UDC: 004.932 https://doi.org/10.15407/jai2023.01.155

SEARCH FOR IDENTICAL POINTS IN THE INTER-PIXEL SPACE OF VIDEO IMAGES

P. Sabelnikov[1], **Yu. Sabelnikov**[2]
[1,2] V. M. Glushkov Institute of Cybernetics of the NAS of Ukraine, Ukraine
Academik Glushkov Avenue, 40, Kyiv, 03187
pavelsabel@gmail.com
[1] https://orcid.org/0000-0003-4709-8693
[2] https://orcid.org/0000-0003-3488-7140

Abstract. One of the ways to describe objects in images is to identify some of their characteristic points or points of attention. Areas surrounding attention points are described by descriptors (a set of features) in such a way that they can be identified and compared. On these features the search for identical points on other images is carried out by scanning them with a sliding window. The most famous descriptors and methods for finding identical points are: SIFT, SURF, GLOH, BRIEF and others. This group of methods is characterized by the fact that the displacement of identical points in video images can be arbitrary, but the accuracy of calculating their coordinates depends on the bit grid of video images and, in the best case, is equal to the interpixel distance.

Another group of methods that can be used to track identical points of video images are methods built on the basis of optical flow calculation. One of the popular methods of tracking points based on optical flow calculation is the Lucas-Kanade method. It allows you to calculate the displacement of points in the interpixel space due to the solution of differential equations. To date, the Lucas-Kanade method has several modifications. A limitation of these methods is that the neighborhoods of the shifted points must overlap to a large extent.

The article investigates and proposes the complex application of methods of scanning video images with a sliding window and differential calculation of optical flow, which allows to increase the accuracy and speed of calculating the coordinates of identical points in the images in relation to the search for these points only by scanning. A more accurate calculation of the coordinates of the characteristic points of the object in the interpixel space of video images will lead to a more accurate determination of the position and orientation of these objects in 3D space.

The simulation was carried out using the method of rough search for identical points of video images described by invariant moments and specifying their coordinates using the Lucas-Kanade point tracking method. The simulation results indicate an increase in speed by almost an order of magnitude and, according to indirect estimates, the accuracy of calculations.

Keywords: images, characteristic points, identical regions, invariant moments, optical flow.

Introduction

One way to describe image objects is to identify some of their characteristic points or points of interest. Due to such points, comparison of objects is performed, and detectors are used to search and detect attention points. If the image can be represented by a set of such points of attention, then, of course, it is possible to significantly reduce both the redundancy of information required for processing and the time of image search.

To localize points of interest in an image, it is necessary to analyze the local neighborhood of pixels, giving all local features some spatial extent. In this case, the term "area" is used instead of "point of attention". The process of identifying features of the surrounding area should determine not only the position of the attention point, but also the size and possible shape of the local neighborhood. Especially in the process of geometric deformations, this significantly complicates the process due to the fact that the size and shape must be determined in an invariant way.

The regions surrounding the attention points are described by descriptors in such a way that they can be identified and compared. As a result of the construction of descriptors, a set of feature vectors is formed for the initial set of characteristic points of one of the images. Based on these features, the search for identical points on other images is carried out by scanning them with a sliding window and calculating the characteristics of the area in the window.

SIFT (Scale Invariant Feature Transform) [1] – one of the most famous descriptors, which is also a detector. It is based on the idea of calculating the histogram of oriented gradients in the neighborhood of a special point. The PCA-SIFT descriptor [2] is essentially a modification of SIFT and is built according to the same scheme, only for each special point a larger environment is considered. For the resulting set of descriptors, the dimensionality of the vectors is reduced using principal component analysis (Principal Component Analysis, PCA).

The SURF descriptor (Speeded up Robust Features) [3] also belongs to those descriptors that simultaneously search for singular points and create their description, invariant to scaling and rotation type transformations. Additionally, the keypoint search itself is invariant in the sense that the returned scene object has the same set of keypoints as the sample.

The GLOH descriptor (Gradient Location-Orientation Histogram) [4], which was built to increase reliability, is a modification of the Sift descriptor. In fact, the Sift-descriptor is calculated, but the polar grid of the neighborhood breakdown is used.

When developing the DAISY descriptor [5], the ideas of building SIFT and Gloh descriptors were used. Similarly, GLOH selects a circular neighborhood of a special point, while individual blocks are not represented by partial sectors, but by circles.

The goal of creating the descriptor BRIEF (Binary Robust Independent Elementary Features) [6] was to ensure recognition of the same areas of the image that were obtained from different viewing angles. At the same time, the task was to reduce the number of performed calculations. The recognition algorithm is reduced to the construction of a random forest (randomize classification trees) or a naive Bayesian classifier on some training set of images and the subsequent classification of test image sections. A small number of operations is provided by representing the feature vector as a binary string and using the Hamming distance as a measure of similarity. A more efficient alternative to BRIEF and SIFT descriptors is the ORB binary descriptor [7].

The work [8] presents a method in which the regions surrounding points are described using invariant moments.

This group of methods is characterized by the fact that the displacement of identical points in video images can be arbitrary, but the accuracy of calculating their coordinates depends on the bit grid of video images and, in the best case, is equal to the interpixel distance.

Another group of methods that can be used to track (search) identical points of video images are methods built on the basis of optical flow calculation.

There are several definitions of optical flow. One of them [9]: the vector field of apparent movement of objects (pixels), surfaces and edges in a visual scene between frames, caused by the relative movement between the observer (eye, camera) and the scene.

Optical flow is based on the statement that for each point of the original image $I1(x,y)$, where x and y are the coordinates of the point, one can find such a shift $(\delta x,\delta y)$ that the initial point corresponds to the point $I2(x+\delta x,\ y+\delta y)$ in the second image. In order to determine the correspondence between them, a certain function of the point is used, which does not change as a result of the shift [10]. Brightness, gradient, Hessian, Laplacian and others can be used for this.

One of the popular methods of tracking video image points based on optical flow calculation is the Lucas-Kanade method, which allows you to calculate the displacement of points in the interpixel space by solving differential equations. Today, the Lucas-Kanade method has several modifications [11].

In the Tomasi-Kanade method, movement is considered displacement and is calculated by iteratively solving the constructed system of linear equations.

The Shi–Tomasi–Kanade method takes into account affine distortions.

The Jean-Favara-Soatto method takes into account changes in illumination.

The limitation of these methods is that the shifted points must be at such a distance that their surroundings taken for analysis overlap to a large extent (more than 50%).

The purpose of this work is to investigate the possibilities of finding identical points (regions) of video images due to the complex application of the method of scanning images with a sliding window and the method of tracking points based on the calculation of optical flow. Thanks to this combination of methods, it is expected that good indicators of the accuracy of finding identical points and the speed of calculating their coordinates will be obtained in relation to finding these points only by scanning images.

For research, the authors used the method describing the surrounding of points with invariant moments and the classic Lucas-Kanade method proposed in 1981.

Algorithm for finding identical points using invariant moments

The input data for the algorithm [8] are 2 frames obtained under conditions of displacement of image objects. Algorithm steps for one point.

Stage 1.

Before carrying out the following procedures, the color images of the frames are converted to gray (halftone) and smoothed with a Gaussian filter.

Stage 2.

On the first frame, a point is set and the characteristics of its surroundings are calculated with the size of pixels, where w_size varies from 7 to 15. The central window is chosen for which w_size=10, that is, a window with a size of 21x21 pixels, which corresponds to a change in scale by one and a half times when reducing or increasing w_size.

For the surroundings of the point taken as a standard, the moments f0_et, f1_et, f2_et and the average brightness of the pixels surrounding the given point h_et and its four fragments (Fig. 1) h00_et, h01_et, h10_et, h11_et in size are calculated.

First, the geometric moments of the circles of points with the center of coordinates in the upper left corner of the image are calculated.

$$M_{x,y}^{l,k-l} = \sum_{i=x-w\_size}^{x+w\_size} \sum_{j=y-w\_size}^{y+w\_size} p_{i,j} \cdot i^{l} \cdot j^{k-l},$$

$l = (0,..., k)$,

де:

k – order of geometric moments;

x, y – coordinates of a given point;

$M_{x,y}^{l,k-l}$ – geometric moments of the k-th order, respectively further M^{00}, M^{01}, M^{10}, M^{02}, M^{20}, M^{11} for k =2;

$p_{i,j}$ – pixel brightness with coordinates i, j.

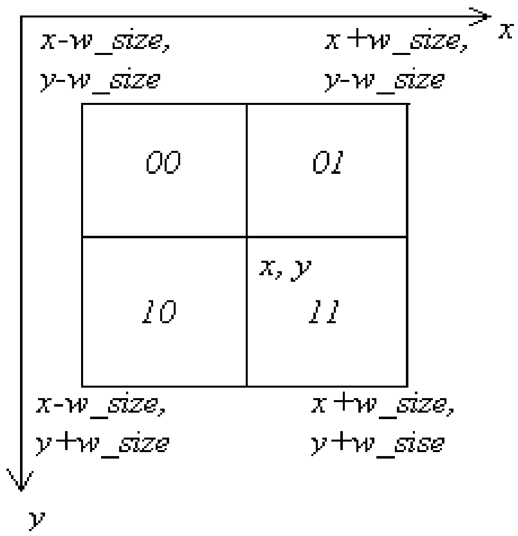

Fig. 1. The environment of a given point, divided into fragments

Then transformations are carried out, which correspond to normalization with respect to displacement, scaling and rotation.

Transformations of moments with respect to shear are defined as follows:

$$\mu_{00} = M^{00},$$
$$\mu_{01} = M^{01} - \Delta y \cdot M^{01},$$
$$\mu_{10} = M^{10} - \Delta x \cdot M^{10},$$
$$\mu_{02} = M^{02} - 2 \cdot \Delta y \cdot M^{01} + \Delta y^2 \cdot M^{00},$$
$$\mu_{20} = M^{20} - 2 \cdot \Delta x \cdot M^{10} + \Delta x^2 \cdot M^{00},$$
$$\mu_{11} = M^{11} - \Delta y \cdot M^{10} - \Delta x \cdot M^{01} + \Delta x \cdot \Delta y \cdot M^{00},$$

де Δx, Δy – the distance from the upper left corner of the image to the specified point (x, y), which is taken as the center of coordinates.

Normalization with respect to scale:

$$\eta_{l,k-l} = \frac{\mu_{l,k-l}}{\mu_{00}^{k/2+1}}.$$

The invariant Hu moments for the image region are normalized with respect to translation, scaling, and rotation:

$$f_0 = \eta_{01}^2 + \eta_{10}^2,$$
$$f_1 = \eta_{20} + \eta_{02},$$
$$f_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2.$$

Тут f_0 – invariant moment provided that (x, y) is not the center of gravity of the region.

The average brightness of the surrounding pixels and their fragments is calculated as M^{00} surroundings or its fragments, divided respectively by the number of pixels in these areas, such as surroundings

$h_{-et} = M^{00} /((2 \cdot w\_size + 1) \cdot (2 \cdot w\_size + 1))$

or a fragment with index 00 (see Fig. 1)

$h_{00\_et} = M_{00\_et}^{00} /((w\_size+1) \cdot (w\_size+1))$.

Stage 3.

For the second frame, the integral representation [8] is calculated for each of the moments of the *k*-th order $M_{x,y}^{l,k-l} = \sum_{i=0}^{x} \sum_{j=0}^{y} p_{ij} \cdot i^l \cdot j^{k-l}$, $l = (0,..., k)$,

де:

x=(w_size,…, image width - w_size -1);

y=(w_size,…, image height - w_size -1).

An integral representation is a matrix of the same size as the image size. For *k=2* there will be 6 such views, let's mark them FM^{00}, FM^{01}, FM^{10}, FM^{02}, FM^{20}, FM^{11}. $M_{x,y}^{l,k-l}$ – matrix elements that are equal to the sum of the geometric moments of individual image pixels located to the left and above the coordinate (x, y) with the center of the coordinate (0, 0) in the upper left corner.

The integral image is sequentially scanned by the central window (*w_size*=10). For each point, geometric moments of all orders are calculated from integral representations [8], as well as, as in the second stage, all invariant characteristics of the region – f_0_ob, f_1_ob, f_2_ob, *h_ob*, h_{00}_ob, h_{01}_ob, h_{10}_ob, h_{11}_ob.

The relative errors of the characteristics are calculated *delta_*, for example for f_0:

$delta\_f_0 = 1 - min(f_0\_et, f_0\_ob)/ max(f_0\_et, f_0\_ob)$, as well as integral squared error – $delta = delta\_f_0^2 + delta\_f_1^2 + delta\_f_2^2$, and the verification is carried out under one of the conditions of comparison of errors with thresholds in different combinations [8]. If the condition is met, the following parameters are saved:

min_delta=delta;
result_x=x;
result_y=y;
result_w_size= w_size.

After scanning the entire image and comparing it with the characteristics of the standards for all *w_size=7,…,15*, we get the coordinates of the identical point on the second frame, which corresponds to the specified point on the first frame (*result_x*, *result_y*), as well as the resulting window size (*result_w_size*), which indicates a change in scale.

Lucas-Kanade algorithm

The Lucas-Kanade method is used quite widely in the tasks of estimating the motion of an object [12]. It refers to local methods of optical flow calculation, as it processes pixels in the vicinity of a certain point.

This method assumes that:

a) the shift of points on the current and previous images is insignificant,

b) the displacement of points in the neighborhood of a certain point is the same,

c) pixel intensity values do not change over time:

$$I(x, y, t) - I(x + \delta x, y + \delta y, t + \delta t) = 0\,,$$

where *I (x, y, t)* − pixel intensity function with coordinates (*x, y*) in the frame *t* and ($\delta x, \delta y$) – pixel displacement between successive frames *t* and $t + \delta t$.

Suppose that $D = \{q_1, q_2, ..., q_n\}$ − a set of points around a point *P*.

Taking into account the small displacement during the linear expansion of the function for each of the points in the Taylor series, we obtain a system of equations that is solved by the method of weighted least squares. A function is used to determine the weighting factors for pixels in the image $W(x, y)$. According to this method, to find a solution, it is necessary to minimize the error:

$$\varepsilon(\upsilon) = \sum_{x,y \in D} W(x,y) \cdot [I(x,y,t) - I(x+\delta x, y+\delta y, t+\delta t)]^2,$$

$$\varepsilon(\upsilon) = \sum_{x,y \in D} W(x,y) \cdot \left(\frac{\partial I}{\partial x} \upsilon_x + \frac{\partial I}{\partial y} \upsilon_y + \frac{\partial I}{\partial t} \right)^2,$$

де $\upsilon = (\upsilon_x, \upsilon_y)$ – shear speed according to the corresponding coordinates.

To find the minimum error, it is necessary to equalize it to zero $\frac{\partial \varepsilon(\upsilon)}{\partial \upsilon_x}$, $\frac{\partial \varepsilon(\upsilon)}{\partial \upsilon_y}$.

As a result, we get an equal-nannies system:

$$\begin{cases} \sum_{x,y \in D} W(x,y) \cdot [\left(\frac{\partial I}{\partial x} \right)^2 \cdot \upsilon_x + \frac{\partial I}{\partial x} \cdot \frac{\partial I}{\partial y} \cdot \upsilon_y + \frac{\partial I}{\partial x} \cdot \frac{\partial I}{\partial t}] = 0 \\ \sum_{x,y \in D} W(x,y) \cdot [\frac{\partial I}{\partial x} \cdot \frac{\partial I}{\partial y} \cdot \upsilon_x + \left(\frac{\partial I}{\partial y} \right)^2 \cdot \upsilon_y + \frac{\partial I}{\partial y} \cdot \frac{\partial I}{\partial t}] = 0 \end{cases},$$

These equations can be presented in matrix form $A \cdot \upsilon + B = 0$,
where

$$A = \begin{bmatrix} \sum_{x,y \in D} W(x,y) \cdot \left(\frac{\partial I}{\partial x} \right)^2 & \sum_{x,y \in D} W(x,y) \cdot \left(\frac{\partial I}{\partial y} \frac{\partial I}{\partial x} \right) \\ \sum_{x,y \in D} W(x,y) \cdot \left(\frac{\partial I}{\partial x} \frac{\partial I}{\partial y} \right) & \sum_{x,y \in D} W(x,y) \cdot \left(\frac{\partial I}{\partial y} \right)^2 \end{bmatrix},$$

$$\upsilon = \begin{bmatrix} \upsilon_x \\ \upsilon_y \end{bmatrix}, \quad B = \begin{bmatrix} \sum_{x,y \in D} W(x,y) \cdot \left(\frac{\partial I}{\partial x} \frac{\partial I}{\partial t} \right) \\ \sum_{x,y \in D} W(x,y) \cdot \left(\frac{\partial I}{\partial y} \frac{\partial I}{\partial t} \right) \end{bmatrix}.$$

Accordingly $\upsilon = -A^{-1} \cdot B$.

If we take the interval between the reception of the previous and current frames as a unit of time, then (υ_x, υ_y) – displacement of a point along the corresponding coordinates, which is not an integer. That is, the coordinates are calculated in the interpixel space. The calculation of the final coordinates of the point for the current frame is carried out by an iterative procedure with the determination of intermediate displacements of the point and its coordinates. Values of pixel intensities surrounding the initial, final and intermediate points are interpolated. The iterative procedure ends when the maximum number of iterations or the amount of displacement is reached $\sqrt{\left(\upsilon_x^2 + \upsilon_y^2\right)}$ less than the specified accuracy.

This algorithm is simple and fast, so it is quite effective in many cases.

Simulation results

As mentioned earlier, the research was carried out using the scanning method [8] for a rough search for identical points described by invariant moments and specifying their coordinates using the Lucas-Kanade (L-K) method.

Two separate frames with a size of 640x360 pixels from the video sequence [13] were taken for research. Points were set on Frame 1, and their search was carried out on Frame 2.

In this experiment, a condition was used to coarsely search for points in frame № 2 [8]:

if (delta_ f_1<min_delta
&& delta_ f_0<0.3
&& delta_ f_2<0.3
&& delta_h<0.01
&& delta_h_00<0.05
&& delta_h_01<0.05
&& delta_h_10<0.05
&& delta_h_11<0.05).

All pixels of the video image taken as the center of the sliding window with the size of 21x21 pixels were scanned and the parameter was minimized f_1.

In fig. 2 presents video images with the numbers of pairs of adjacent points specified in frame 1 and found by the scanning method in frame 2.

Fig. 2. Video images with points set on frame 1 and found by scanning on the frame 2.

To specify the coordinates of the points, the L-K algorithm was applied with some changes and additions, namely:

1) The surroundings of the points on Frame 1 were taken as standards, the refinement began with the coordinates of the points obtained as a result of the preliminary scanning of Frame 2.

2) The condition for the end of the iterative process for each point separately has been strengthened.

– *if (previous offset of the point <0,01) && (current offset of the point <0,01) && (the vector sum of the previous and current displacements of the point <0,01) {КЗ=1};*

– *if (the vector sum of the previous and current displacements of the point <0,01) {КЗ=2};*

– *if (the specified maximum number of iterations is reached) {КЗ=3}.*

1) The second check means the beginning of the oscillating process, the short circuit is the code for the end of the iterative process.

2) 1) An error check was performed at each iteration.

$$Eps(\upsilon) = \sum_{x,y\in D} W(x,y)\cdot[I(x,y,t) - I(x+\delta x, y+\delta y, t+\delta t)]^2$$

The result was taken as the coordinates of the point obtained on the iteration in which min (errors) was reached.

The results of this experiment are presented in Table 1. Short circuit = 1 was obtained for all points.

Table 1. Results of the experiment of scanning frame 2 without thinning and refinement of point coordinates by the L-K algorithm with the condition № 2.

№ points	x/y points Frame1	x/y points Frame2 Scan.	x/y points Frame2 specification of L-K *min (Eps)*	Quantity. *Eps*/ № *min (Eps)*
1.1	196/62	92/61	92,105/61,330	4/4
1.2	197/61	94/60	93,112/60,315	5/5
2.1	219/107	114/106	114,828/106,495	5/5
2.2	220/106	116/105	115,915/105,505	5/2
3.1	446/186	342/185	342,146/184,687	5/5
3.2	447/185	343/184	343,146/183,638	4/4
4.1	475/92	371/90	370,872/90,545	5/5
4.2	474/91	370/90	369,868/89,540	5/2
5.1	491/31	386/29	386,659/29,150	4/4
5.2	492/30	387/28	387,658/28,160	4/4
6.1	609/132	506/130	504,841/130,278	6/6
6.2	608/131	504/132	503,877/129,262	6/6

A comparison of the difference in the coordinates of a pair of neighboring points is taken as an indirect indication of the achieved accuracy of the coordinate calculation (for example $x_{1.2}$ - $x_{1.1}$, and $y_{1.2}$ - $y_{1.1}$) in frame 1 and frame 2. In order to reduce the influence of the change in the angle of shooting frames, pairs of adjacent points with a coordinate difference on Frame 1 equal to 1 were taken. The deviation of the coordinate difference of adjacent points from unity obtained as a result of the search, in our opinion, may indicate the achieved accuracy of their calculation . The result of the comparisons is presented in Table 2.

Table 2. Results of calculations of the difference in the coordinates of neighboring points when scanning without thinning.

№ pairs of points	x_2-x_1/ y_2-x_1 pairs of points Frame1	x_2-x_1/ y_2-x_1 pairs of points Frame2 Scan.	x_2-x_1/ y_2-x_1 pairs of points Frame2 specification L-K *min Eps*
1	1/-1	2/-1	1,007/-1,015
2	1/-1	2/-1	1,087/0,99
3	1/-1	1/-1	1,0/-1,049
4	-1/-1	-1/0	-1,004/-1,005
5	1/-1	1/-1	0,999/-0,99
6	-1/-1	-2/2	-0,964/-1,016

As can be seen from Table 2, the difference between the coordinates of the points of Frame 1 and Frame 2 after refinement by the L-K algorithm is less than 0.1 part of the interpixel distance.

Acceleration of the search for identical points

If as a result of scanning you need to approximate the coordinates of identical points, you can move the sliding window by several pixels to speed it up. Table 3 presents the results of the experiment for the case of scanning frame 2 with horizontal and vertical cutting through 3 pixels. At the same time, the conditions for comparing the characteristics of the standard and the characteristics calculated in the sliding window were relaxed, namely:

if (delta_ f_1<min_delta
&& delta_ f_0<0.4
&& delta_ f_2<0.4
&& delta_h<0.015
&& delta_h_00<0.06
&& delta_h_01<0.06
&& delta_h_10<0.06
&& delta_h_11<0.06).

Table 4 presents the results of calculating the difference in the coordinates of neighboring points when scanning with thinning by 3 pixels. From the data in this table, it can be seen that the difference between the coordinates of the points of Frame 1 and Frame 2 after refinement by the L-K algorithm is also less than 0.1 part of the inter-pixel distance. Taking into account the insignificant time used by the L-K algorithm, the calculation is accelerated by almost an order of magnitude compared to the algorithm with a check for the identity of all points of Frame 2.

Conclusions

As a result of the conducted research, a complex application of the methods of scanning video images with a sliding window and differential calculation of optical flow is proposed, which allows to increase the accuracy and speed of calculating the coordinates of identical points in the images in relation to the search for these points only by scanning.

Table 3. Results of the experiment of scanning frame 2 with thinning and refinement of point coordinates by the L-K algorithm with a softened condition.

№ points	x/y points Frame1	x/y points Frame2 Scan.	x/y points Frame2 specification L-K *min (Eps)*	Quantity Eps/ № *min (Eps)*
1.1	196/62	91/61	92,105/61,330	5/3
1.2	197/61	94/61	93,112/60,315	5/5
2.1	219/107	115/106	114,828/106,495	5/2
2.2	220/106	115/106	115,915/105,505	5/5
3.1	446/186	343/184	342,146/184,687	5/4
3.2	447/185	343/184	343,146/183,638	4/4
4.1	475/92	373/91	370,874/90,545	7/4
4.2	474/91	370/91	369,866/89,539	6/3
5.1	491/31	385/28	386,659/29,150	5/5
5.2	492/30	388/28	387,658/28,160	4/2
6.1	609/132	505/130	504,844/130,277	5/2
6.2	608/131	502/130	503,877/129,262	6/6

Table 4. Results of calculations of the difference in the coordinates of neighboring points when scanning with thinning by 3 pixels.

№ pairs of points	x_2-x_1/ y_2-x_1 pairs of points Frame1	x_2-x_1/ y_2-x_1 pairs of points Frame2 Scan.	x_2-x_1/ y_2-x_1 pairs of points Frame2 specification L-K *min Eps*
1	1/-1	3/0	1,007/-1,015
2	1/-1	0/0	1,087/0,99
3	1/-1	0/0	1,0/-1,049
4	-1/-1	-3/0	-1,008/-1,006
5	1/-1	3/0	0,999/-0,99
6	-1/-1	-3/0	-0,967/-1,015

A more accurate calculation of the coordinates of the characteristic points of the object in the interpixel space of video images will lead to a more accurate determination of the position and orientation of these objects in 3D space.

Modeling, using the example of using the method of rough search for identical points described by invariant moments and specifying their coordinates using the Lucas-Kanade point tracking method, confirmed the conclusions about an increase in speed by almost an order of magnitude and, according to indirect estimates, the accuracy of calculations.

Research should be continued in the direction of choosing various combinations of scanning methods and optical flow calculation, the complex application of which will allow searching for identical points regardless of arbitrary rotation and scale changes of video image objects and will increase the speed and accuracy of calculations.

References

1. Lowe D. Distinctive image features from scaleinvariant keypoints. *Intern. Journal of Computer Vision*. 2004. № 60. P. 91–110.
2. Ke Y., Sukthankar R. PCA-SIFT: A more distinctive representation for local image descriptors. Proc. of the *Conf. on Computer Vision and Pattern Recognition (CVPR'04),* 2004. V. 2. P. 506–513.
3. Bay H., Ess A., Tuytelaars T., Gool L.V. SURF: speed up robust features. *Computer Vision and Image Understanding (CVIU*). 2008. V. 110, № 3. P. 346–359.
4. Mikolajczyk K., Schmid C. *Scale and affine invariant interest point detectors. Intern. Journal of Computer Vision*. 2004. №60(1). P. 63–86.
5. Tola E., Lepetit V., Fua P. A Fast Local Descriptor for Dense Matching. Proc. of the *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR'08)*. 2008. P. 1–8.
6. Calonder M., Lepetit V., Strecha C., Fua P. BRIEF: Binary Robust Independent Elementary Features. Proc. of the *11th European Conference on Computer Vision (ECCV'10),* 2010.
7. Rublee E., Rabaud V., Konolige K., Bradski G. ORB: an efficient alternative to SIFT or SURF. Proc. of the *Intern. Conf. on Computer Vision (CVPR'11)*. 2011. P. 2564–2571.
8. Sabelnikov P.Iu., Sabelnikov Yu.A. Poshuk totozhnykh oblastei u zobrazhenniakh iz vykorystanniam invariantnykh momentiv. *Shtuchnyi intelekt*. 2021. № 2. S. 55 – 62.
9. Warren D.H. Electronic Spatial Sensing for the Blind: Contributions from Perception, Rehabilitation, and Computer Vision / D.H. Warren, Edward R. Strelow, – Springer, 1985. – 521 p.
10. Horn B.K.P. Determining Optical Flow / B.K.P. Horn and B.G. Rhunck // Artificial Intelligence. - 1981. - Volume 17, Issues 1–3.
11. Andrew V. Goldberg (2007) Point-to-Point Shortest Path Algorithms with Preprocessing, *Proceedings of 33rd Conference on Current Trends in Theory and Practice of Computer Science*, pp. 88-102.
12. V.P. Boiun, P.Iu. Sabelnikov, Yu.A. Sabelnikov. Prystrii obrobky videodanykh dlia avtomatychnoho suprovodzhennia obiekta, vyznachenoho u zobrazhenni operatorom. Nauka ta innovatsii. 2016, № 12(2). S. 29−39.
13. Oborona Ukrainy https://www.youtube.com/watch?v=IB8IuWvWj5c.

The article has been sent to the editors 23.10.22
After processing 20.11.22