10.24425/acs.2018.124709

Archives of Control Sciences
Volume 28(LXIV), 2018
No. 3*, pages 401–418*

Definition of dynamic characteristics of pointer measuring devices on the basis of automatic indications determination

VOLODYMYR KUCHERUK, IGOR KURYTNIK, PAVEL KULAKOV, ROMAN LISHCHUK, YULIA MOSKVICHOVA and ANNA KULAKOVA

The article presents the method and algorithm of automatic pointer measuring devices (voltmeter, manometer, metronomes etc.) indications determination in order to determine their dynamic characteristics with the help of web-camera and personal computer. The results of testing and experimental research of developed tool for determining the dynamic characteristics of pointer measuring devices are given. Using this method, the algorithm and the software developed, the process of determining the dynamic characteristics of the pointer measuring devices was automated. The time of recognition and calculation of one measured value for a dual-core processor and webcam with a resolution of 0.3 Mp averages 250–330 ms.

Key words: pointer measuring devices, binarization, skeletonization, recognition, dynamic characteristics

1. Introduction

Due to the development of information technology, computer vision systems are widely used in various fields of industry as they have a simple design, high reliability and a relatively low price. At the present time a large variety of pointer measuring devices is produced serially, they allow to visually observe the results of various technological processes parameters measuring.

Volodymyr Kucheruk, Pavel Kulakov and Anna Kulakova are the authors with Department of Metrology and Industrial Automation, Faculty of Computer Systems and Automation, Vinnytsia National Technical University, 95 Khmelnytske shose, Vinnytsia, Ukraine. E-mails: vladimir.kucheruk@gmail.com, kulakovpi@gmail.com, ashnei@icloud.com

Igor Kurytnik is the author with Department of Management and Production Engineering, State Higher School in Oswiecim, Kolbego, 8 str., Oswiecim, Poland. E-mail: ikurytnik@outlook.com

Roman Lishchuk is the author with Department of Tourism and Hotel and Restaurant Affairs, Faculty of Management, Uman National University of Horticulture, 1 Instytutska str., Uman, Ukraine. E-mail: roma0lir@gmail.com

Yulia Moskvichova is the author with Department of Musicology and Instrumental Training, Faculty of Preschool, Elementary Education and Arts, Vinnytsia State Pedagogical University, 32 Ostrozky str., Vinnytsia, Ukraine. E-mail: julia.moskvichova@gmail.com

Received 12.02.2018. Revised 2.09.2018.

These measuring devices include voltmeters, ommeters, wattmeters, pressure measuring devices, thermometers, metronomes, etc. In serial production of such devices, it is necessary to control their static and dynamic metrological characteristics, which is difficult to implement without the automation of this process. In the absence of automation of this process, errors arise that are caused by the human factor, for example, the limitations of the human eye characteristics. Therefore, at present, the using of computer vision systems to determine the indications of pointer measuring devices during their production is necessary.

In article [1] the method for automatic reading of scale impressions based on computer vision is proposed, which uses the method of increasing areas to determine the zone of the circle scale and its center. After that, an advanced central projection method is used to determine the values of the circular scale by the polar coordinate system and calculate the indications on a scale. Further, Hough's transformation is used to get the direction of the pointer with its contour representation. Finally, the result of the determination of indications is obtained by comparing the placement of a pointer with marks on the scale.

In article [2], the task of automatic determining the indications on the scale of the pointer device is related to finding the pointer rotation angle. First, the area of the pointer moves image is highlighted on the display. Then the Canny's method [3] is used to find a clear pointer image. Next, the method of Hough's transformation to recognize the line is used. To improve the result of determining the indications, the minimum size and thickness of the line is controlled, these options are available for real-time adjustment.

It should be noted that the considered systems require high-precision positioning of the measuring device relative to the camera. Therefore, there is a need for lengthy and complex adjustments in determining the characteristics of each pointer device, the accuracy of adjustments is influenced by the human factor.

The existing means of automatic determining the indications of pointer measuring devices have certain drawbacks, which reduces their effectiveness. First of all, this is the need for complex manual adjustment of existing systems in determining the characteristics of each individual switch device. The main task of this article is the development of a method and corresponding algorithmic support for automatic determining the indications of pointer measuring devices and their dynamic characteristics.

2. The automatic determining the indications of pointer measuring devices

The proposed method for automatic indications determination of the pointer measuring devices is based on the identification of the device pointers image with certain indications and the subsequent determination of the angle between them. The value of the pointer measuring device indications is determined by

the relevance between the pointer angle deviation and a scale mark. The block diagram of a measuring device that implements the proposed method is shown in the Fig. 1.

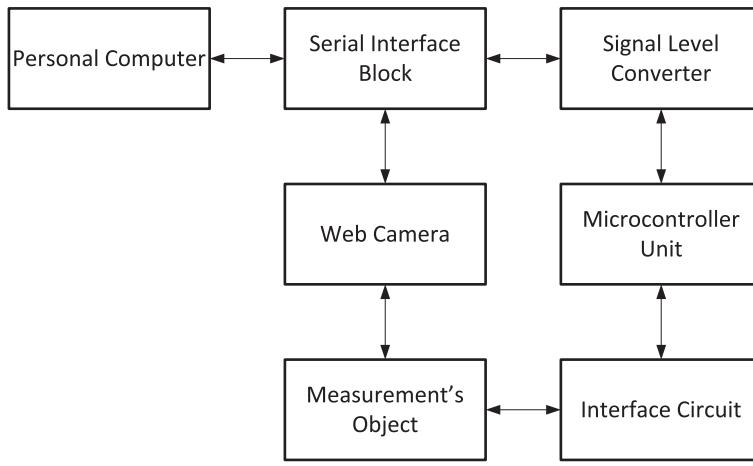

Figure 1: The block diagram of a measuring device

As an object of research, an manometer with a circular scale is used. To capture device scale images, a webcam with a resolution of 640×480 pixels and a frame rate of up to 30 frames per second is used. The received image has a 24-bit RGB model, for further processing it has been binarized.

The grayscale uses one byte (8 bit) of information for each pixel. This scale transmits 256 shades (gradations) of gray color (brightness). In this case, standard conversion algorithms are used.

Lets represent the binary image as an integral image. An integral image can be found in the presence of the function $f(x,y)$ which is a certain relationship between the coordinates of pixels and certain parameters (for example, the brightness of the pixels). In order to calculate the integral image, it is necessary to calculate for each of the rectangular areas of the image the number $I(x,y)$ which is the sum of all values $f(x,y)$ for pixels located to the left and above the pixel with coordinates (x,y). Each pixel corresponds to the expression

$$I(x,y) = f(x,y) - I(x-1,y-1) + I(x,y-1) + I(x-1,y). \tag{1}$$

In the presence of a counted integral image, the sum of the function $f(x,y)$ for any rectangular area with the upper left corner in the pixel (x_1,y_1) and the lower right angles in the pixel (x_2,y_2) can be calculated by the expression

$$\sum_{x_1}^{x_2}\sum_{y_1}^{y_2} f(x,y) = I(x_2,y_2) - I(x_2,y_1-1) - I(x_1-1,y_2) + I(x_1-1,y_1-1). \tag{2}$$

The given method of adaptive threshold processing is a simple extension of the Bernsen method [4], the main idea of which is to compare the parameters of each pixel with the average arithmetic values of the surrounding pixels parameters. In the proposed method, during the first iteration, an integral image is calculated. During the second iteration, the average brightness value in a rectangle of size $s \times s$ is calculated using the integral image for each pixel, and then the comparison is made. If the value of the current pixel is less than the average, it is set to black in binary image, otherwise white in [5]. To find the pointer deviation angle, the position of the device pointer in the initial position and with certain indications is captured (Fig. 2).

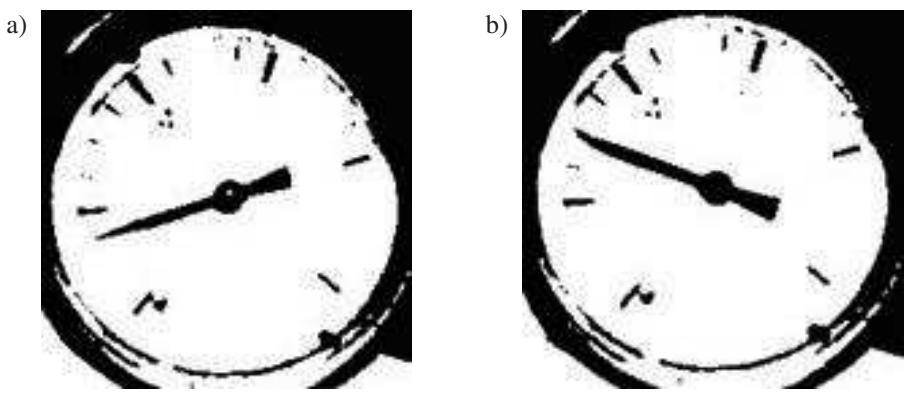

Figure 2: Photographs of the device after adaptive threshold processing; a) the image in the initial position of the pointer, b) the image with certain indications of the device

Shown on Fig. 1 images are almost identical, the difference between them only in the position of the pointer. The result of these images subtraction is a new image with clearly marked pointers (Fig. 3) and residual pixels (noise) that are eliminated by the median filter.

Median filter is a window filter [6] that slides gradually over an array of images and changes at each step one of the pixels that hit the aperture of the filter. Lets imagine a filter aperture in the form of one-dimensional array $Y = \{y_1, y_2, ..., y_n\}$, the number of its elements corresponds to the size of the window, and their location is arbitrary. As a rule, windows with an odd number n of points are used (this is automatically provided with the central symmetry of the aperture and when the central point enters its composition). If arrange the sequence $\{y_i, i = \overline{1,n}\}$ elements in ascending order, then its median will be the element that occupies the central position in this sequence. The resulting number is a filtration product for the current point of the image. The result of such processing does not depend on the sequence in which the image elements in the work array Y are represented.

Figure 3: The result of images subtraction without median filter

In [7] it was established that the aperture of the filter 3×3 retains the residual noise, and with the aperture of the filter 7×7 the objects are distorted. Optimal is a median filter with an aperture of 5×5 (Fig. 4). From Fig. 3 follows that the image of the pointers is cumbersome and it is impossible to determine with a high accuracy the angle of deviation. To solve this problem, its nessesary to narrow the pointers to 1 pixel, that is getting a skeleton representation of this image.

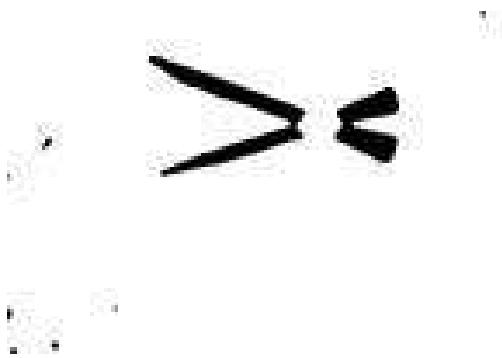

Figure 4: Images using a median filter with an aperture of 5×5

In Fig. 5 shows the skeletal image of the device pointer obtained with the help of an improved waveform method of images skeletonization.

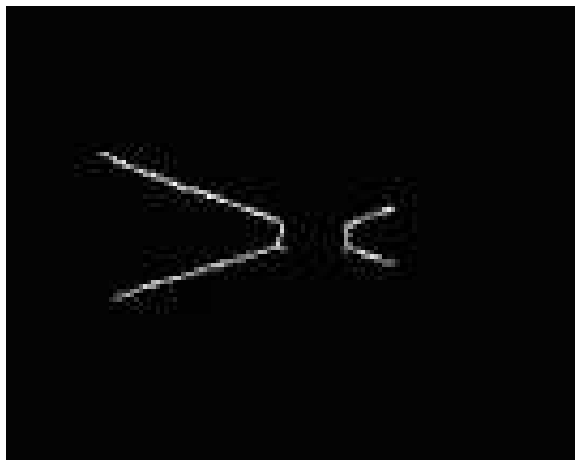

Figure 5: Skeleton representation of the device pointers

The basic idea of skeletolization was taken from the wave method [7]. The difference from the wave method is that the generation of each next circle occurs at the found point of the middle axis of the object. To solve this problem, the following algorithm is proposed:

1. We have a binary image on which there is an object.

2. Scan the image on the lines or columns to find the extreme point of the object.

3. From this point, we build a circle that is larger than the width of the object. The following is a search for the point O – the point of the middle axis of the object, and point A is searched the same way. A circle crosses an object in two points. The center of the circle and these two points form an isosceles triangle (the segments AB and AC are radius). By the property of the median of the isosceles triangle, conducted to the base of the triangle, the point O is the middle of the segment of the Sun and the point of the middle axis of the object (Fig. 6).

4. From the point O, build the next circle (repeat paragraph 3).

As a result of the algorithm, we will have points of the middle axis of the object that can be used to represent the skeleton image.

The proposed method can take into account the extension (narrowing) and the rotation of the object. It is steady to noise in the image.

Using Hough's transformation [8], the coordinates of the lines (pointers) are determined and simple geometric transformations are performed by which the

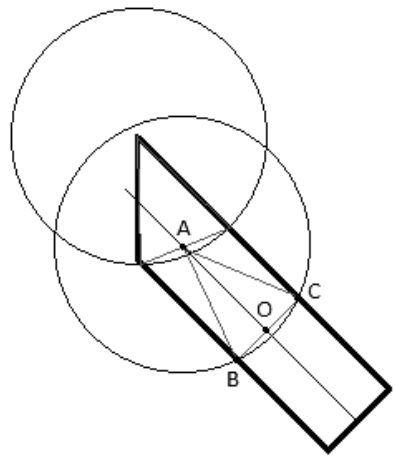

Figure 6: Schematic representation of the proposed method of image skeletalization

angle of deviation of the pointer is determined. In the simplest case, the Hough's transformation is a linear transformation for direct lines detection. The direct line can be given by the equation $y = mx + b$ and can be calculated for any pair of points in the image (x, y). Direct line $y = mx + b$ can be represented as a point with coordinates (b, m)in the parametric space. However, the vertical lines have infinite values for the parameters m and b. Therefore, it is more convenient to represent direct line in coordinates using the parameters ρ and φ. Parameter ρ is the length of the radius-vector of the point on the line which nearest to the origin, and φ is the angle between this radius-vector and the abscissa. In this way, the equation of a direct line can be written as $y = (-\cos\varphi / \sin\varphi)x + (\rho / \sin\varphi)$, or after conversion $\rho = x\cos\varphi + y\sin\varphi$. Therefore, the obvious connection of each line in the initial image in the XY plane with the unique point with coordinate (ρ, φ) in the parametric space, if $\varphi \in [0, 2\pi]$, $\rho \geqslant 0$. Through the point in the XY plane there can be infinitely many direct lines. If this point has coordinates $(x0, y0)$, then all the lines passing through it correspond to the equation $\rho(\varphi) = x_0\cos\varphi + y_0\sin\varphi$. This corresponds to a sinusoidal curve in a parametric space that is unique to a given point. If the curves corresponding to two points are superimposed on one another, then the point (in the parametric space), where they intersect, corresponds to a direct lines (in the real place of the image) which passes through both points. In the general case, the points that form a direct line determine the sinusoids that intersect at the parameters point for this line. Thus, the task of detecting collinear points can be reduced to the task of finding intersecting curves.

The Hough's transformation algorithm uses an array called the accumulator to determine the presence of a line $y = mx + b$. The accumulator size is equal to the number of unknown space parameters ρ and φ. With the help of the algo-

rithm for finding the local maximum, the elements of the accumulator with the maximum values are determined, the most appropriate lines are determined on their basis. The coordinates ρ and φ are transformed into cartesian coordinates by means of expressions $x = r\cos\varphi$, $y = r\sin\varphi$. The coordinates of the points through which the first line passes are determined by the expressions

$$x_{11} = 0; \qquad y_{11} = \frac{r_1}{\sin\varphi_1}; \qquad x_{12} = \frac{r_1}{\cos\varphi_1}; \qquad y_{12} = 0. \tag{3}$$

The coordinates of the points through which the second line passes are determined by the expressions

$$x_{21} = 0; \qquad y_{21} = \frac{r_2}{\sin\varphi_2}; \qquad x_{22} = \frac{r_2}{\cos\varphi_2}; \qquad y_{22} = 0. \tag{4}$$

To find the angle of pointer rotation it is necessary to consider the type of scale: arc or circular. Below, as an example, the arc scale is considered. For arc scale there are two cases of mutual arrangement of pointers – the pointers are in one quarter or in different quarters, as shown on Fig. 7.

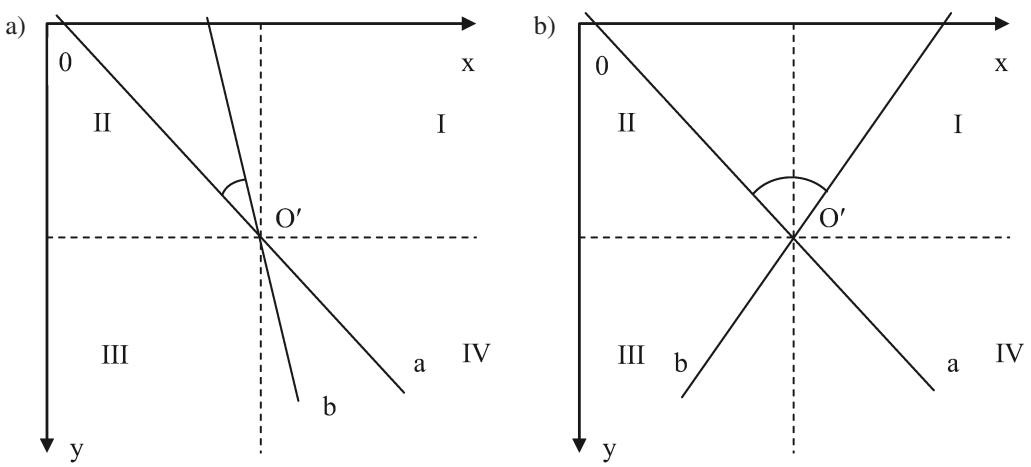

Figure 7: The position of the pointers (straight lines a and b) on the arc scale a) the pointers are in the same quarter (II); b) the pointers are in the different quarters (I and II)

Line a corresponds to the initial position of the pointer, and the line b – to the position of the pointer on a specific value of the scale. In this case, it is impossible to determine the quarters of the pointers and the angle between them. That is, if the intersection of the lines a and b coincides with the origin of coordinates at the point O', then it is not clear which line determines the initial position of the pointer. To solve this problem, the direct lines a and b, which are recognized by the Hough's transformation, are superimposed on the skeleton representation of the pointers. This process is illustrated with the help of Fig. 8.

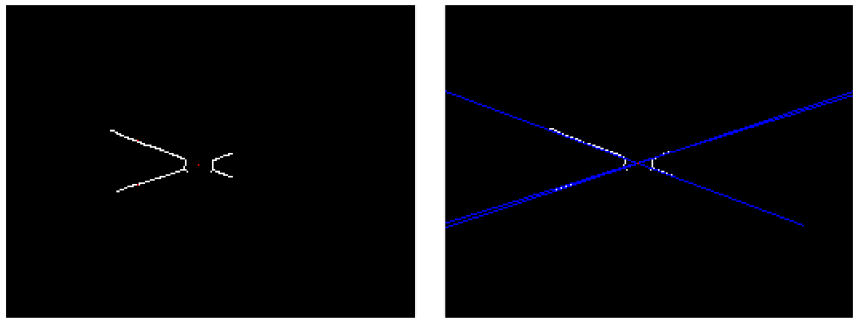

Figure 8: Demonstration of the process of lines a and b overlaying on a skeleton representation of pointers

To determine the quarters in which the pointers are located, it is necessary to find the coordinates of the lines a and b intersection point. To determine the coordinates of this point, the system of equations must be solved:

$$\begin{cases} \dfrac{x-x_{11}}{x_{12}-x_{11}} = \dfrac{y-y_{11}}{y_{12}-y_{11}}, \\ \dfrac{x-x_{21}}{x_{22}-x_{21}} = \dfrac{y-y_{21}}{y_{22}-y_{21}}. \end{cases} \tag{5}$$

As a result of the system of equations (5) solution we obtain expressions for the coordinates of the intersection lines point $O'(x,y)$.

$$\begin{aligned} x &= \frac{(x_{11}\cdot y_{12}-x_{21}\cdot y_{11})\cdot(x_{22}-x_{21})-(x_{21}\cdot y_{22}-x_{22}\cdot y_{21})\cdot(x_{21}-x_{11})}{(y_{11}-y_{12})\cdot(x_{22}-x_{21})-(y_{21}-y_{22})\cdot(x_{12}-x_{11})}, \\ y &= \frac{(y_{21}-y_{22})\cdot x-(x_{21}\cdot y_{22}-x_{22}\cdot y_{21})}{(x_{22}-x_{21})}. \end{aligned} \tag{6}$$

Further, from the point O' on the lines a and b, sections $O'O_1$, $O'O_2$, $O'B_1$, $O'B_2$, which have the same length are constructed on both sides (Fig. 9).

Next, in the vicinity of the points A_1, A_2, B_1, B_2, which has a size of one pixel, the pixels that belong to the pointers are determined. If there are more than two, it is concluded that this image is a pointer. After that, the quarter in which the pointer is located, is determined. The above processes are illustrated with the help of Fig. 10.

In determining the pointer position, the angular coefficients k_1 and k_2 of the lines a and b and their angles of inclination φ_1 and φ_2 to the OX axis are taken

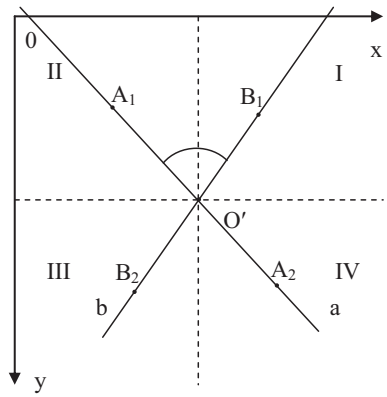

Figure 9: To determine the lines a and b intersection point coordinates

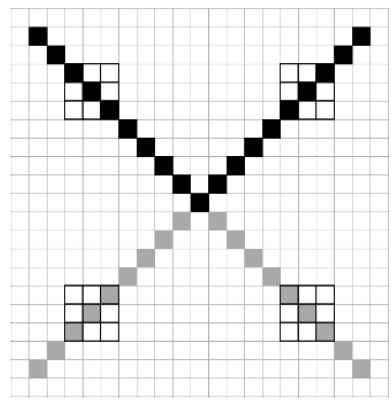

Figure 10: Determination of the quarters in which the pointers are located

into account:

$$\begin{aligned} \tan\varphi_1 = k_1 = \frac{x_{12}-x_{11}}{y_{12}-y_{11}}, \qquad \varphi_1 = \arctan\left(\frac{x_{12}-x_{11}}{y_{12}-y_{11}}\right), \\ \tan\varphi_2 = k_2 = \frac{x_{22}-x_{21}}{y_{22}-y_{21}}, \qquad \varphi_2 = \arctan\left(\frac{x_{22}-x_{21}}{y_{22}-y_{21}}\right). \end{aligned} \tag{7}$$

For this case, the following condition holds: if $A_1 \in \{\text{II}\}$ and $B_1 \in \{\text{I}\}$, $k_1 < 0$ and $k_2 \geqslant 0$, then angle is calculated by the expression $\varphi = 180^\circ - (\varphi_1 + \varphi_2)$.

Let's consider possible variants of pointer arrangement, for arc scale there are 2 variants (Fig. 11), and for a circular scale there are 5 variants (Fig. 12). The variants of the pointers position in the figures are given in accordance with their motion on the scale.

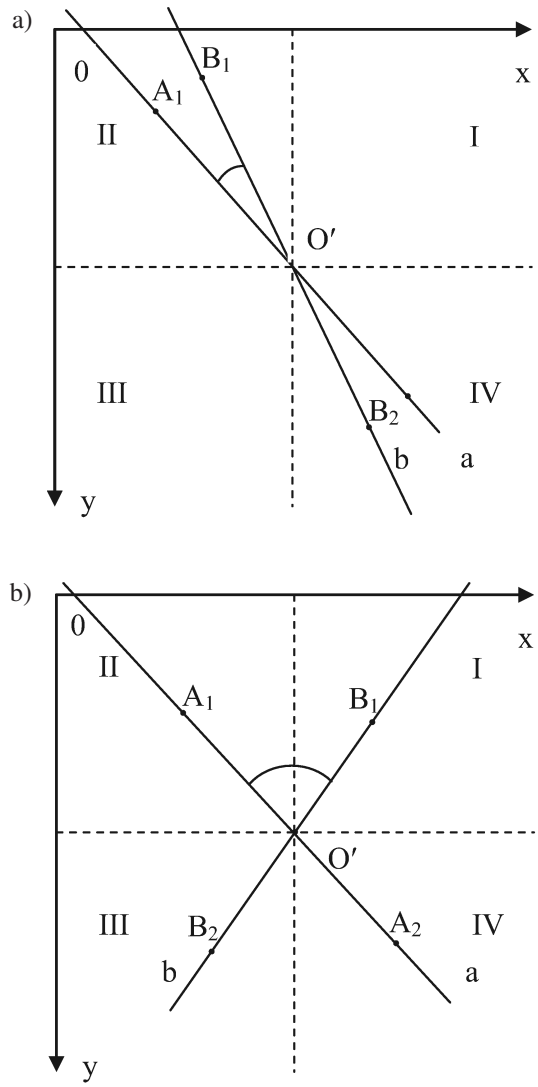

Figure 11: Variants of the pointers positions on the arc scale

Using the method described above for each of the variants on Fig. 11 and Fig. 12 an expressions are obtained, by which the angle of the pointer inclination for the arc and circular scales of the device is calculated.

For arc scale this expression is

$$\varphi = \begin{cases} ||\varphi_1| - |\varphi_2||, \ \textit{if}\, A_1 \in \{\text{II}\}\ \textit{and}\, B_1 \in \{\text{II}\},\ k_1 \geqslant 0\ \textit{and}\ k_2 \geqslant 0, \\ |180^\circ - (|\varphi_1| + |\varphi_2|)|, \ \textit{if}\, A_1 \in \{\text{II}\}\ \textit{and}\, B_1 \in \{\text{I}\},\ k_1 < 0\ \textit{and}\ k_2 \geqslant 0, \end{cases} \tag{8}$$

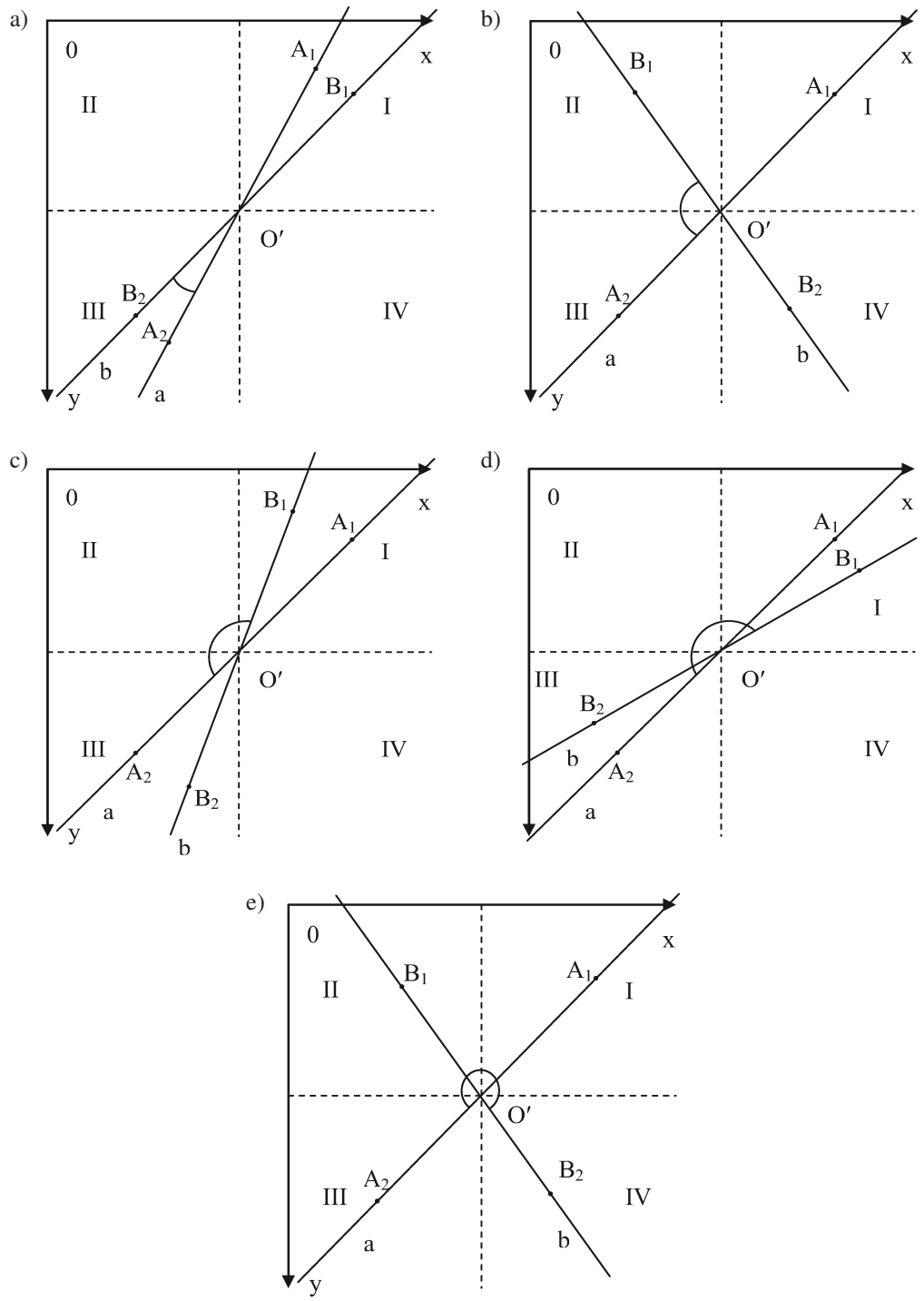

Figure 12: Variants of the pointers positions on the circular scale

for circular scale this expression is

$$\varphi = \begin{cases} ||\varphi_1| - |\varphi_2||, \text{ if } A_2 \in \{\text{III}\} \text{ and } B_2 \in \{\text{III}\}, \ k_1 < 0 \text{ and } k_2 \leqslant 0, \\ |\varphi_1| + |\varphi_2|, \text{ if } A_2 \in \{\text{III}\} \text{ and } B_1 \in \{\text{II}\}, \ k_1 < 0 \text{ and } k_2 \geqslant 0, \\ |180^\circ - |\varphi_1| + |\varphi_2||, \text{ if } A_2 \in \{\text{III}\} \text{ and } B_1 \in \{\text{I}\}, \ k_1 < 0 \text{ and } k_2 < 0, \\ |180^\circ + |\varphi_1| - |\varphi_2||, \text{ if } A_2 \in \{\text{III}\} \text{ and } B_1 \in \{\text{I}\}, \ k_1 < 0 \text{ and } k_2 < 0, \\ 180^\circ + |\varphi_1| + |\varphi_2|, \text{ if } A_2 \in \{\text{III}\} \text{ and } B_2 \in \{\text{IV}\}, \ k_1 < 0 \text{ and } k_2 \geqslant 0. \end{cases} \quad (9)$$

The Fig. 13 shows a generalized algorithm of the device for recognizing the angular position of the arrow.

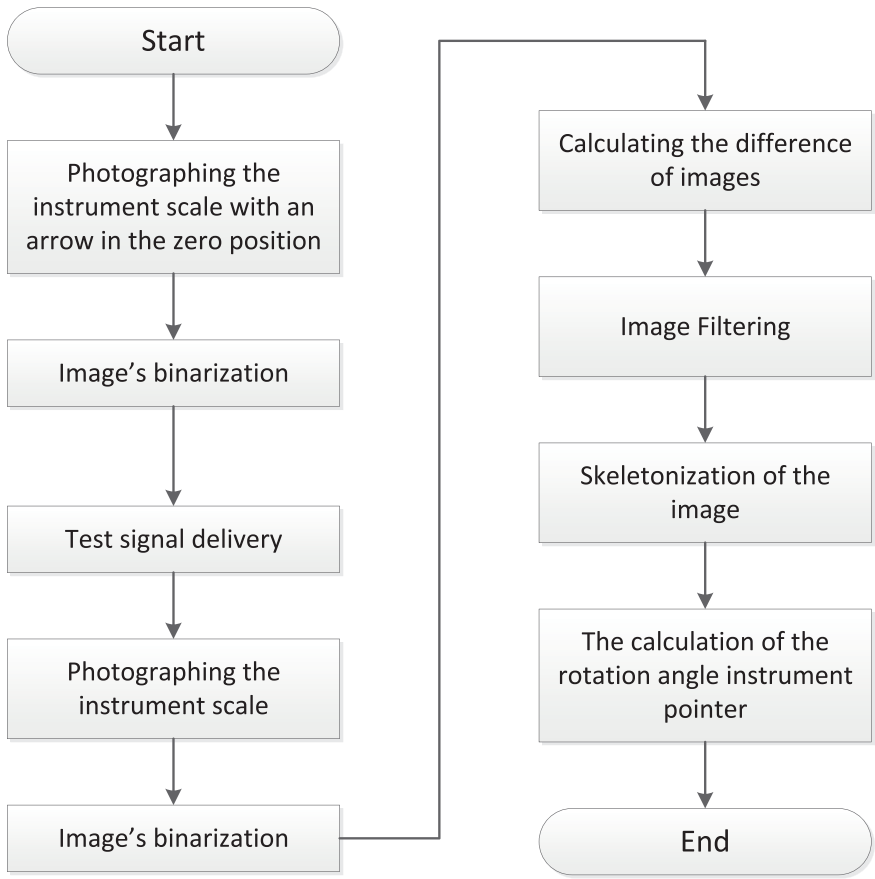

Figure 13: The figure shows a generalized algorithm of the device for recognizing the angular position of the arrow

3. Determination of dynamic characteristics of the pointer measuring devices

The pointer measuring device in the transient mode is considered as a second order oscillation link, the complex transfer function of this link is described by the expression

$$W(S) = \frac{k}{T^2S^2 + 2\xi TS + 1}, \tag{10}$$

where k – the transfer coefficient; T – time constant; ξ – relative coefficient of attenuation, $0 \leqslant \xi < 1$.

An expression defining a transient characteristic of this oscillation link is obtained by using the inverse Fourier transform of expression (10)

$$h(t) = k\left[1 - \frac{1}{\sqrt{1-\xi^2}} e^{-\frac{\xi}{T}\cdot t} \cdot \sin\left(\frac{\sqrt{1-\xi^2}}{T}t + \psi\right)\right]. \tag{11}$$

Equation (11) describes a fading oscillation process with a relative coefficient of attenuation ξ, frequency $\sqrt{1-\xi^2}/T$ and initial phase ψ. The fetched value of the function is defined as $k = \lim\limits_{t\to\infty} h(t)$, the typical function graph is shown in Fig. 14.

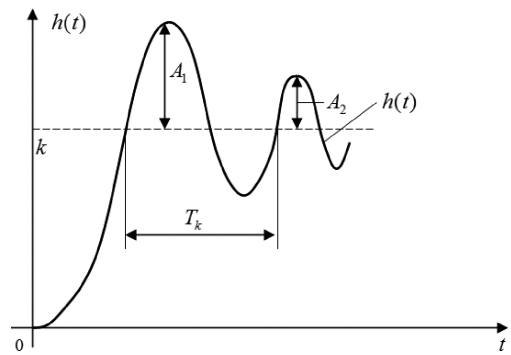

Figure 14: Fading oscillatory process; T_k – period of oscillations; A_1, A_2... – amplitudes of two adjacent oscillations

According to the graph of function $h(t)$ the parameters of the oscillation link are determined in the following way. The transfer coefficient of oscillation link is defined as the fetched value of transient characteristic k. The time constant O and the attenuation coefficient ξ are determined by expressions

$$T = \frac{T_k\sqrt{1-\xi^2}}{2\pi}, \tag{12}$$

$$\xi = -\frac{T}{T_k}\ln\frac{A_1}{A_2}. \tag{13}$$

In order to determine the dynamic characteristics of the pointer measuring devices, it is necessary to obtain a number of the device readings and a time series corresponding to the time value at the moment of the device photographing. To do this, it is not necessary to hold a series of photographs of the device indications with an interval at a certain discrete time. As a result of a discreteness step decrease, more device indications will be obtained, this will enable more accurately determine the parameters of the transient characteristic.

When photographing the device during an experimental study, the pointer on the image was blurred. Another problem was the impossibility of taking pictures at a rate of 30 frames per second. The image with a minimal exposure value was dark, but the webcam in this case was less inert. To eliminate this problem, the light source was sent to the device. During the experimental studies it was found that one frame of the image was obtained on average 15–40 ms. During cyclical photography of the device a folder with digital images of the device was created. After that, these images were recognized in cyclic mode with the help of the developed software.

As a result, an array of device indications images was received with a corresponding number of time values corresponding to the moments of receiving images. These arrays were imported into the MatLab software package to determine the dynamic characteristics of the pointer measuring devices. Using the CurveFitting package, an approximation of these values is carried out in accordance with equation (11), the result of approximation is shown on Fig. 15.

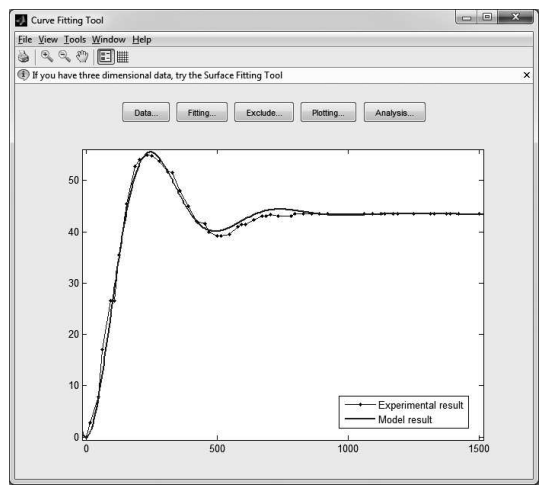

Figure 15: The result of the transient characteristics approximation

The visual analysis of the mathematical modeling results indicates a good approximation between the experimental and theoretical characteristics, which is confirmed by the statistical indicators and which values are: SSE = 10.41

(Sum of squares due to error); R-square = 0.9984; adjusted R-square = 0.9983; RMSE = 0.4706 (Root mean squared error). Parameters of the mathematical model (confidence interval with reliability 95%): $T = 72.53\ (71.92, 73.15)$; $k = 43.44\ (43.3, 43.59)$; $\xi = 0.3776\ (0.3697, 0.3855)$.

The recognized voltage values and the voltage values obtained using the model are given in Table. 1.

Table 1: The recognized voltage values and the voltage values obtained using the model.

Number	Time, [ms]	Experimental value of voltage U_e, [V]	Theoretical value of voltage U_t, [V]	Error	
				Absolute, Δ	Relative, ε, [%]
1	0	0.003	0.004	0.001	30.00
2	15	0.189	0.138	0.051	26.86
3	46	0.522	0.466	0.056	10.73
4	62	0.833	0.766	0.067	8.08
5	93	1.540	1.479	0.061	3.94
6	109	1.800	1.869	0.069	3.81
7	125	2.333	2.246	0.087	3.74
8	156	2.953	2.885	0.068	2.30
9	187	3.413	3.346	0.067	1.97
10	203	3.533	3.504	0.029	0.83
11	234	3.667	3.660	0.007	0.18
12	250	3.650	3.672	0.022	0.60
13	281	3.583	3.589	0.006	0.17
14	312	3.450	3.416	0.034	0.99
15	328	3.400	3.309	0.091	2.69
16	359	3.133	3.097	0.037	1.17
17	390	3.001	2.911	0.091	3.02
18	421	2.801	2.773	0.029	1.02
19	453	2.767	2.691	0.075	2.72
20	468	2.667	2.673	0.007	0.25
21	500	2.609	2.673	0.063	2.43
22	515	2.609	2.686	0.077	2.94
23	546	2.636	2.733	0.097	3.69
24	578	2.733	2.794	0.061	2.22
25	593	2.763	2.823	0.060	2.17
26	609	2.763	2.853	0.089	3.23
27	640	2.823	2.901	0.077	2.74
28	671	2.867	2.933	0.067	2.33
29	687	2.867	2.943	0.077	2.67
30	703	2.890	2.950	0.060	2.08
31	734	2.867	2.951	0.085	2.95
32	781	2.867	2.935	0.069	2.40
33	796	2.900	2.927	0.027	0.94
34	828	2.900	2.909	0.009	0.32
35	859	2.900	2.894	0.006	0.21

The absolute error was calculated by the expression

$$\Delta = |U_t - U_e|. \tag{14}$$

The relative error was calculated by the expression

$$\varepsilon = \frac{\Delta}{U_e} \cdot 100\%. \tag{15}$$

4. Conclusion

In the article the method and algorithm of a pointer measuring devices automatic indications determination using a webcam and a personal computer in real time are presented in order to determine the dynamic characteristics of the measuring device. The time of recognition and calculation of one measured value for a dual-core processor and webcam with a resolution of 0.3 Mp averages 250–330 ms [9]. Using this method, the algorithm and the software developed, the process of determining the dynamic characteristics of the pointer measuring devices was automated. If you use a camera with a higher framerate per second, we will get a better result and a smaller error, but the time for processing the video stream will be reduced (now about 4 frames, that is 250–300 ms, are processed in 1 second). Using a higher resolution camera does not make sense – the productivity of the software will drop.

After analyzing the statistical indicators that were obtained as a result of experimental studies, we can conclude that the work of the developed software is adequate, but the SSE is somewhat high. This is due to the high speed of the pointer at the beginning of the movement, which the webcam can not respond to, but this only lasts for the first 50–60 ms. Therefore, we recommend using a webcam with a higher frame rate. The calculated absolute maximum error is 0.097 V, and the average relative error is 3.95%.

References

[1] G.J. Chi, L. Liu, J.L., Z. Jiang and G. Zhang: Machine Vision Based Automatic Detection Method of Indicating Values of a Pointer, *Mathematical Problems in Engineering*, **2015**, (2015), Article ID 283629, 19 pages.

[2] D. Alves de Lima, G.A. Silva Pereira and F.H. de Vasconcelos: Computer Vision System to read Meter Displays, *16th IMEKO TC4 Symposium Exploring New Frontiers of Instrumentation and Methods for Electrical and Electronic Measurements*, Sept. 22–24, 2008, Florence, It, 22 August 2008, 1–5.

[3] J. CANNY: Computational Approach To Edge Detection, *IEEE Trans. Pattern Analysis and Machine Intelligence*, **8**(6), (1986), 679–698.

[4] J. BERNSEN: Dynamic Thresholding of Grey-Level Images, *Proc. of the 8th Int. Conf. on Pattern Recognition*, Paris (1986), 1251–1255.

[5] R. LISHCHUK, V. KUCHERUK and I.P. KURYTNIK: Adaptive binarization with non-uniform image illumination, *Pomiary Automatyka Robotyka*, **6**, (2014), 72–76.

[6] T. HUANG, G. YANG and G. TANG: A fast two-dimensional median filtering algorithm, *IEEE Trans. Acoust., Speech, Signal Processing*, **27**(1), (1979), 13–18.

[7] R.I. LISHCHUK and V.Y. KUCHERUK: Modified wave method of images skeletonization, *Bulletin of the Engineering Academy of Ukraine*, Kyiv, (2014), 73–77.

[8] R.O. DUDA and P.E. HART: Use of the Hough Transformation to Detect Lines and Curves in Pictures, *Comm. ACM*, **15**, (1972), 11–15.

[9] YOUTUBE: Youtube – Available at: http://youtube/A50yN6ev9Ys