DOI: 10.20535/1810-0546.2018.3.136433
UDC 004.932

O.S. Shkurat, Ye.S. Sulema*, A.I. Dychka
Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine

COMPLICATED SHAPES ESTIMATION METHOD FOR OBJECTS ANALYSIS IN VIDEO SURVEILLANCE SYSTEMS

Background. The evaluation of video image objects is a relatively difficult task. While solving the task of the geometric representation of a surveillance object, the following additional factors should be considered: possible overlapping of objects, similarity of complex elements, similarity of object elements and background, etc.
Objective. The development of a method for complicated objects shape evaluation for application in video surveillance systems for estimation of dynamics of an object's movement, examination of the object's behavior on a probable execution of unauthorized actions, and for other tasks.
Methods. The procedure of the background subtraction is used for identification of a raster shape of the surveillance object. To detect a vector shape of the object contours, the DEI approach is applied. The sorting procedures are used for identification of reference contour points and for forming the smooth curves.
Results. The proposed method includes the following stages: color space conversion and normalization, object shape detection, contours detection and analysis, sorting of vector data, forming of smooth contour curve, object area computing. When the contour points number is reduced in 1.5 times, an average error of the proposed method compared with the DEI approach for accuracy rate is 0.75 %, for performance rate it is 8.43 %, for resource consuming rate it is 3.09 %.
Conclusions. The proposed method allows to define an array of vector contour points which represent an "approximate" surveillance object of a complicated shape and it minimizes the data volume to be used in further analysis of a motion trajectory and other similar tasks without decreasing the accuracy. In addition, this method enables describing the surveillance object by an equal quantity of contour points that in turn can simplify the task of surveillance objects classification.
Keywords: DEI approach; image feature extraction; vector filtering of image; vector contour analysis.

Introduction

Video surveillance is an important task which can be focused on visual control of road traffic [1–5], monitoring of hazardous situations [6, 7], monitoring in different areas of manufacturing [8, 9], health care [10] or other applications. The essential part of the video surveillance procedure is the detection of presence of an object under observation. This detection is based on a certain specific feature of the object such as color, shape, behavior, etc. In our research we consider the object shape as the main feature.

The object shape is determinate by a contour which outlines the object as accurate as possible. This contour can be represented as either raster or vector curve. In the second case, the shape analysis is more flexible because we can scale the curve and change the number of points defining the contour. The latter is very important for both maximization of analysis accuracy and minimization of the consuming of computing resources used for further data processing because if there is insufficient data set about the object shape, its further analysis can lead to a wrong result; at the same time, if there is redundant data about the object shape, the need in increasing computing resources can appear. Therefore, it is desirable to find relatively optimal amount of data which enables an object shape analysis with minimal requirements to the computing resources.

A proposed method enables estimation of an object shape with a desirable level of accuracy and, thus, it can help to get an amount of data about the object shape which can be considered as optimal for a certain task.

Problem Statement

The research objective is the development of a method for shape evaluation of a complicated object. The method is supposed to be used in video surveillance systems as the preprocessing for further analysis of the object's movement dynamics, examination of the object's behavior on a probable execution of unauthorized actions, and in other similar tasks.

Related Work

The literature review shows that there is a relatively large number of researches related to tasks similar to the task we solve in our investigation.

* corresponding author: sulema@pzks.fpm.kpi.ua

However, the approaches used for solving these related tasks are quite different.

Thus, in [11], the Moghaddam and Enright compared three interpolants defined for three-dimensional elliptic Partial Differential Equations (PDEs) over an unstructured mesh. The authors assert that the obtained results showed that pure tri-cubic interpolant generates more accurate results rather than tri-quadratic interpolants.

In [12], the authors proposed three fast contouring algorithms for visualizing the solution of PDEs based on the Pure Cubic Interpolant. These algorithms do not need a fine structured approximation and the authors assert that their algorithms work efficiently with the original scattered data. The basic idea of the proposed approach is to identify the intersection points between contour curves and the sides of each triangle and then draw smooth contour curves connecting these points.

Enright developed in [13] Differential Equation Interpolant (DEI) approach which efficiently approximates the values of the solution of a PDE at off-mesh points. This approach allows to reach high precision at the resulting off-mesh points.

In [14], Barequet and Sharir presented a technique for piecewise-linear surface reconstruction from a series of parallel polygonal cross-sections. The proposed algorithm uses a partial curve matching technique.

Lee, Wolberg, and Shin presented in [15] a fast algorithm for scattered data interpolation and approximation. The authors assert that their algorithm makes use of a coarse-to-fine hierarchy of control lattices to generate a sequence of bicubic B-spline functions whose sum approaches the desired interpolation function.

These and other similar methods enable producing an accurate contour of an object, however, in our opinion they can be complemented by additional procedures which enables minimization of the data amount about an object shape without appreciable change in accuracy and, in this way, we can achieve minimization of the requirements to the computing resources of a video surveillance system.

Method Description

The proposed method of video image analysis includes the following steps:

A. Color space conversion and normalization.
B. Object shape detection.
C. Contours detection and analysis.
D. Sorting of vector data.
E. Forming of smooth contour curve.
F. Object area computing.

A. Color Space Conversion and Normalization

Video data can be represented as a multiframe array. Array elements are images, which can be binary, gray-scale, and color.

Binary and gray-scale images of size $M \times N$ can be described by a two-dimension matrix $I(i,j) = a$, where a is a color intensity of a pixel with the coordinates (i,j), $i = 1,2,\ldots,M$, $j = 1,2,\ldots,N$. For binary images, $a \in [0;1]$, for gray-scale images $a \in [0;255]$.

A color image of size $M \times N$ can be descrybed by a three-dimension matrix $I(i,j,k) = a$, where a is a color intensity of a pixel, $a \in [0;255]$, i,j are the pixel coordinates, $i = 1,2,...,M$, $j = 1,2,...,N$, k is an order number of the color component, which can be considered as an additional coordinate which defines the color space, $k = 1,2,3$. Alternatively, the color image can be represented by the matrix $I(i,j) = (c_1, c_2, c_3)$, where c_1, c_2, and c_3 are intensities of the color components. It means that each pixel of a color image corresponds to a vector in a certain color space. The most frequently used color model is RGB model, which represents the image as $I(i,j) = (r,g,b)$, where r,g,b are color components intensity of a pixel accordingly, i,j are the pixel coordinates, $i = 1,2,...,M$, $j = 1,2,...,N$. Vectors (r,g,b) in the color space of RGB model identify the pixel color. However, the most important information for the object shape estimation is luminance and we need to separate it from other color data. It can be fulfilled by conversion the image from RGB model into HSV, YCbCr, YIQ, XYZ, Lab or other similar models. In our research, we use conversion into color model YCbCr [16], which is defined by the formula:

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.00 \\ 112.00 & -93.786 & -18.214 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix} + \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix},$$

where $R,G,B \in [0;255]$.

As a result of conversion procedure, the range of pixel values by components are the following: $Y \in [16;235]$, $Cb, Cr \in [16;240]$, where Y is the luminance component and Cb and Cr are the blue-difference and red-difference chrominance components correspondingly. In order to simplify further processing, including negation operation, these values should be normalized according to the formula:

$$\begin{bmatrix} Y' \\ Cb' \\ Cr' \end{bmatrix} = \begin{bmatrix} \sin\left(\frac{Y}{255}\right) \\ \sin\left(\frac{Cb}{255}\right) \\ \sin\left(\frac{Cr}{255}\right) \end{bmatrix},$$

where $Y' \in [0.0627; 0.7966]$, $Cb', Cr' \in [0.0627; 0.8083]$.

B. Object Shape Detection

Initial data for this stage are color images, which are the background image I_b and the image I_t to be tested on the presence of any moving object. The background image is the image of the ordinary view of the scene to be under surveillance, it does not include any moving objects on it.

The color vectors (b_1, b_2, b_3) and (t_1, t_2, t_3) represent values of the pixel intensity of these images:

$$I_b(i, j) = (b_1, b_2, b_3),$$
$$I_t(i, j) = (t_1, t_2, t_3)$$

where $i = 1, 2, \ldots, M$, $j = 1, 2, \ldots, N$ are the pixel coordinates of the images of size $M \times N$, $b_1, t_1 \in$ $\in [0.0627; 0.7966]$, $b_{2,3}, t_{2,3} \in [0.0627; 0.8083]$ are the pixel intensity values by Y', Cb', Cr' components of the background image and the object image on the background accordingly.

In order to calculate the logical negation operation for a color image, we use an additional matrix which is considered as complementary to the color image to be analyzed. In this research we use the matrix V, which corresponds to white color $V(i, j, k)$, $k = 1, 2, 3$. Alternatively, the elements values of matrix V can belong to the range $[0.8083; 1]$. The logical negation operation is implemented in the following way:

$$I_b^n(i, j) = \begin{cases} (V(i, j) - I_b(i, j)), \text{ if } (V(i, j) - I_b(i, j)) > 0 \\ 0, \text{ otherwise} \end{cases}$$
$$= (b_1^n, b_2^n, b_3^n),$$

$$I_t^n(i, j) = \begin{cases} (V(i, j) - I_t(i, j)), \text{ if } (V(i, j) - I_t(i, j)) > 0 \\ 0, \text{ otherwise} \end{cases}$$
$$= (t_1^n, t_2^n, t_3^n),$$

where b_k^n, t_k^n are intensity values of the image pixels, which are the result of logical negation operation, $k = 1, 2, 3$.

The detection of the moving object appeared on the background is fulfilled according to the formula:

$$M(i, j) = |(I_b(i, j) \cup I_t^n(i, j)) - (I_b^n(i, j) \cup I_t(i, j))|.$$

In case if input images are color, the following formula is used:

$$M(i, j)$$
$$= |\max(b_k(i, j), t_k^n(i, j)) - \max(b_k^n(i, j), t_k(i, j))|.$$

C. Contours Detection and Analysis

The input data for the contour detection procedure is luminance data which corresponds to Y-component of the color vector.

We use DEI approach [13, 16] to determine the image contour of the vector shape. As a result, a matrix of the image contour is obtained:

$$Contours =$$
$$= [Contour(1), \ldots, Contour(q), \ldots, Contour(Q)],$$

$$Contour(q) = \begin{bmatrix} level_q & x_1 & x_2 & \ldots & x_u \\ num_q & y_1 & y_2 & \ldots & y_u \end{bmatrix},$$

where Q is a quantity of the image contours, x_u, y_u are the contour points coordinates, $level_q$ is the contour level, num_q is points quantity of q-contour accordingly, $q = 1, 2, \ldots, Q$.

Let us consider q-contour, which displays the shape or rather the area that the moving object occupies in the two-dimensional space as the main contour:

$$\theta = \max(num_q),$$

where $q = 1, 2, \ldots, Q$, θ is the quantity of points of the main contour.

For implementation of the sorting and approximation, it is necessary to normalize the obtained data by performing the procedure of redistribution of the contour points relatively to the coordinate origin. Let $Contour$ and O matrices have sizes $2 \times \theta$ and 2×1 accordingly. The values of $Contour$ are point coordinates of the main contour, values of O are equal to zero. The minimum distance and the corresponded point, the location of which is the closest to the coordinate origin, can be determined by the formula:

$$[\mu, \rho] = $$
$$= \min\left(\sqrt{\begin{matrix}(Contour(x_i) - O(x))^2 \\ + (Contour(y_i) - O(y))^2\end{matrix}} \right),$$

where μ is the value of a minimum distance, ρ is a column number of the *Contour* matrix, x_i, y_i are coordinates of the contour points, $i = 1, 2, \ldots, \theta$.

The next step is the redistribution of the *Contour* matrix values (Fig. 1). The redistribution can be fulfilled either clockwise (Fig. 1, *a*) or counterclockwise (Fig. 1, *b*).

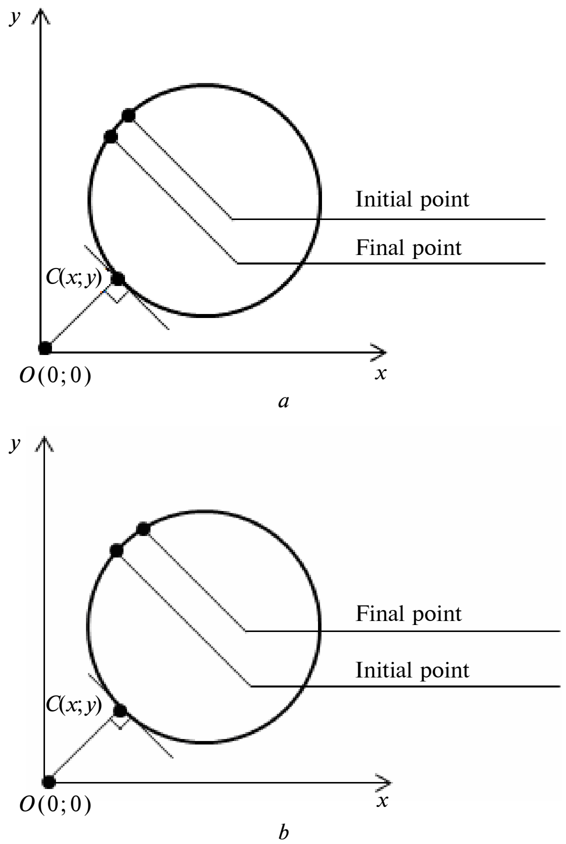

Fig. 1. Options for the initial location of the points in *Contour*: *a* — clockwise, *b* — counterclockwise

The contour image obtained as a result of the actions described above can include excess contour points which are visualized as loops, wave-like inequalities etc. and can be considered as noise. To smooth the obtained contour, the median filtration can be used [17–21].

Since vector data processing, including images, significantly improves speed index compared to raster data, the alternative approach of contour smoothing is applied in the proposed method. This approach is the post-processing procedure for contour points detection.

Let image contour be described by the matrix:

$$Contour = \begin{pmatrix} x_1\, x_2 & & x_{i-1}\, x_i \\ & \ldots & \\ y_1\, y_2 & & y_{i-1}\, y_i \end{pmatrix},$$

where x_i, y_i are coordinates of contour points, $i = 1, 2, \ldots, \theta$, θ is the quantity of points.

Therefore, for each stage of the contour smoothing procedure the coordinates of the vectors with permanent size $1 \times W$, which contain x, y coordinates of contour points, are determined. For example, for some contour point C_i with coordinates (x_1, y_1) and $W = 5$, these vectors are determined in the following way:

$$\omega_x = (x_1, x_2, x_3, x_4, x_5),$$
$$\omega_y = (y_1, y_2, y_3, y_4, y_5).$$

The next stage is to redistribute values of the obtained vectors and to determine the median. For example, if ω_x, ω_y values can be described after sorting as follows:

$$\omega'_x = (x_2, x_3, x_4, x_5, x_1),$$
$$\omega'_y = (y_5, y_1, y_2, y_3, y_4),$$

then x_4 and y_2 are medians of ω_x, ω_y vectors accordingly. Thus, the initial coordinates of contour points are changed, and the coordinates of C_i point are (x_4, y_2).

D. Sorting of Vector Data

The data sorting procedure allows both to detect the “reference” contour points and to remove the uninformative points, solving the task of vector data compression.

The initial stage of sorting is to determine quantity of lines or segments that will collectively shape the object contour. The next stage is to shape a template of the segment by value range of which the “reference” contour points are defined.

A contour of the surveillance object can be described as follows:

$$L = \{ l_p(x, y) \mid p \in [1; P],\ l_p \in Contour \}, \quad (1)$$

where P is a parameter, values of which indicate the quantity of segments.

The range of values of the template segment is limited by a size of the input images and is determined by T vector:

$$T = 1 : D : N,$$

where D is a parameter of refinement, $0 < D < N$, N is a size of the vector image by x coordinates or of the raster image by j coordinates.

The procedure of sorting is implemented for x coordinates values for each l_p segment. Let us detect the values of x and y coordinates of the first segment:

$$l_{1x} = \{x_1, x_2, \ldots, x_{\theta/P}\},$$

$$l_{1y} = \{y_1, y_2, \ldots, y_{\theta/P}\}.$$

Values of T vector are constant for each l_p segment:

$$T = \left(1, 1 + D, 1 + 2 \times D, \ldots, 1 + \mathrm{int}\left(\frac{N-1}{D} \right) \times D \right),$$

where int is an operation of the integer-valued rounding.

The finding of x coordinates for "reference" vector points of L segments set is performed by finding a minimum distance for each T element from l_{px} vectors:

$$[\sigma, \tau] = \min\left(\sqrt{(T_h - l_{1x})^2} \right),$$

where σ is a value of the minimum distance, τ is the column number of l_{1x} vector, $h = 1, 2, \ldots, 1 + \mathrm{int}\left(\frac{N-1}{D} \right)$.

The y coordinate for "reference" points remains unchanged and is taken from l_{py} set, that is, in this case equals y_τ, $y_\tau \in l_{1y}$. Thus, the set of segments

$$L' = \{ l'_p(x, y) \mid p \in [1; P], l'_p \in Contour \}, \quad (2)$$

which connects "reference" points of initial *Contour*, is formed.

Singularity of this algorithm of data sorting consists in giving an opportunity to describe any surveillance object and any its contours by an equal quantity of points which can be calculated in the following way:

$$R = P \times \left(\mathrm{int}\left(\frac{N-1}{D} \right) + 1 \right). \quad (3)$$

Representation of all surveillance objects by the equal quantity of contour points allows to solve the tracing task effectively since further processing and analysis will be performed on similarity-represented objects. It is worth mentioning that after a point falls to L' set, it is removed from L set.

The quantity of segments and the value of the refinement parameter directly influence the accuracy of reproduction of the initial contour. The more the quantity of segments is and/or the less the refinement parameter is, the more accurate results of sorting are obtained. At the same time, both parameters influence the performance rate significantly. To disregard this indicator is expedient if the surveillance object has a complex shape. As a result, the additional procedures, which allow to determine the values range of these parameters, appear and complicate the proposed method.

The alternative approach for solving this task is to apply the data sorting algorithm on the assumption of that the points which fall into L' set are not removed from L set. It enables reproducing the initial contour of the object in a maximally accurate way (depending on the parameter D). However, the negative aspect – the presence of certain quantity of duplicated contour points – appears in this case. Another alternative approach is to set the more accurate range of T template vector applying the sorting procedure twice, but it decreases the performance rate of the proposed method. Therefore, it is expedient to use the first approach of data sorting since the result of its applying is satisfactory for a wide range of input data.

E. Forming of Smooth Contour Curve

For further processing and analysis of the surveillance object, it is necessary to represent L' set by a certain figure, which can be described by lines that consistently combine a strictly ordered array of points. For this, it is necessary to apply the sorting procedure of l'_p segments (2) by x and y coordinates. The procedure of determination of the point location on l'_p segments follows the rule of minimum distance:

$$\left(\sqrt{(l'_p(x_i) - l'_p(x_{i+1}))^2 + (l'_p(y_i) - l'_p(y_{i+1}))^2} \right) \to \min,$$

where x_i, y_i are coordinates of "reference" contour points.

In order to avoid an additional processing procedure, the first points of l_p segments (1) are set as the initial points. These points must be removed once the array L of ordered points is formed:

$$L'' = \{l''_p(x, y) \mid p \in [1; P], l''_p \in Contour\}.$$

F. Object Area Computing

In order to fulfill the preliminary classification of the moving object, we calculate its area using the formula proposed in [16]:

$$S = \frac{1}{2}\sum_{r=1}^{R-1}(x_r + x_{r+1})$$

$$\times (y_r - y_{r+1}) + (x_R + x_1) \times (y_R - y_1),$$

where R is a total quantity of "reference" contour points and it is defined in (3).

Results

In our experiments, we used MP4 and 3GP videos got from CCTV cameras installed on roads, shopping centers, supermarkets, and offices. Each video file is converted into a multi-frame array of JPEG images. The analysis was performed for each surveillance object detected by using a pair of frames: the background frame, where the object is absent, and every next test frame, where the object can be present. The experiment included approximately 200 frames.

The task of contour smoothing can be solved by two ways. The first way is to apply the median filtering as preprocessing of the contour detection procedure. The second way is to apply the vector filtering as the post-processing of the contour detection procedure (Fig. 2). Since the proposed method uses vector data processing, the second way was used in the experiments.

The initial quantity of contour points in one of frames in the test videos is 2093 points. The example of contour detection by using the proposed method is showed in Fig. 3. The final quantity of contour points is 1405. Thus, the number of points is decreased approximately 1.5 times.

Due to the accuracy parameter used in the proposed method, the quantity of contour points can be reduced in several times along with the correct representation of the surveillance object shape (Fig. 4).

The criteria for estimation of the proposed method include accuracy, performance rate, and computing resource consuming (Fig. 5). The accu-

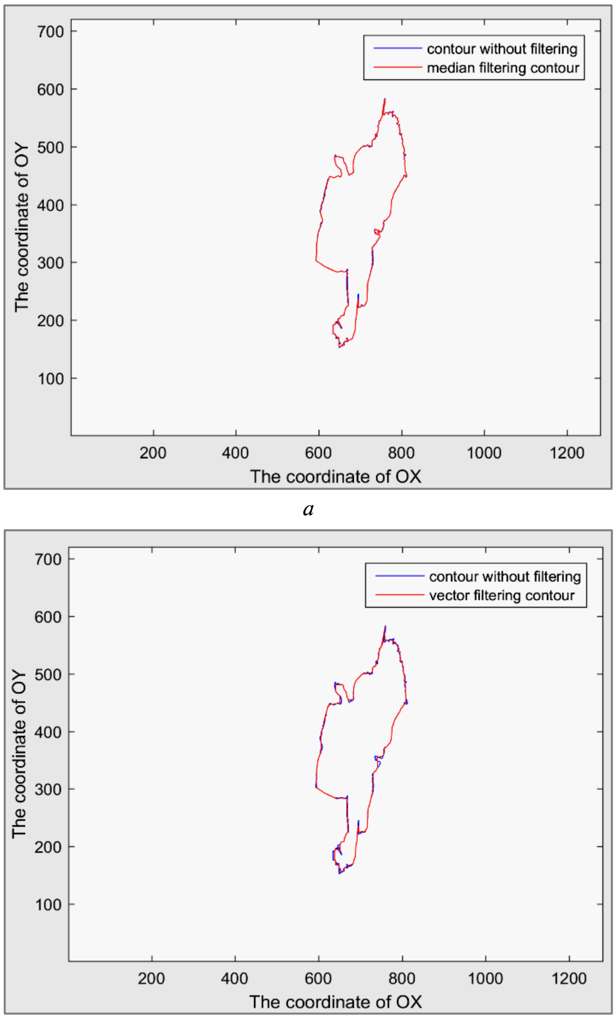

Fig. 2. Smoothing filters: *a* – median, *b* – vector

racy rate is defined by an error value of the surveillance object area in comparison with the DEI approach. Delaunay triangulation [22] and the determination of the total area of obtained triangles are used for estimation of the performance rate and computing resource consuming as additional load procedures in order to make the gain in the performance more evident.

As alternative approaches, the two algorithms of contour detection of the surveillance object (Vectoring of Raster Method 1, Vectoring of Raster Method 2) are applied for comparison of application results of the proposed method. The first technique includes the following processing stages: the detection of a surveillance object, extraction of Y-component data, thresholding processing, contour detection by the DEI approach, detection of a main contour and redistribution of contour points relatively to the coordinate origin, determining of

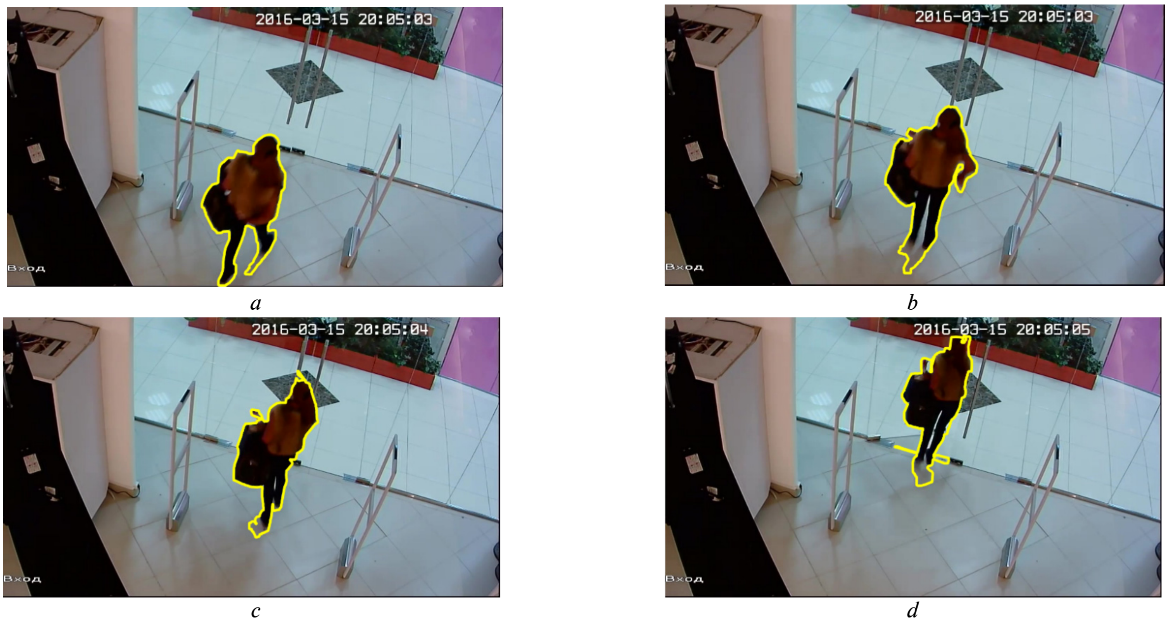

Fig. 3. Contour of the surveillance object: *a* — movement 1; *b* — movement 2; *c* — movement 3; *d* — movement 4

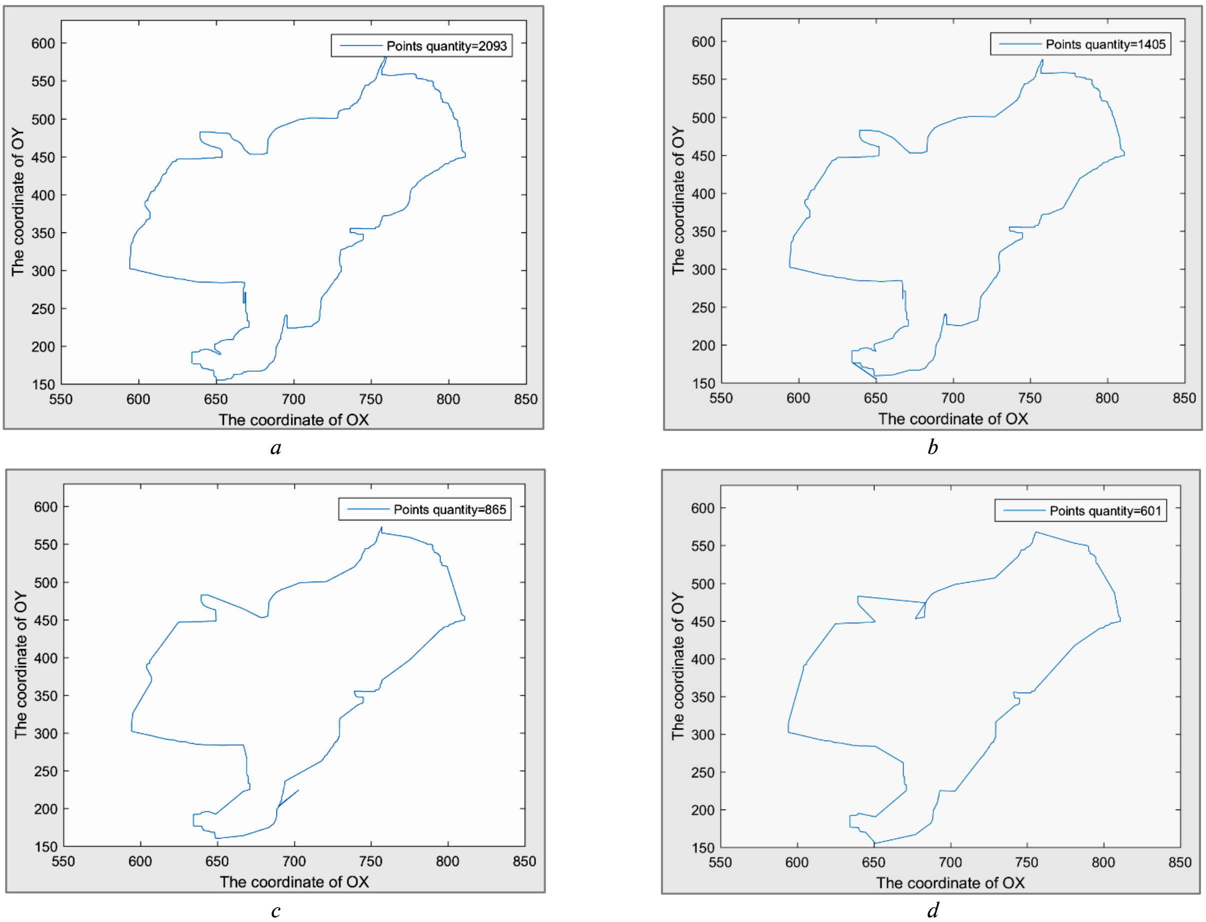

Fig. 4. Object contour and points quantity: *a* — initial contour, quantity of points is 2093; *b* — final contour, 1405 points; *c* — contour with 865 points; *d* — contour with 601 points

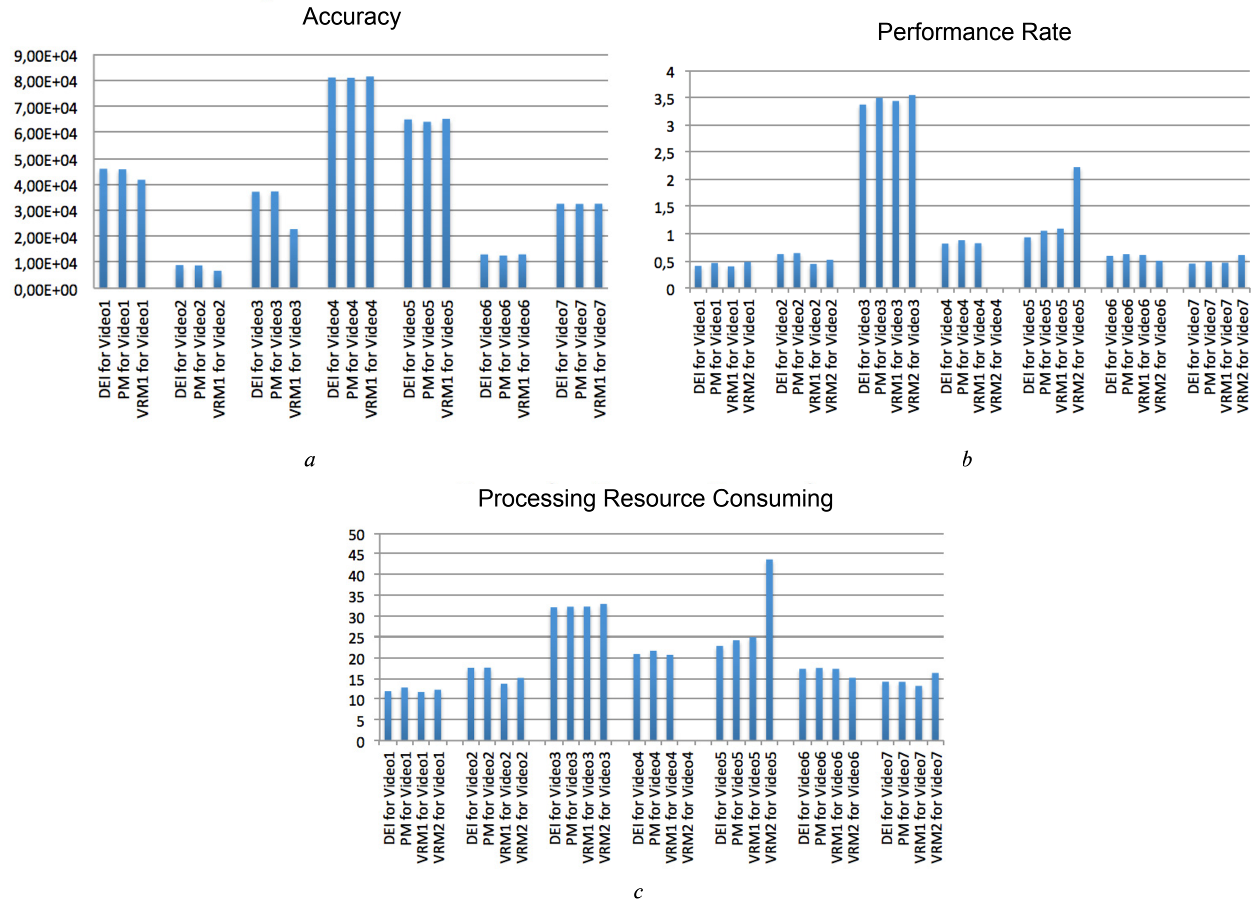

Fig. 5. Results of DEI approach (DEI), proposed method (PM), Vectoring of rather method 1 (VRM1) and Vectoring of rather method 2 (VRM2) for rates: *a* — Accuracy, *b* — Performance Rate, *c* — Processing Resource Consuming

Table. Comparison of DEI approach and the proposed methods

Method	Average Values			
	Accuracy	Area of triangles	Performance Rate, sec	CPU usage, %
DEI approach	4.1047e+04	5.1981e+04	0.6379	17.452
Proposed method $D > 1.5$	4.0740e+04	5.1100e+04	0.6917	17.992
Error	0.75 %	1.69 %	8.43 %	3.09 %

the object area and Delaunay triangulation, and determining of areas of the obtained triangles. The second technique differs from the first on the additional stage of the image contour detection by Sobel method after the thresholding processing.

The results in Fig. 5 match a mean value of the chosen criteria for certain frames of a test video. The table demonstrates the good results of the proposed method in comparison with the DEI approach for accuracy, performance rate, and computing resource consuming. The contour point quantity was reduced more than 1.5 times.

Conclusions

The proposed method of complicated shapes estimation for objects analysis in a video surveillance system consists of the following stages: color space conversion and normalization, object shape detection, contours detection and analysis, sorting of vector data, forming of smooth contour curve, object area computing.

When the contour points number is reduced in 1.5 times, an average error of the proposed method compared with the DEI approach for accuracy rate is 0.75%, for performance rate it is 8.43%, for resource consuming rate it is 3.09%. When the contour points number is reduced more than 2.5 times, an average error of the proposed method decreases.

The proposed method allows to define an array of vector contour points which represent an "approximate" surveillance object of complicated shape and it decreases the data volume to be used in further analysis of a motion trajectory. In addition, this method enables describing the surveillance object by an equal quantity of contour points that in turn can simplify the task of surveillance objects classification.

The further research concerns the transformation of a surveillance object subspace for the analysis of a motion trajectory, including examination of the object's behavior on a probable execution of unauthorized actions.

References

[1] D.J. Dailey and L. Li, "Video image processing to create a speed sensor", University of Washington, Res. Rep. 52, 2000.

[2] A.N. Rajagopalan and R. Chellappa, "Vehicle detection and tracking in video", in *IEEE Proc. Image Proc.*, Vancouver, BC, Canada, 2000, pp. 351–354. doi: 10.1109/ICIP.2000.900967

[3] R. Cucchiara *et al.*, "Image analysis and rule-based reasoning for a traffic monitoring system", *IEEE Tran. Intell. Transport. Syst.*, vol. 1, no. 2, pp. 119–130, 2000. doi: 10.1109/6979.880969

[4] B.L. Tseng *et al.*, "Real-time video surveillance for traffic monitoring using virtual line analysis", in *IEEE Proc. Multimedia and Expo, ICME 2002*, Lausanne, Switzerland, 2002, vol. 2, pp. 541–544. doi: 10.1109/ICME.2002.1035671

[5] V. Kastrinaki *et al.*, "A survey of video processing techniques for traffic applications", *Image and Vision Computing*, vol. 21, no. 4, pp. 359–381, 2003. doi: 10.1016/S0262-8856(03)00004-0

[6] G. Marbach *et al.*, "An image processing technique for fire detection in video images", *Fire Safety J.*, vol. 41, no. 4, pp. 285–289, 2006. doi: 10.1016/j.firesaf.2006.02.001

[7] S.E. Memane and V.S. Kulkarni, "A review on flame and smoke detection techniques in video's", *Int. J. Adv. Res. Electrical, Electronics Instrumen. Eng.*, vol. 4, no. (2), pp. 885–889, 2015.

[8] E. Cabello *et al.*, "A new approach to identify big rocks with applications to the mining industry", *Real Time Imaging*, vol. 8, no. 1, pp. 1–9, 2002. doi: 10.1006/rtim.2000.0255

[9] R.J. Ferraria *et al.*, "Real-time detection of steam in video images", *Pattern Recogn.*, vol. 40, no. 3, pp. 1148–1159, 2007. doi: 10.1016/j.patcog.2006.07.007

[10] B. Munzer *et al.*, "Content-based processing and analysis of endoscopic images and videos: A survey", *Multimedia Tools and Applications*, vol. 77, no. 1, pp. 1323–1362, 2018. doi: 10.1007/s11042-016-4219-z

[11] H.G. Moghaddam and W.H. Enright, "A scattered data interpolant for the solution of three dimensional PDEs", in *Proc. Eur. Conf. Computational Fluid Dynamics, ECCOMAS CFD 2006*, Netherland, 2006.

[12] H.G. Moghaddam and W.H. Enright, "Efficient contouring on unstructured meshes for partial differential equations", *ACM Trans. Math. Software*, vol. 34, no. 4, 2007. doi: 10.1145/1377596.1377599

[13] W.H. Enright, "Accurate approximate solution of partial differential equations at off-mesh points", *ACM Trans. Math. Software*, vol. 26, no. 2, pp. 274–292, 2000. doi: 10.1145/353474.353482

[14] G. Barequet and M. Sharir, "Piecewise-linear interpolation between polygonal slices", *Computer Vision and Image Understanding*, vol. 63, no. 2, pp. 251–272, 1996. doi: 10.1145/177424.177562

[15] S. Lee *et al.*, "Scattered Data Interpolation with Multilevel B-Splines," *IEEE Trans. Visual. Comp. Graphics*, vol. 3, no. 3, pp. 228–244, 1997. doi: 10.1109/2945.620490

[16] *Matlab Documentation* [Online]. Available: https://www.mathworks.com/help/matlab

[17] R.C. Gonzalez and R.E. Woods, *Digital Image Processing*, 2nd ed. Prentice Hall, NJ, 2002.

[18] I. Pitas and A.N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications.* Boston, MA: Kluwer, 1990, 392 p.

[19] V.V. Khryashchev *et al.*, "Image denoising using adaptive switching median filter", *IEEE Int. Conf. Image Process.*, vol. 1, pp. 117–120, 2005. doi: 10.1109/ICIP.2005.1529701

[20] Y. Zhao *et al.*, "Performance enhancement and analysis of an adaptive median filter", in *Proc. Int. Conf. Commun. Networking*, 2007, pp. 651–653. doi: 10.1109/CHINACOM.2007.4469475

[21] S. Shrestha, "Image denoising using new adaptive based median filter", *Signal & Image Processing: An Int. J.*, vol. 5, no. 4, pp. 1–13, 2014. doi: 10.5121/sipij.2014.5401

[22] P. Su and R.L.S. Drysdale, "A comparison of sequential Delaunay triangulation algorithms", *Computational Geometry*, vol. 7, no. 5-6, pp. 61–70. 1995. doi: 10.1016/S0925-7721(96)00025-9

О.С. Шкурат, Є.С. Сулема, А.І. Дичка

МЕТОД ОЦІНКИ СКЛАДНИХ ФОРМ ДЛЯ АНАЛІЗУ ОБ’ЄКТІВ У СИСТЕМАХ ВІДЕОСПОСТЕРЕЖЕННЯ

Проблематика. Оцінка форми об’єктів відеозображень є відносно складною задачею. При розв’язанні задачі геометричного подання об’єкта спостереження необхідно враховувати низку додаткових факторів: можливість накладання елементів зображення, однорідність складних елементів, однорідність елементів тла та об’єкта тощо.

Мета дослідження. Розроблення методу оцінки форми складних об’єктів для використання у системах відеоспостереження для визначення динаміки переміщення об’єкта спостереження, швидкості та характеру його руху, оцінки ймовірності виконання об’єктом несанкціонованих дій та для інших подібних задач.

Методика реалізації. Для виділення растрової форми об’єкта спостереження застосовується процедура “віднімання фону”. Для визначення контурів об’єкта векторної форми застосовується метод DEI. Для визначення опорних контурних точок і формування гладких кривих застосовуються процедури сортування.

Результати дослідження. Запропонований метод складається з таких етапів: перетворення колірного простору та нормування, визначення форми об’єкта, визначення й аналіз контурів, сортування векторних даних, формування гладких контурних кривих, визначення площі об’єкта. Середня похибка запропонованого методу при скороченні кількості точок у 1,5 разу порівняно з підходом DEI за точністю становить 0,75 %, за швидкодією – 8,43 %, за ресурсомісткістю – 3,09 %.

Висновки. Використання запропонованого методу дасть змогу визначати масив векторних контурних точок, що відображають “апроксимований” об’єкт спостереження складної форми, і, відповідно, зменшувати обсяг даних для подальшого аналізу траєкторії руху та інших подібних задач без втрати точності. Також цей метод дає можливість описувати об’єкти спостереження однаковою кількістю контурних точок, що своєю чергою може спростити задачу класифікації об’єктів спостереження.

Ключові слова: метод DEI; вилучення ознак зображення; векторна фільтрація зображення; векторний контурний аналіз.

О.С. Шкурат, Е.С. Сулема, А.И. Дичка

МЕТОД ОЦЕНКИ СЛОЖНЫХ ФОРМ ДЛЯ АНАЛИЗА ОБЪЕКТОВ В СИСТЕМАХ ВИДЕОНАБЛЮДЕНИЯ

Проблематика. Оценка формы объектов видеоизображений является относительно сложной задачей. При решении задачи геометрического представления объекта наблюдения необходимо учитывать ряд дополнительных факторов: возможность наложения элементов изображения, однородность составных элементов, однородность элементов фона и объекта и т.д.

Цель исследования. Разработка метода оценки формы составных объектов для использования в системах видеонаблюдения для определения динамики перемещения объекта наблюдения, скорости и характера его движения, оценки вероятности выполнения объектом несанкционированных действий и для других подобных задач.

Методика реализации. Для выделения растровой формы объекта наблюдения применяется процедура “вычитания фона”. Для определения контуров объекта векторной формы используется метод DEI. Для определения опорных контурных точек и формирования гладких кривых применяются процедуры сортировки.

Результаты исследования. Предложенный метод состоит из следующих этапов: преобразование цветового пространства и нормализация, определение формы объекта, определение и анализ контуров, сортировка векторных данных, формирование гладких контурных кривых, определение площади объекта. Средняя погрешность предложенного метода при сокращении количества точек в 1,5 раза по сравнению с методом DEI по точности составляет 0,75 %, по быстродействию – 8,43 %, по ресурсоемкости – 3,09 %.

Выводы. Использование предложенного метода позволит определять массив векторных контурных точек, определяющих “аппроксимированный” объект наблюдения сложной формы, и, соответственно, уменьшать объем данных для дальнейшего анализа траектории движения и других подобных задач без потери точности. Также данный метод позволяет описывать объекты наблюдения одинаковым количеством контурных точек, что в свою очередь может упростить задачу классификации объектов наблюдения.

Ключевые слова: метод DEI; выделение признаков изображения; векторная фильтрация изображения; векторный контурный анализ.

Рекомендована Радою
факультету прикладної математики
КПІ ім. Ігоря Сікорського

Надійшла до редакції
27 березня 2018 року

Прийнята до публікації
31 травня 2018 року