N. Andrushchak, B. Vasylyshyn, V. Chornenkyy
Computer-Aided Design Systems
Lviv Polytechnic National University

COMPARATIVE ANALYSIS OF ALGORITHMS FOR PROJECTED LASER LINE IDENTIFICATION AND RECOGNITION FOR 3D SCANNING DEVICES

Проведено порівняльний аналіз алгоритмів визначення та розпізнавання лінії лазерного випромінювання, описано принцип дії, негативні і позитивні сторони кожного методу, а також показано можливості їхнього застосування для пристроїв 3D-сканування. Тестування алгоритмів проведено на експериментальній установці з використанням засобів C++ і бібліотеки OpenCV. Показано, що залежно від розмірів зображення та деталізації вхідного зображення слід рекомендувати той чи інший алгоритм.

Ключові слова: медіанний метод, метод середніх квадратів, апроксимація, 3D-сканер.

This paper is devoted to the comparative analysis of algorithms for identification and recognition of the projected laser line. The principles of these methods, their negative and positive aspects and the possibilities of their implementation for 3D-scanning devices are described. Algorithms approbation was conducted on experimental setup using C++ programming language and OpenCV library. It is shown that, depending on the image size and details of the input image, one or another algorithm should be applied.

Key words: median method, least square method, approximation, 3D-scanner.

Introduction

Receiving a 3D model of a scanned object is a cumbersome task from both theoretical and practical perspectives. Over the past ten years 3D scanners have been actively spread due to their broad applications in the fields of design, medicine, reverse-engineering, architecture, building industry, and quality control [1–3]. One of the most popular on the market a laser 3D scanner that works on the principle of acousto-optical deflectors (for scanning at large distances) and triangulation method (in portable scanners) [4, 5]. The principle of laser scanner operation is that the conical laser line is projected on the scanning object and creates a unique pattern on this object, which is recorded on a camera with further processing.

The main difficulty on providing precise performance of the laser scanner is the accuracy of the laser line identification and recognition, which is projected on the scanning object [6, 7]. It is worth mentioning that the intensity distribution of the projected laser line is heterogeneous (beam intensity is mainly concentrated around its center, not on the edges) and can be described by a Gaussian distribution (see Fig. 1).
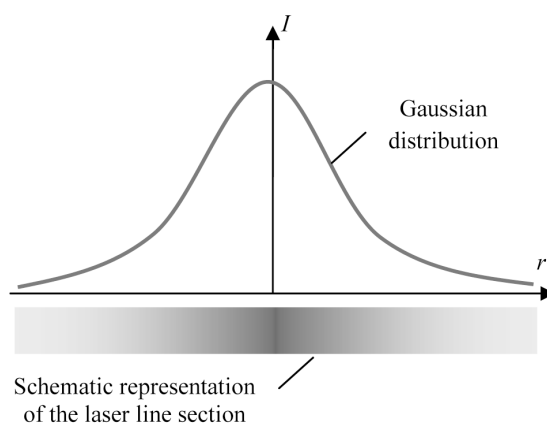

Fig. 1. Schematic representation of the laser line intensity

Another problem of accurate projected laser line identification is that the beam is 2–3 mm wide, and therefore the beam limits must not only be recognized, but also the center of beam intensity must be defined. For that purpose two methods were chosen: the median method and the approximation using polynomials (lease square method).

A median method for finding the middle of the laser beam can be used in most cases due to its simplicity of implementation and absence of conducting complex calculations [8]. The downside of this method is a lack of accuracy when applying on objects of complex shape. In addition, while implementing this method minor inaccuracies can be present. Thus, the best possible way to increase usefulness of algorithm is by applying algorithms of smoothing.

The method can be represented by the formula [8]:

$$z = \frac{\sum_{i=0}^{n} x_i}{n}, \tag{1}$$

where z is a point in the middle of the beam that is located in a particular plane; x_i is a point of the Gaussian distribution of the cross section; n is number of points in a Gaussian distribution.

As a result of all calculations of the laser projection using formula 1, it is possible to get a line that reproduces object in details. This allows to reuse the shape of the scanned object for the particular application.

An approximation method is used in order to obtain analytical dependence for describing large data arrays [9, 10]. The idea of the method is that the data array is substituted by some function that describes the tendency of data changes and provides the function of the minimum sum of squared deviations for the experimental data. The most common criterion used in solving the approximation problems is the mean squared deviation, which is a minimized sum of squared deviations of the experimental data from the analytic function Q(x). It is defined for a given set of points as

$$\sum_{i=0}^{n} (Q(x_i) - y_i)^2 \Rightarrow \min. \tag{2}$$

In this formulation, the approximation of the experimental data can have multiple solutions. To achieve a unique solution, the value of Q(x) must be set by a power law, trigonometric or orthogonal polynomial.

The positive feature of the approximation method is that it reflects accurately the curve approximation to the original data in a given interval. However, this method requires significant computing power of computers and on outside of the desired data sets the precision of this method is low. This method is very accurate, although, at low computational capabilities, it requires more time, if compared, e.g., with the median method.

Practical implementation

To perform a comparative analysis of previously described methods the experimental setup for carrying out analysis of 3D objects was used. A schematic representation of the setup is shown in Fig. 2.
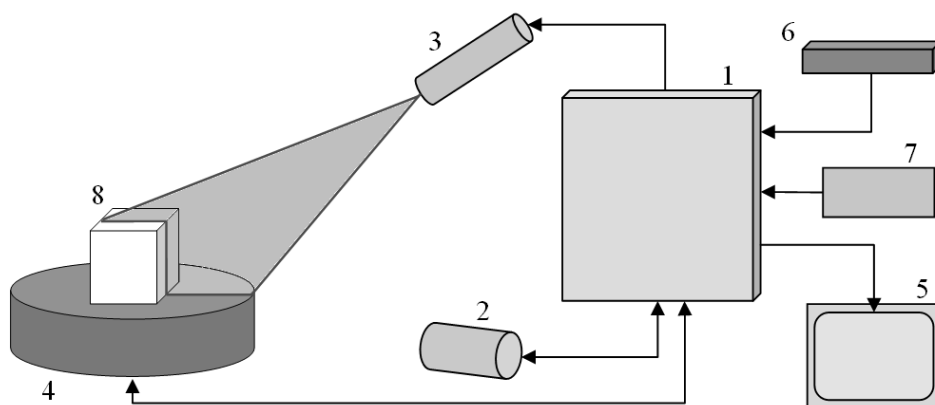

Fig. 2. Schematic representation of the experiment for 3D scanning:
1 – the device for control and analysis; 2 – a camera; 3 – a laser (λ = 650 nm);
4 – rotating device; 5 – the device for data visualization; 6 – the power supply;
7 – the input device; 8 – object of scanning

The principle of operation is based on the method of triangulation. The laser beam 3 is projected on an object of scanning 8, where the projected line on the sample shape is recorded by a camera 2 (Raspberry Pi NoIR). After that, the data is transferred from the camera to a device for control and analysis 1 (Raspberry Pi B+), where the transformation and analysis of the data are performed. The processed data are displayed on a device for data visualization 5. For a series of measurements, in order to create a 3D image of the scanned object, a rotating device 4 is used, which is guided by a control and analysis device 1. The setup is powered by a power supply 6 (the bias 5–12 V).

The principles of algorithms

The median method is based on the binary matrix that displays the image in black and white colors. For example, the one can take a binary matrix of 10x10 size, where the arch is displayed (see Fig. 3):

$$
\begin{matrix}
\mathbf{0}\,\mathbf{0}\,1\,1\,1\,1\,1\,1\,1\,1 \\
1\,1\,\mathbf{0}\,\mathbf{0}\,\mathbf{0}\,1\,1\,1\,1\,1 \\
1\,1\,1\,1\,1\,\mathbf{0}\,\mathbf{0}\,1\,1\,1 \\
1\,1\,1\,1\,1\,1\,\mathbf{0}\,\mathbf{0}\,1\,1 \\
1\,1\,1\,1\,1\,1\,1\,\mathbf{0}\,1\,1 \\
1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0}\,1 \\
1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0}\,1 \\
1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0}\,1 \\
1\,1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0} \\
1\,1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0}
\end{matrix}
$$

Fig. 3. A binary matrix of 10x10 size:
"1" is a black color indication, "0" is a white color

The algorithm can operate to view the rows or columns. For this example the view on rows is chosen. We can find a series of white pixels in each row if $x_{ij} = 0$ i $x_{ij+1} \neq 1$ using the following formula:

$$
k = \sum_{i=0}^{n} x_{ij}, \tag{3}
$$

where k is the number of white pixels in the series, i a line index in matrix, j a column index, n a total number of lines, and x_{ij} a current matrix element (pixel image).

Once we see that $x_{ij} = 1$, we get a series of points and find their mean:

$$
z = \frac{k}{2}, \tag{4}
$$

where z is an index of the found point in matrix and k a number of white pixels in the series.

Using the value that was found using the formula (4) the result is stored in the resulting matrix x_{ij} – z = 0, and continue to bypass the column to the end, and afterwards move to the next line. As the result it will be possible to achieve a matrix (see Fig. 4) of the such form:

$$
\begin{matrix}
\mathbf{0}\,\mathbf{1}\,1\,1\,1\,1\,1\,1\,1\,1 \\
1\,1\,\mathbf{1}\,\mathbf{0}\,\mathbf{1}\,1\,1\,1\,1\,1 \\
1\,1\,1\,1\,\mathbf{1}\,\mathbf{0}\,\mathbf{1}\,1\,1\,1 \\
1\,1\,1\,1\,1\,1\,\mathbf{0}\,1\,1\,1 \\
1\,1\,1\,1\,1\,1\,1\,\mathbf{0}\,1\,1 \\
1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0}\,1 \\
1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0}\,1 \\
1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0}\,1 \\
1\,1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0} \\
1\,1\,1\,1\,1\,1\,1\,1\,1\,\mathbf{0}
\end{matrix}
$$

Fig. 4. A binary matrix of 10x10 size
*after applying the algorithm*

From the results that are depicted in Fig. 4 it is possible to conclude that using the median method minimal distortion lines and gaps in the line can be presented.

The method of polynomial approximation is based on the correct decision of the polynomial. For this experiment the polynomial approximation of power law was used. In solving this problem, it comes down to determining the polynomial coefficients $\{a_0, a_1, a_2, ..., a_n\}$. In this case, the average deviation criterion can be presented using the form:
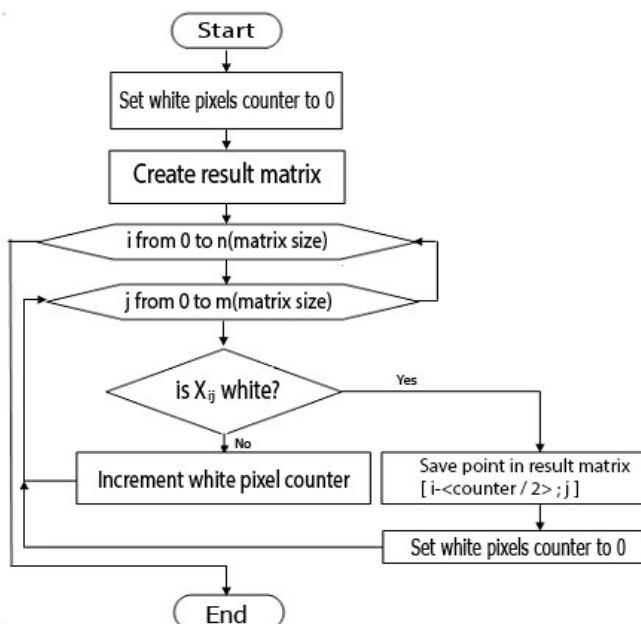
$$\sum_{i=0}^{n}(Q(x_i) - y_i)^2 = \sum_{i=0}^{n}(a_0 + a_1 x + a_2 x^2 + ... + a_m x^m - y_i)^2 \Rightarrow \min. \tag{5}$$

In order to make an approximation, we need to form and solve the matrix (see Ref. [9])

$$
\begin{bmatrix}
n & \sum_{i=0}^{n} x_i & \sum_{i=0}^{n} x_i^2 & ... & \sum_{i=0}^{n} x_i^m \\
\sum_{i=0}^{n} x_i & \sum_{i=0}^{n} x_i^2 & \sum_{i=0}^{n} x_i^3 & ... & \sum_{i=0}^{n} x_i^{m+1} \\
\sum_{i=0}^{n} x_i^2 & \sum_{i=0}^{n} x_i^3 & \sum_{i=0}^{n} x_i^4 & ... & \sum_{i=0}^{n} x_i^{m+2} \\
... & ... & ... & ... & ... \\
\sum_{i=0}^{n} x_i^m & \sum_{i=0}^{n} x_i^{m+1} & \sum_{i=0}^{n} x_i^{m+2} & ... & \sum_{i=0}^{n} x_i^{2m}
\end{bmatrix}
\cdot
\begin{bmatrix}
a_0 \\ a_1 \\ a_2 \\ ... \\ a_m
\end{bmatrix}
=
\begin{bmatrix}
\sum_{i=0}^{n} y_i \\
\sum_{i=0}^{n} x_i y_i \\
\sum_{i=0}^{n} x_i^2 y_i \\
... \\
\sum_{i=0}^{n} x_i^m y_i^m
\end{bmatrix}. \tag{6}
$$
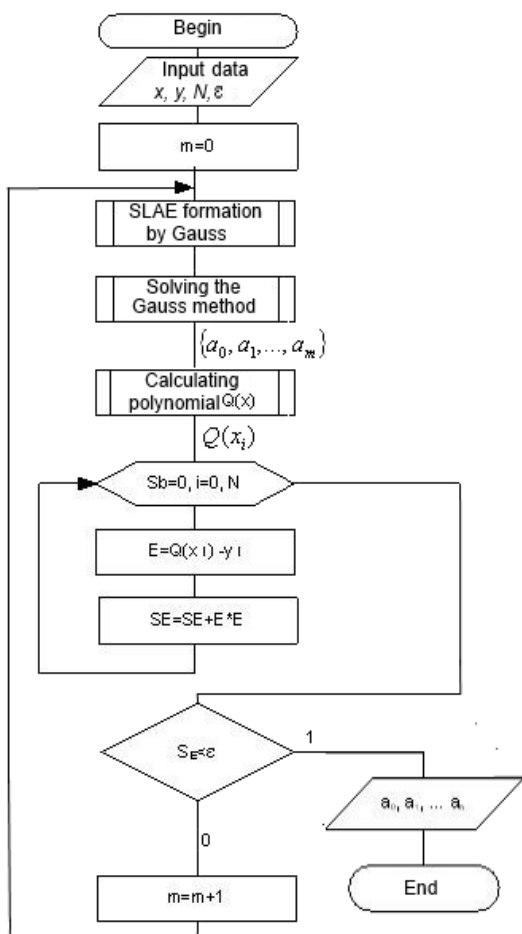
Fig. 5. Block-diagrams of algorithms:
a – approximation by least squares method; b – median method

In order to solve this matrix, a Gaussian method is used until the equation (7) becomes true:

$$\sum_{i=0}^{n}(Q(x_i) - y_i)^2 \leq \varepsilon. \tag{7}$$

Once the equation (7) is solved the coefficients $\{a_0, a_1, a_2, ..., a_n\}$ of the approximating polynomial will be calculated.

Stages of program execution

For implementation of our algorithms an OpenCV library is used [11]. The OpenCV (Open Source Computer Vision Library) is a library of functions and algorithms for computer vision, image processing and numerical algorithms for general purpose with open source. It can be freely used in academic and commercial purposes because of BSD license.

The algorithms were developed and tested in 2013 IDE Visual Studio on Windows, then was adapted for use on Linux family operating systems to run on the Raspberry Pi B+ type.

The necessary task to be realized within the median method consisted in finding the middle line of the laser beam projected on the scanning object. The software function in C++ coding that implements the median method is presented below. The only parameter for the function is a binary matrix created by means of OpenCV library. The result of the algorithm is the output matrix with an identified and recognized line, which is saved as a file with the coordinates as data arrays and image, which is visualized on the screen. The function has two cycles that implement step-by-step matrix observation by rows and columns. Each pixel in the image is verified on the condition of equality. If the pixel appears to be white, then the white pixel counter increases. When the algorithm finds a black pixel it is checked on the noise level, which is set in the calibration of the scanner. When the number of white pixels is greater than the noise, the middle of the series need to be found and is subtracted from the current iteration position. Afterwards the counter is set to zero. As a result, using the median method the middle of the projected laser beam can be found.

Code of function that implements the median method:

```
Mat lineMedianSelection(Mat source){
        int white_counter = 0; // Number of whites
        int noise_coef = 4;
        Mat result(source.rows, source.cols, CV_8U);
        for (int j = 0; j < source.cols; j++){
                for (int i = 0; i < source.rows; i++){
                        if ((int)source.at<uchar>(i, j) == 255){ // is it white?
                                white_counter++;
                        }
                        else if (white_counter > noise_coef){
                                result.at<uchar>((i – white_counter / 2), j) = 0;
                                white_counter = 0;
                        }
                } // for(j) end
                white_counter = 0;
        } // for(i) end
        return result;
}
```

The principle of projected laser line allocation using the least squares method is as follows. This function is implemented in the construction and solution matrix. It is obtained from the matrix with data obtained by image processing. Finally, we get a polynomial curve shown in Fig. 6.

Comparative analysis of laser lines recognition methods

Comparative analysis using two methods was performed using the experimental setup, a schematic representation is shown in Fig. 2. For testing and data analysis the specially designed software with necessary set of functions was used.

Using a digital camera the image for investigation, which is presented in Fig. 4, was observed. After that the Canny edge algorithm was applied for identification of the projected line on the scanning object [12]. As a result, a line of 2–3 mm was received, depending on the viewing angle and camera position. Next, the two methods were used for projected laser line approximation: method of polynomial approximation (lease squared method) method and median method, respectively (see Fig. 6, b, c).

In Fig. 6, d a comparison of two lines that were received from both algorithms are depicted. As can be seen from Fig. 6, d two methods are accurately reproducing the laser line, despite some inaccuracies in the representation of straight lines.

For example, for more precise calculation of the middle point on the curve using the median method the angle of rotation of the scanning object was taken into account, also including mathematical model of the developed experimental setup. For the method of least squares, a significant inaccuracy may be made through the wrong choice of approximating polynomial. This inaccuracy can be corrected, but the time for this algorithm will increase that negatively influence the performance of the algorithm in real time, as well as in analyzing a very large number of images.
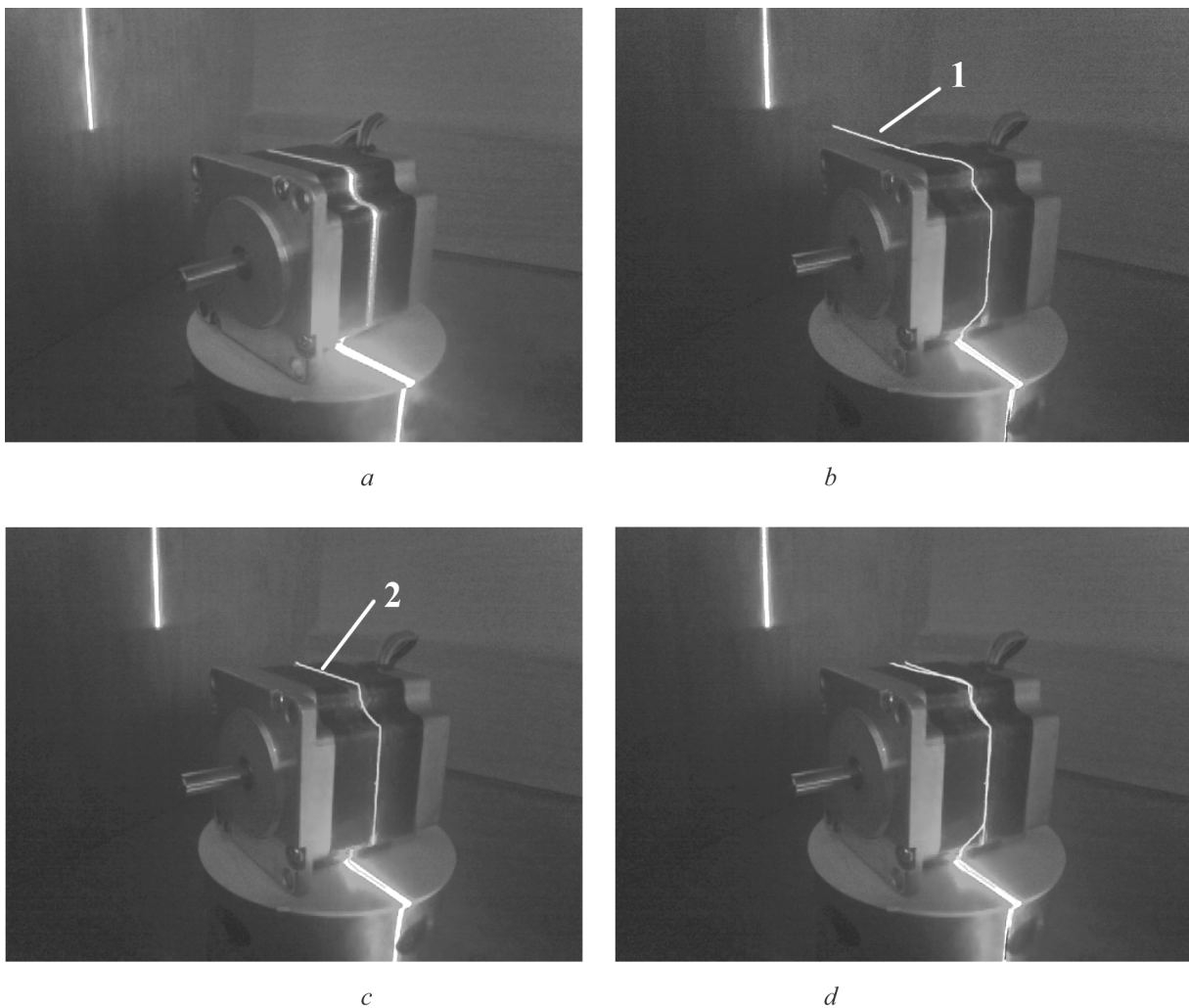
Fig. 6. The image that was analyzed:
a – input image from the camera; b – the input image with the line from method
of least squares (line 1); c – the input image with the line from median method (line 2);
d – the input image with the lines from two methods

In Table 1 the basic parameters of algorithms execution that critically affect their proper operation are presented. The main parameters that have been allocated are the size of the input image, the corresponding performance of the algorithm (duration of the algorithm) and the accuracy of algorithm.

The level of acceptable noise for the median method is set statically, but it can be described as manual tuning algorithm on the first run, depending on the scanning object lighting and image quality of those images to be used for analysis. For the method of approximation, the noise level is characterized by the complexity of the scanning object surface (are there bands, sharp curves, holes) and the power of the weighting coefficients and correlation coefficient.

Comparison of methods of laser line approximation

	Median method	Polynomial approximation method
The size of the input image, px	1936×2592	1936×2592
Duration of the algorithm, ms	25	1324
The accuracy of determination	1/1024	/2

Conclusions

In this paper a comparative analysis of the algorithms used for identification and recognition of the projected laser line is performed. The methods analyzed can be applied for 3D scanning devices, thus determining possible deficiencies on the early stages of production, finding inconsistencies and other parts of the workpieces. Each of the investigated methods is described in detail and their main advantages and disadvantages are demonstrated, along with their block-diagrams and the relevant working formulas.

The algorithms are tested using a specially developed experimental setup, which is the first prototype of our future 3D scanning device. In addition, the software implementation of these methods by means of the programme language C++ and the OpenCV library are presented.

Finally, we have shown that, depending on the size of image and complexity of the input image, one has to adopt one or another algorithm. Moreover, the median method turns out to be the best whenever it comes to the performance of the algorithm though does not require high accuracies. If it is necessary to consider the details and structures of bends on the image, then it is necessary to use the least squares method.

References

1. Gonzalez R. C. Digital Image Processing / Gonzalez R. C. & Woods R. E. – New York: Addison Wesley, 1992. 2. Sonka M. Image Processing Analysis and Machine Vision / Sonka M., Hlavac V., Boyle R. – Ch. 6. – Second ed. International Thomson Computer Press, 1999. 3. Vernon D. Machine Vision / D. Vernon. – Ch. 5 – Prentice-Hall, 1991. 4. Vosselman G. Airborne and Terrestrial Laser Scanning / Vosselman G., Maas H-G. – Dunbeath: Whittles Publ., 2010. 5. Franca J. A 3d scanning system based on laser triangulation and variable field of view / Franca J., Gazziro M., Ide A., Saito J. // IEEE International Conference on Image Processing, ICIP 2005. – Vol. 1, P. 425–428. 6. Cao F. Extracting meaningful curves from images / Cao F., Musé P., Sur F. // J. Math. Imaging Vis. 22(2), 2005. – P. 159–181. 7. Lee Y.-S. A straight line detection using principal component analysis / Lee Y.-S., Koo H.-S. and Jeong C.-S. // Pattern Recognition Letters 27(14), 2006. 8. Neter J. Applied Linear Statistical Models / Neter J., Kutner M. H., Nachtsheim C. J. and Wasserman W. – 4th Ed. – Boston: McGraw Hill, 1996. 9. Casella G. Statistical Inference // G. Casella and R. Berger. – 2nd edition. – Pacific Grove:Duxbury Advanced Series, 2002. 10. Moore D. Introduction to the Practice of Statistics / D. Moore and G. McCabe // W. H. Freeman and Co. – London, 2003. 11. Bradski G. Learning OpenCV: Computer Vision with the OpenCV Library / Bradski G., Kaehler A. – Cambridge: O'Reilly, 2008. 12. Canny J. F. A computational approach to edge detection / Canny J. F. // IEEE Trans. Pattern Anal. Mach. Intell. – 1986. – 8 (6). – P. 679–698.