $s \in S$  and  $s' \in S' \cap H(s)$ , where  $H(s)$  is the operation of inquiry to the spatial hash, which results in finding all the segments from  $S'$ , which are closer to  $s$  than  $R$ , we perform a trace if the pair  $(s, s')$  was not stored as a result of previous traces. For the trace,  $\mathbf{P}_j$  is stored and  $Q_j$  (3) is calculated, and the counter  $j$  is incremented by one.
3. All  $\mathbf{P}_j$  are scanned in decreasing order of  $Q_j$ . If  $s_{ji} \notin V$  and  $s'_{ji} \notin V$ ,  $i = 1, 2, \dots, n_j$ , then  $V \leftarrow V \cup s_{ji} \cup s'_{ji}$ ,  $G \leftarrow G \cup j$ .
4. Calculate the visibility of the edges of the second image on the first:

$$\sigma = \frac{\sum_{j \in G} \sum_{i=1}^{n_j} l(s_{ji}, s'_{ji})}{\sum_{s' \in S'} |s'|}, \quad (4)$$

where  $|s'|$  – the length of segment  $s'$ .

To process segments that are only partially matched, a list of occupied parts is maintained. An example of the associated edges is shown in Fig. 2b.

Before matching, the parametric edges are aligned relative to each other by shifting the edges of the second image by the vector  $\delta$ , calculated on the basis of the discrete edges (2). Using the vector  $\delta$  makes it possible to compensate for relative displacements of more than 20% of the image width.

#### IV. BUILDING REFERENCE EDGES

To determine the visibility of a volcano in a certain image, it is necessary to compare the edges detected in this image to the reference edges (edges that are saved regardless of the time of year). To construct the reference edges, a certain number of  $m \in [3..10]$  of the previously selected images of the volcano, obtained in clear weather at different times of the year - reference images, is used.

For each  $i$ -image,  $i = 1, 2, \dots, m$ , discrete edges  $G_i$  are constructed and the shifts  $\delta(G_i, G_k)$ ,  $k = \arg \max_k |G_k|$  are calculated. For each reference image, the offset is calculated

$$\delta_i = \delta(G_i, G_k) - \frac{1}{2} \left( \min_{l=1,2,\dots,m} \delta(G_l, G_k) + \max_{l=1,2,\dots,m} \delta(G_l, G_k) \right),$$

where **min** and **max** are componentwise functions of minimum and maximum. Displaced edges on all reference images are compared with each other. For each segment, the number of times it was placed in the set  $V$  (Section 3.B) is calculated. Segments marked at least  $\gamma m$  times are considered as reference,  $\gamma \in [0.5..1.0]$ . From the reference segments, reference edges are compiled. If a certain segment has been added to the reference edges, all segments forming pairs with the added segment are not added. In Fig. 3a, b are shown examples of reference edges obtained for  $m=6$ ,  $\gamma = 0.5$  and  $\gamma = 1.0$ .

The reference edges are divided into inner and outer edges. The outer ones are made up of segments of lines that have an intersection point with vertical lines drawn at a distance of one pixel with the maximum coordinate  $y$  - Fig. 3c. Final visibility estimate for the test image

$$\sigma' = \max \left( \min \left( 1, \frac{\sigma_{ext}}{\min_{i=1,2,\dots,m} \bar{\sigma}_{ext,i}} \right), \min \left( 1, \frac{\sigma_{int}}{\min_{i=1,2,\dots,m} \bar{\sigma}_{int,i}} \right) \right), \quad (5)$$

where  $\sigma_{ext}$  and  $\sigma_{int}$  are the estimates (4) obtained by comparing the edges of the test image with the external and internal reference edges, and  $\bar{\sigma}_{ext,i}$  and  $\bar{\sigma}_{int,i}$  – estimates (4) obtained by comparing the edges of the  $i$ -th reference image with external and internal reference edges.

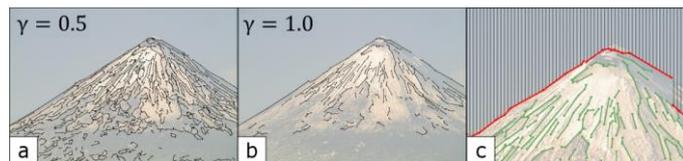


Fig. 3. Reference edges: a. –  $\gamma = 0.5$ , b. –  $\gamma = 1.0$ , c. – separation of external and internal edges, vertical lines are thinned.

#### V. RESULTS

For testing the algorithms for constructing, aligning, and comparing parametric edges, 3000 shots were taken for volcano Sheveluch for the time periods from 02.12.2014 to 02.02.2015, for volcano Klyuchevskoy - from 27.11.2014 to 08.12.2014 and for volcano Kizimen - from 27.11.2014 to 01.05.2015. As reference images, 4 images were taken for each camera. Image processing was carried out at the Shared Facility "Data Center FEB RAS" [12] on a computer server with an Intel Xeon 5150 processor, a clock speed of 2.66 GHz and an 8 GB RAM. Test results are presented in Table. 1.

Table I. DISTRIBUTION OF RESULTS BY VOLCANO VISIBILITY CATEGORIES

Visibility of volcano on image	Volcanoes		
	<i>Sheveluch</i>	<i>Klyuchevskoy</i>	<i>Kizimen</i>
Volcano is not visible	48.3%	63.5%	64.4%
Volcano is visible	51.7%	36.5%	35.6%

The processing time of the images of Kizimen volcano, having a resolution of 800x600 was 20 minutes, and the images of volcano Sheveluch and Klyuchevskoy, having a resolution of 1024x768 – processing time was 36 and 30 minutes respectively. The quality of the algorithm was manually estimated by determining the number of incorrectly classified images, using a developed web application in the form of a gallery (Table 2).

Table II. DISTRIBUTION OF THE RESULTS OF INCORRECTLY CLASSIFIED IMAGES

Visibility of volcano on image	Volcanoes		
	<i>Sheveluch</i>	<i>Klyuchevskoy</i>	<i>Kizimen</i>
Volcano is not visible	28 (0.93%)	15 (0.5%)	15 (0.5%)
Volcano is visible	1 (0.03%)	0 (0%)	58 (1.93%)

Visual assessment of the test results showed that most of the images are analyzed correctly, and the proportion of erroneously classified images is insignificant.

## VI. CONCLUSION

The considered algorithms for constructing, aligning and comparing parametric edges were tested on the archive of images of the volcano monitoring system [1]. Data were examined from three different cameras, observing Klyuchevskoy, Sheveluch and Kizimen volcanoes. In total, more than 9000 images were processed, while the result of the visibility evaluation is completely consistent with the human perception.

In comparison with previous studies [6], [11], the algorithm for equalizing the shifts between the edges was modified, which allowed processing images with relative shifts of more than 20% of the image size. It was also suggested to divide the reference edges into internal and external and separate estimation calculations, which significantly increased the visibility scores in pictures with unsuccessful lighting, where the volcano is clearly visible.

## ACKNOWLEDGMENTS

The studies were supported by the Russian Foundation for Basic Research (Grants No. 16-07-00156, 16-37-00026 Mol\_a) and FEB RAS (Grants No. 15-I-4-071, No. 15-I-4-072). When processing images, the resources of the Shared Facility Center "Data Center of FEB RAS"

## REFERENCES

[1] A.A. Sorokin, S.P. Korolev, I.M. Romanova, O.A. Girina, I.P. Urmanov, "The Kamchatka volcano video monitoring system", Proceedings of 2016 6th International Workshop on Computer Science and Engineering, WCSE 2016, Tokyo, Japan, pp.734-737, 17-19 June 2016.

[2] Th. Sutter, F. Nater, Ch. Sigg, "Camera based visibility estimation", In Proc. TECO – 2016 (Technical Conference on Meteorological and

Environmental Instruments and Methods of Observation), P2(64), 27-30 September 2016.

[3] L. Caraffa, J.-Ph. Tarel, "Daytime fog detection and density estimation with entropy minimization", ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 2014, vol. II-3, pp.25–31.

[4] R.J. Radke, S. Andra, O. Al-Kofahi, B. Roysam, "Image change detection algorithms: a systematic survey", IEEE Transactions on Image Processing, 2005, vol. 14(3), pp.294–307.

[5] Y. Benezeth, P.-M. Jodoin, B. Emile, H. Laurent, Ch. Rosenberger, "Comparative study of background subtraction algorithms", Journal of Electronic Imaging, Society of Photo-optical Instrumentation Engineers, 2010, vol. 19(3), DOI: 10.1117/1.3456695.

[6] I.P. Urmanov, A.N. Kamaev, A.A. Sorokin, S.P. Korolev, "The assessment of the visibility and the status of volcanoes using sequences of video observation images", Computational Technologies, 2016, vol. 21(3), pp.80-90.

[7] J. Canny. "A computational approach to edge detection. Pattern analysis and machine intelligence", IEEE Transactions on PAMI, vol. 8(6), pp.679–698, November 1986.

[8] J. H. Elder, S. W. Zucker, "Local scale control for edge detection and blur estimation", IEEE Transactions on PAMI, 1998, vol. 20(7), pp.699–716.

[9] D. Martin, C. Fowlkes, J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues", IEEE Transactions on PAMI, 2004, vol. 26(5), pp.530–549.

[10] G. Borgefors, "Distance transformations in digital images", CVGIP, vol. 34(3), pp. 344–371, June 1986.

[11] I.P. Urmanov, A.N. Kamaev, A.A. Sorokin, "Complex analysis of images to determine the visibility of volcanoes", 26th International Conference and School-seminar on Computer Graphics and Vision GraphiCon 2016, Nizhny Novgorod, Russia, 2016, pp.95-99.