ISSN 0005-1179 (print), ISSN 1608-3032 (online), Automation and Remote Control, 2023, Vol. 84, No. 7, pp. 831–851.

Method for Unmanned Vehicles Automatic Positioning Based on Signal Radially Symmetric Markers Recognition of Underwater Targets

R. M. Shakirzyanov[*,a], **M. P. Shleymovich**[*,b], **and S. V. Novikova**[*,c]

[*] *Tupolev National Research Technical University, Kazan, Russia*

e-mail: [a] *RMShakirzyanov@kai.ru,* [b] *MPShleymovich@kai.ru,* [c] *SVNovikova@kai.ru*

Received November 19, 2022

Revised March 16, 2023

Accepted April 28, 2023

Abstract—The article describes a method for automatically recognizing the target points of the trajectories of unmanned vehicles moving underwater, such as autonomous submarines and flying underwater vehicles of aircraft-like structures. The coordinate of the center of an object with radial symmetry properties is considered a terminal control point. A method for constructing a multiscale weighted image model based on the developed fast radial symmetry transformation and the Hough method is proposed, which ensures noise stability and high speed of calculating the coordinates of the desired point. When the object of interest has a contour of a specific color, a model is based on our proposed chromatic and weight components. As an example of detection, we have given an algorithm for detecting a base underwater station with light markers as a signal luminous ring.

Keywords: automatic transport systems, unmanned underwater vehicle, unmanned flying underwater vehicle, computer vision, image processing, object detection in images, Hough method, fast radial symmetry transformation, image weight model, multiscale image weight model

DOI: 10.25728/arcRAS.2023.24.60.001

1. INTRODUCTION

When building automatic control systems for unmanned vehicles, a necessary condition is to set the final position of the device in space. In most works on automatic control of unmanned vehicles, the researcher sets the final position in the form of a set of specific coordinates [1, 2]. However, in actual conditions, problems arise when the device must independently detect an object of interest, the position of which will set the final position of the device during control. Transport systems, where the device independently determines the end point of its movement and calculates the trajectory of movement to it, are usually called intelligent transport systems [3]. The method for determining the terminal point and constructing the optimal trajectory does not necessarily have to include elements of artificial intelligence (AI), such as machine learning models, neural networks, etc. However, if the calculation method does not involve AI, the system is more often defined as "automatic" or "automated."

The most common unmanned vehicles are unmanned aerial vehicles (UAVs) [4]. The most common unmanned vehicles are unmanned aerial vehicles (UAVs) [5, 6]. For unmanned vehicles moving underwater, the so-called autonomous uninhabited underwater vehicles (AUV) [7], positioning in most cases is carried out using ultrasonic signals [8]. The main disadvantage of this approach is the mandatory presence of special navigation equipment on board the vehicle and the need for a response signal from the target positioning point, which makes the vehicle vulnerable to external detection.

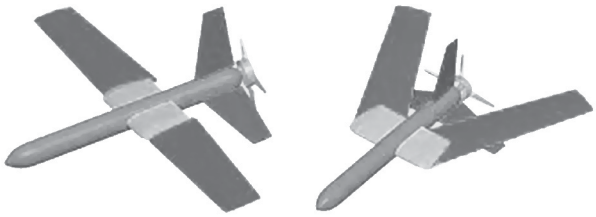

Fig. 1. The design of a diving aircraft with a folding wing (source—see [16]).

Fig. 2. The mechanics of moving guillemots in the air and underwater, applied to simulate the UAV dynamics (source—see [18]).

Positioning systems based on computer vision do not have these disadvantages. They can provide high recognition accuracy and, at the same time, do not give out their presence by external signals. In addition, visual positioning is only possible during underwater target setting for such modern unmanned systems as hybrid unmanned aerial underwater vehicles (HUAUVs) [9]. HUAUVs can overcome distances first by air, dive into the water, and perform underwater maneuvers. Due to signal propagation and dispersion characteristics at the junction of two air-water media, positioning using ultrasonic signals is challenging to implement.

For UAVs and unmanned ground vehicles, the problem of object detection based on computer vision has been well studied [10–12]. There are works on visual positioning of the UAV landing point [13], determining the position of an aircraft based on previously made satellite images of the terrain and neural network recognizers [14] etc. A common disadvantage of the proposed approaches is the need to have additional information for positioning and high requirements for computing power equipment on board the UAV.

For unmanned vehicles moving underwater, computer vision algorithms must meet additional requirements due to the peculiarities of light propagation in water and additional interference due to the turbidity of natural water bodies [15]. In the case of HUAUVs, the speed of the algorithm is also of great importance due to the relatively high speeds of movement of the vehicle and the number of calculations necessary for detecting an object since flying underwater vehicles are not capable of carrying high-power computers and their energy consumption is high relative to AUVs. Therefore, developing a visual positioning algorithm for a target set underwater with minimal energy and computational costs is especially relevant for hybrid UAUVs.

The case when the final target position of the device is underwater is more typical for flying underwater vehicles of the "diving aircraft" type [16], than, for example, for floating quadrocopters [17]. Aircraft-like structures are mainly adapted only to the direct air-to-water transition and experience significant difficulties or cannot carry out the reverse water-to-air transition at all.

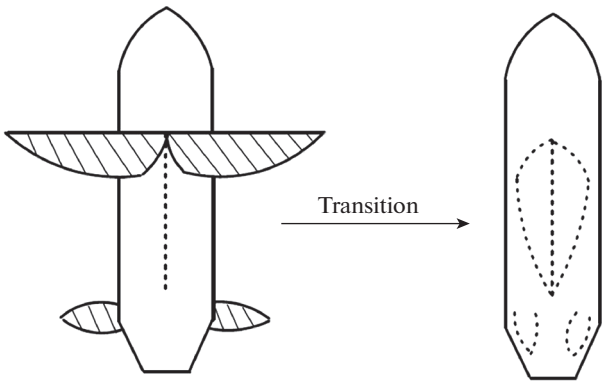

Fig. 3. UAV tail configuration in various environments (source—see [19]).

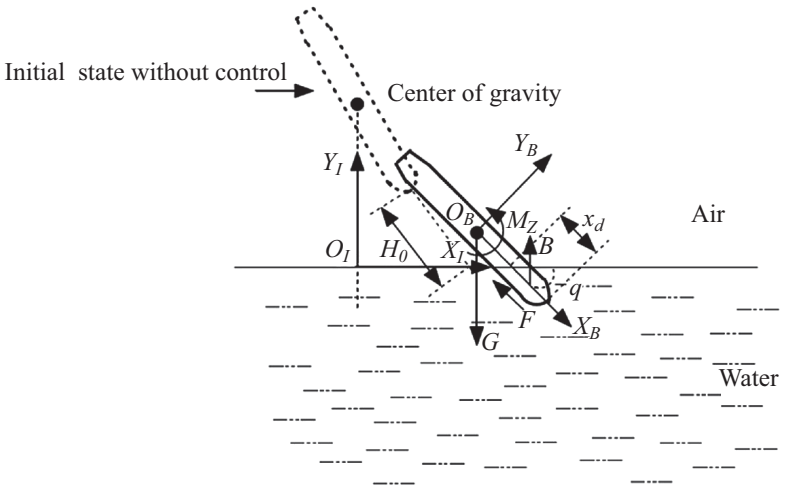

Fig. 4. The authors propose a control system for reaching the endpoint underwater by the vehicle (source—see [19]).

In [16] a design of a diving aircraft with wings deviating by 65 degrees during diving was proposed (see Fig. 1).

The design includes a variable-sweep wing and a compressed carbon dioxide engine. In [18] a design of a diving HUAUV simulating the flight and diving of a guillemot was proposed (see Fig. 2).

In [19] the automatic control system of a diving aircraft with a folding tail is described (see Fig. 3).

In [19] the authors propose a control system for reaching the endpoint underwater by the vehicle (see Fig. 4).

The control optimality criterion is the accuracy of reaching a given position (x_{target}, y_{target}) at the final time t_f.

$$J = \sqrt{(x_h(t_f) - x_{target})^2 - (y_h(t_f) - y_{target})^2}. \tag{1}$$

The control parameters are the initial pitch angle θ_0 and the initial velocity along the axis OX v_{x_0}, the trajectory is calculated without the possibility of maneuvering. As the target coordinates (x_{target}, y_{target}) the authors define a specific point underwater. However, in the case of adaptive control, the device can calculate these coordinates independently in an automatic mode based

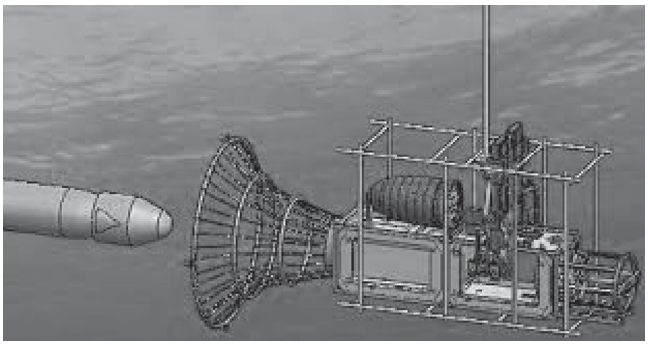

Fig. 5. An example of docking an autonomous uninhabited underwater vehicle with a docking station (source—see [20]).

on machine vision. Thus, the problem arises of recognizing the coordinates of a specific target object underwater, the mathematical center of gravity of which will be the target control point (x_{target}, y_{target}). This problem is one of the possible typical examples of a recognition problem to solve the optimal control problem.

For underwater unmanned vehicles, setting objects of interest underwater is the only possible one. Such an object of interest can mainly be a particular docking station [20], sed to recharge the AUV battery and/or transmit data (see Fig. 5).

In this case, the positioning features of the AUV and HUAUV will be the same since the target is underwater in both cases. The fact that the target is under water is vital for developing a target object detection method since water turbidity, the presence of suspension, and image distortion due to light refraction impose stringent requirements on the detection method for resistance to external noise. For example, optical positioning methods for underwater targets are described in [21, 22]. One [23] and two [24] cameras can be used in this case. The main limitation of the proposed methods is their high consumption of computing and energy resources since almost all of them are designed for AUVs. This circumstance makes the proposed methods poorly applicable to HUAUVs. So, in [20] a convolutional neural network CNN is used as a mathematical recognition model. The efficiency of such networks is high if they are specially trained for a relatively narrow class of objects. For example, it has been shown that CNNs can recognize handwritten text with an accuracy of up to 99% [25]. However, their accuracy drops to 70% or less when recognizing arbitrary natural objects. In addition, CNNs require powerful computing devices for training, which can last from several hours to several days [26]. Modern modifications can give either higher accuracy or higher speed but cannot yet combine both of these qualities. So, in [27] the so-called two-pass algorithms of the FASTER RCNN type are described, reaching an accuracy of 73.2% (VOC07[1] dataset). The network training time is not described, but the network has a relatively low recognition rate of 17 images per second. Single-pass networks like YOLO achieve a recognition rate of 155 images per second, but their accuracy is only 52.7% on the same VOC07 dataset.

Thus, developing an algorithm for detecting objects underwater with minimal computational and time costs is relevant, especially for HUAUVs.

2. FORMULATION OF THE PROBLEM

Consider the problem of recognizing an underwater object in automatic mode from an unmanned vehicle by detecting a base docking station with light markers in the form of a signal luminous ring. The center of the ring defines the target point of the optimal control trajectory without maneuvering

[1] VOC07 is a free access data set containing classified images (Visual Object Classes-VOC) for 20 classes, edition 2007

(x_{target}, y_{target}) according to the example of the problem (1). In the general case, the task can be solved simultaneously for several possible N_{target} targets.

Given:

(a) color image $I = (I_{ij})_{N_x \times N_y}$ in RGB color model;

(b) the set of radii of circles $\mathbf{N}$ to be found;

(c) the set of admissible colors $\overline{\mathbf{C}}$, in shades of which the desired circles can be colored;

(d) allowable mismatch between the positions of the centers of the circles Δ.

Find:

(e) the set of centers of circles with their radii defining the terminal point of the optimal control $\tilde{A} = \left\{ \left(x^k_{target}, y^k_{target} \right), r^k \right\}$, $k = \overline{1, N_{target}}$.

This study aims to develop a computational algorithm that takes positions (a)–(d) as input and generates positions (e) as output. An additional requirement for the algorithm is resistance to external noise and minimization of computational costs and computation time.

When the actual size of a circle (for example, a docking ring) is known, several decreasing values from the real to the minimum possible radius should be chosen as the desired image radii. Subsequently, based on the found value of the radius on the image, it will be possible to determine the distance to the object using additional measuring instruments.

The method should provide recognition under slight deformations of the circle into an ellipse with a ratio of the minor and significant semiaxes of at least 0.95, corresponding to a shift of the direct projection of the view by no more than 20° [28].

3. METHOD FOR SOLVING THE PROBLEM OF RECOGNITION OF COLORED ROUND OBJECTS

An important task solved using computer vision is processing images from an unmanned vehicle to detect and localize given objects — light markers. These objects are characterized by shape and color. Therefore, methods for analyzing color and shape features are used to detect and recognize them.

Recently popular algorithms based on deep neural networks for this problem statement will be redundant since a simple geometric figure is subject to recognition rather than a complex multi-color object. In addition, the generalization complexity and the training stage's duration remain topical problems when using neural networks. Therefore, classical methods in which image analysis is performed remain relevant.

Many practical approaches to the analysis of the shape of objects in images are based on the methods of contour analysis [29]. Light markers for docking underwater vehicles are colored objects with a radially symmetrical shape. For their detection and localization, methods based on analyzing the properties of radial symmetry can be applied. The proposed method is based on the joint use of the well-known Hough method [30] and the transformation of fast radial symmetry using the construction of the Gaussian pyramid.

3.1. Fast Radial Symmetry Transformation

This paper proposes an efficient approach to solving the problem under consideration based on the fast radial symmetry transformation FRST (Fast Radial Symmetry Transform). The FRST transformation makes constructing a weighted image model possible, which can effectively localize the center of a radially symmetric object [31, 32].

To obtain a weighted image model, you must perform the following steps:

1) convert the image to grayscale;

2) calculate the gradients of the brightness function on the image;

3) calculate the values of the weight matrix elements;

4) normalize the values of weight matrix elements;

5) calculate the values of the elements of the matrices of generalized weights;

6) perform low-pass filtering of generalized weight matrices;

7) calculate the values of the matrix elements of averaged weights.

The first step is to convert the image to grayscale, in which each pixel is assigned a brightness value. In the simplest case, the transformation calculates the average values of the intensities of the red, green, and blue color components. Thus, as a result of the first step, the image is represented as a brightness function, the arguments of which are the values of the pixel coordinates.

In the second step, the gradients of the brightness function on the image are calculated using the following operators:

$$g_x(\mathbf{p}) := \hat{\mathbf{I}}(i+1,\, j) - \hat{\mathbf{I}}(i,\, j); \tag{2}$$

$$g_y(\mathbf{p}) := \hat{\mathbf{I}}(i,\, j+1) - \hat{\mathbf{I}}(i,\, j); \tag{3}$$

$$|\mathbf{g}(\mathbf{p})| = \sqrt{g_x^2(\mathbf{p}) + g_y^2(\mathbf{p})}, \tag{4}$$

where $\mathbf{g}(\mathbf{p}) = (g_x(\mathbf{p}),\, g_y(\mathbf{p}))$ is the gradient at pixel $\mathbf{p}$ with coordinates $(i,\, j)$; $g_x(\mathbf{p})$, $g_y(\mathbf{p})$ are the components of the gradient for the horizontal and vertical directions at pixel $\mathbf{p}$ respectively; $\hat{\mathbf{I}}(i,\, j)$, $\hat{\mathbf{I}}(i+1,\, j)$, $\hat{\mathbf{I}}(i,\, j+1)$ are the brightness values in pixels of the grayscale image with coordinates $(i,\, j)$, $(i{+}1,\, j)$ and $(i,\, j{+}1)$ respectively. In expressions (2)–(4), the sign := denotes the assignment operator, which will be used in the future.

In the third step, the values of the elements of the weight matrices are calculated. For this, the following procedure is applied:

1) the set of integer values $\mathbf{N}$ is determined, where $\mathbf{N}$ is the set of radii of the objects to be found;

2) for each value n from the set $\mathbf{N}$, the initial values of the elements of two weight matrices are formed:

$$\mathbf{M}^n(\mathbf{p}) := 0; \tag{5}$$

$$\mathbf{O}^n(\mathbf{p}) := 0, \tag{6}$$

where p is the vector of coordinates $(i,\, j)$.

3) for all elements of the weight matrices determined by the values n from the set $\mathbf{N}$, the following operators are applied:

$$\mathbf{p}_+ := \mathbf{p} + \left\lceil \frac{g_x(\mathbf{p})}{|\mathbf{g}(\mathbf{p})|}\, n \right\rceil; \tag{7}$$

$$\mathbf{p}_- := \mathbf{p} - \left\lceil \frac{g_x(\mathbf{p})}{|\mathbf{g}(\mathbf{p})|}\, n \right\rceil; \tag{8}$$

$$\mathbf{M}^n\left(\mathbf{p}_+\right) := \mathbf{M}^n\left(\mathbf{p}_+\right) + |\mathbf{g}(\mathbf{p})|; \tag{9}$$

$$\mathbf{M}^n\left(\mathbf{p}_-\right) := \mathbf{M}^n\left(\mathbf{p}_-\right) - |\mathbf{g}(\mathbf{p})|; \tag{10}$$

$$\mathbf{O}^n\left(\mathbf{p}_+\right) := \mathbf{O}^n\left(\mathbf{p}_+\right) + 1; \tag{11}$$

$$\mathbf{O}^n\left(\mathbf{p}_-\right) := \mathbf{O}^n\left(\mathbf{p}_-\right) - 11, \tag{12}$$

where $\lceil \cdot \rceil$ is the operator of rounding an actual number to the nearest no less than integer value.

In the fourth step, the elements of the weight matrices are normalized:

$$\mathbf{M}^n(\mathbf{p}) := \frac{|\mathbf{M}^n(\mathbf{p})|}{\max\limits_q \{|\mathbf{M}^n(\mathbf{q})\}|}; \tag{13}$$

$$\mathbf{O}^n(\mathbf{p}) := \frac{|\mathbf{O}^n(\mathbf{p})|}{\max\limits_q \{|\mathbf{O}^n(\mathbf{q})\}|}. \tag{14}$$

In the fifth step, the values of the elements of the matrices of generalized weights are calculated:

$$\mathbf{F}^n(\mathbf{p}) := (\mathbf{O}^n(\mathbf{p}))^\alpha \, \mathbf{M}^n(\mathbf{p}), \tag{15}$$

where α is the radial stiffness parameter.

In the sixth step, low-pass filtering of the generalized weight matrices is performed. In this case, a Gaussian low-pass filter is used as a rule. Filtering is expressed using the convolution operator

$$\mathbf{S}^n := \mathbf{F}^n * \mathbf{G}^n, \tag{16}$$

where $\mathbf{G}^n$ is matrix of coefficients of the Gaussian low-pass filter, defined for the value n from the set $\mathbf{N}$.

In the last, seventh step, the values of the elements of the matrix of averaged weights are calculated:

$$\mathbf{S}(\mathbf{p}) := \frac{1}{|\mathbf{N}|} \sum_{n \in \mathbf{N}} \mathbf{S}^n(\mathbf{p}). \tag{17}$$

The matrix of average weights obtained as a result of the FRST transformation is a weight model of the image, the analysis of which makes it possible to determine the shape parameters of objects, for example, the center coordinates and the radius of round objects.

3.2. Image Weight Model

When building a weighted image model using a fast radial symmetry transformation, it is necessary to perform a relatively large number of calculations, which is determined by the power of the set of integer values $\mathbf{N}$. The elements of this set are used to analyze significant changes in brightness at the corresponding distances from the current pixel, i.e., have the meaning of radii for radially symmetrical objects centered in image pixels.

Objects in actual images have a multiscale nature since they have different sizes. For the problem under consideration, this means using a set $\mathbf{N}$ with many elements, which are determined by the range of changes in the radii of objects of interest. Therefore, analyzing the original image presented in a multiscale form is advisable. Such a multiscale analysis reduces the number of calculations and improves object detection accuracy in the image.

To build a multi-scale image weight model, you must perform the following steps:

1) convert the image to grayscale;
2) construct a Gaussian pyramid;
3) calculate the values of the average weight matrices' elements for all Gaussian pyramid levels;
4) bring the matrices of averaged weights to the size of the original image;
5) calculate the values of the elements of the integral matrix of averaged weights.

The first step is to convert the image to grayscale, in which each pixel is assigned a brightness value.

In the second step, the Gaussian pyramid is built [33]. It is a set of images $\mathbf{P}=\{\mathbf{P}_l \,|\, l=\overline{0, L-1}\}$, where L is the number of levels in the pyramid. The original grayscale image $\hat{\mathbf{I}}$ is considered as the zero level of the pyramid $\mathbf{P}_0$. The remaining levels of the pyramid are formed as follows:

$$\mathbf{P}_l := \mathbf{G} * (2 \downarrow [\mathbf{P}_{l-1}]), \tag{18}$$

where $\mathbf{G}$ is the coefficient matrix of the Gaussian low-pass filter; $2 \downarrow [\cdot]$ is the image downsampling operator, for example, by removing every second pixel in a row and column. An element of the Gaussian pyramid of level l will have dimensions four times smaller compared to the dimensions of an element of level $l-1$.

In the third step, with the help of operators (5)–(17), weight models of images of all levels of the Gaussian pyramid are formed, i.e., a set of matrices of averaged weights $\left\{\mathbf{S}^{(0)}, \mathbf{S}^{(1)}, \ldots, \mathbf{S}^{(L-1)}\right\}$ is built.

In the fourth step, the set $\left\{\mathbf{S}^{(0)}, \mathbf{S}^{(1)}, \ldots, \mathbf{S}^{(L-1)}\right\}$ is transformed into the set $\left\{\mathbf{R}^{(0)}, \mathbf{R}^{(1)}, \ldots, \mathbf{R}^{(L-1)}\right\}$, each element which $\mathbf{R}^{(l)}$ is the result of reducing the matrix $\mathbf{S}^{(l)}$ to the dimensions of the original image, i.e., to the dimensions $\mathbf{S}^{(0)}$. The specified transformation is performed as follows:

$$\mathbf{R}^{(l)} := \mathbf{G} * \left(2 \uparrow \left[\mathbf{S}^{(l)}\right]\right), \tag{19}$$

where $\mathbf{G}$ is the coefficient matrix of the Gaussian low-pass filter; $2 \uparrow [\cdot]$ is the image upsampling operator (the matrix $\mathbf{S}^{(l)}$ is considered as an image with each pixel which is associated with a weight value), for example, by duplicating each pixel in a row and column.

In the last, fifth step, the resulting matrices $\mathbf{R}^{(0)}$, $\mathbf{R}^{(1)}$, $\mathbf{R}^{(L-1)}$ are added to construct an integral matrix of averaged weights:

$$\mathbf{S}(\mathbf{p}) := \frac{1}{L} \sum_{l=0}^{L-1} \mathbf{R}^{(l)}(\mathbf{p}), \tag{20}$$

where $\mathbf{p}$ is a pixel of the original image corresponding to the coordinates of the weight matrix elements.

As a result of the above steps, a multiscale weight model is formed, the analysis of which makes it possible to determine the shape parameters of objects in the image. The use of this model makes it possible to increase the accuracy of work and reduce computational complexity due to the smaller image area at the upper level of the pyramid.

3.3. Chromatic Image Model

When analyzing images of light markers from an unmanned underwater vehicle, it is advisable to use a model containing chromatic and weight components. The first allows you to consider the color characteristics of objects of interest, and the second is the geometric features of their shape.

Then the image model can be represented as $\langle \mathbf{C}, \mathbf{S} \rangle$, where $\mathbf{C}$ is the chromatic component, and $\mathbf{S}$ is the weight component. The weight component of the model is formed using the procedures described in the previous sections. The chromatic component is formed due to color transformation and color segmentation.

To describe the chromatic component of the model, the advantages and disadvantages of several color spaces were investigated: RGB, Lab, and HSV. These spaces were compared in terms of description methods to highlight areas of the image, the color of which lies in a specific range [34]. As a result, it was noted that the main factor limiting and complicating the use of the RGB color

model for color segmentation is a single description of both the illumination and the color component of the object, which makes it difficult to clearly define the ranges of the desired colors [35]. The indisputable advantage of the L*a*b* color model is a separate L* channel responsible for illumination and planes (a*, b*), responsible for color. However, this model describes color ranges (chromatic characteristics) as geometric figures on a plane, possibly having a complex structure. This fact complicates constructing a color distribution model for segmentation [36]. The HSV color model, in contrast to L*a*b*, allows you to set the range of desired colors by specifying the range of values of the H channel (in most cases, it is enough to specify the threshold value of the saturation set in the S channel above the value whose color purity becomes acceptable), which significantly simplifies the implementation of the method [37].

Thus, the HSV color space (model) was chosen as the color space for analyzing color images. This model is based on the characteristics of information perception by the human visual system and is represented as a set of color channels H, S, and V, which determine the color's tone, saturation, and value (brightness).

The chromatic component $\mathbf{C}$ of the image model is represented as follows:

$$\begin{cases} (h(\mathbf{p}),\, s(\mathbf{p}),\, v(\mathbf{p})), & \exists\, k,\, l,\, m:\ h(\mathbf{p}) \in \mathbf{H}^k \wedge s(\mathbf{p}) \in \mathbf{S}^l \wedge v(\mathbf{p}) \in \mathbf{V}^m; \\ (0,\, 0,\, 0), & \forall\, k,\, l,\, m:\ h(\mathbf{p}) \notin \mathbf{H}^k \vee s(\mathbf{p}) \notin \mathbf{S}^l \vee v(\mathbf{p}) \notin \mathbf{V}^m, \end{cases} \tag{21}$$

where $\mathbf{p}$ is the pixel of the original image; $\mathbf{H}^k$, $\mathbf{S}^l$, $\mathbf{V}^m$ are the kth, lth, and mth intervals of the H, S and V channels, respectively, from the set of color intervals of the given objects.

3.4. Algorithm for Detecting Round Objects on Images of Light Markers

In general, the procedure for detecting round objects of a given color is as follows:

1) convert the original image from the RGB to the HSV color model;
2) build the chromatic component of the image model;
3) convert the original image to grayscale, considering the chromatic component;
4) find circles using the Hough method on the image by the chromatic component;
5) build the weight component of the image model (weight image);
6) threshold the weight image;
7) find contours on the weight image;
8) calculate the centers of mass for each contour;
9) find circles on the image by the weight component;
10) remove falsely found circles on the halftone image by the Hough method, the centers of which do not coincide with the permissible error with any center of mass of the contours on the weighted image;
11) generate a set of coordinates of target control points with radii $\tilde{A} = \left\{ \left(x^k_{target},\, y^k_{target} \right),\, r^k \right\}$.

4. COMPUTATIONAL EXPERIMENTS

The presented algorithm was tested on the problem of recognizing circular luminous markers of an underwater docking station at different angles in turbid water conditions. For computational experiments, images of the docking station were obtained during the experiments of Liu S., Ozay M., Okatani T., published in [20]. Examples of images of the docking station docking block in the form of illuminated circles, which were recognized as a result of the experiments, are shown in Fig. 6.

The images were taken with a 620 TVL (0.8 Mpx) NanoSeaCam monocular color camera with viewing angles of $59^\circ \times 44^\circ \times 72^\circ$ (up-down-sideways).

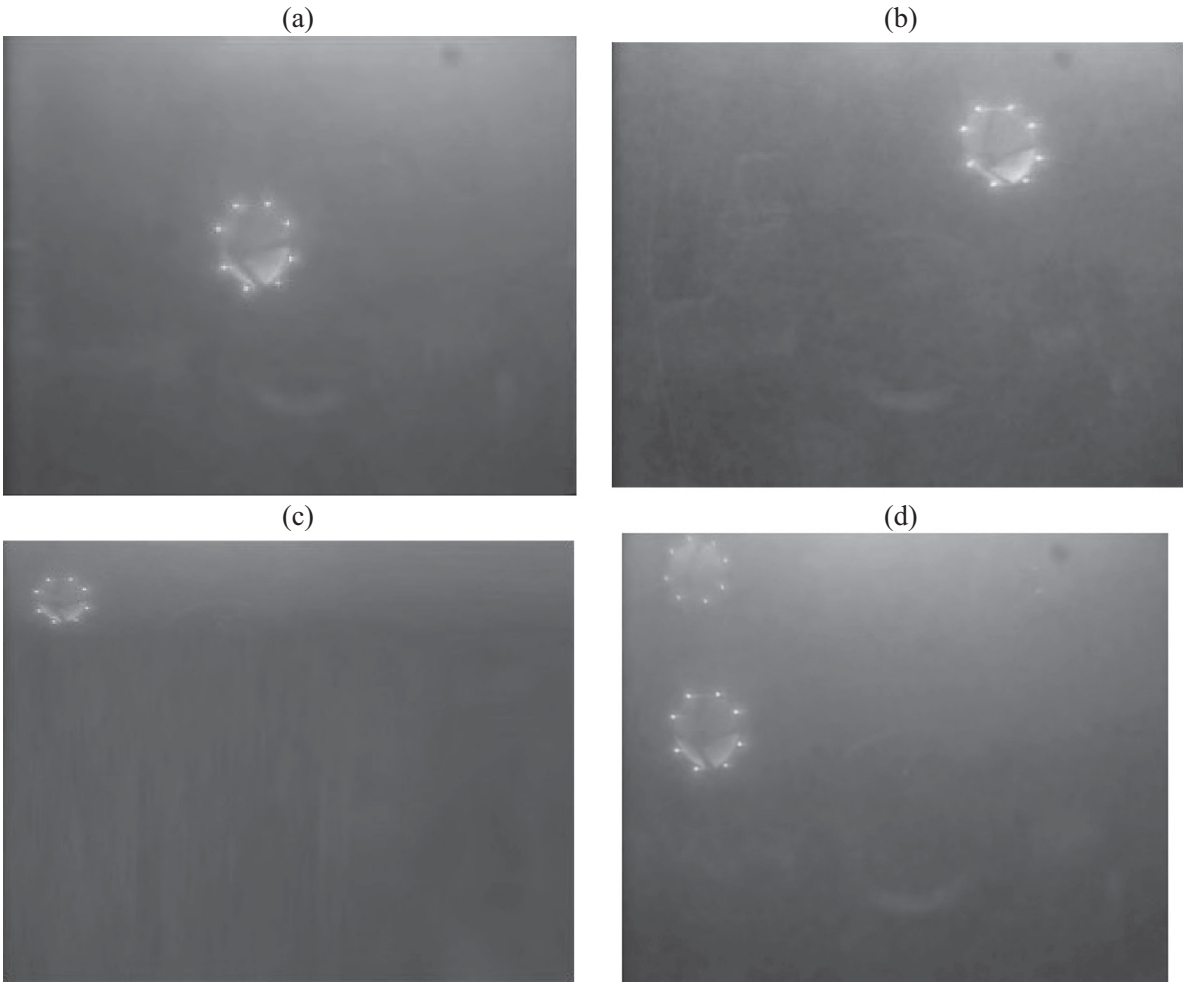

Fig. 6. Variants of images of circles — the docking luminescent ring of the underwater docking station: (a) a single object in the central part of the image, the area of the object of interest relative to the image is more than 5%; (b) a single object displaced relative to the center of the image, the area of the object of interest relative to the image is more than 5%; (c) a single object displaced relative to the center of the image, the area of the object of interest relative to the image is less than 5%; (d) multiple objects of interest in one image. (source: freely available dataset http://vision.is.tohoku.ac.jp/liushuang/a-vision-based-underwater-docking-system/dataset).

4.1. The Criterion for Assessing the Accuracy of the Algorithm

The effectiveness of the developed algorithm was evaluated based on the so-called pixel-oriented technique described in [38].

This is a statistical evaluation measure based on the count of misclassified pixels. To obtain such an estimate, we must calculate the probabilities that a randomly selected pixel in an image segmented using the algorithm belongs either to the desired object or, respectively, to the background.

Unlike generally accepted metrics such as Accuracy, Precision, or Recall, followed by ROC analysis, the pixel-based metric evaluates not the accuracy of object classification (object detected or not detected) but the detection quality. Evaluation of the detection quality includes assessing the accuracy of object localization (determining the coordinates of the center) and determining its geometric characteristics, i.e., the radius. Therefore, for the problem posed, the pixel-oriented estimate is preferable [39].

Table 1. Initial data and preliminary results of experiments on the recognition of test images

Experiment number	Prior data				Data obtained after applying the algorithm	
	$p(o)$	$p(b)$	Data obtained after applying the algorithm, %	Estimation of the object's relative area	Percentage of erroneous recognitions algorithm by criterion $p(b\|o)$	Percentage of erroneous recognitions algorithm by criterion $p(o\|b)$
1	0.01	0.99	1%	$\leq 5\%$	1	1
2	0.01	0.99	1%	$\leq 5\%$	1	5
3	0.01	0.99	1%	$\leq 5\%$	1	10
4	0.01	0.99	1%	$\leq 5\%$	1	25
5	0.05	0.95	5%	$\leq 5\%$	1	1
6	0.05	0.95	5%	$\leq 5\%$	1	5
7	0.05	0.95	5%	$\leq 5\%$	1	10
8	0.05	0.95	5%	$\leq 5\%$	1	25
9	0.1	0.9	10%	$> 5\%$	1	1
10	0.1	0.9	10%	$> 5\%$	1	5
11	0.1	0.9	10%	$> 5\%$	1	10
12	0.1	0.9	10%	$> 5\%$	1	25
13	0.25	0.75	25%	$> 5\%$	1	1
14	0.25	0.75	25%	$> 5\%$	1	5
15	0.25	0.75	25%	$> 5\%$	1	10
16	0.25	0.75	25%	$> 5\%$	1	25
17	0.5	0.5	50%	$> 5\%$	1	1
18	0.5	0.5	50%	$> 5\%$	1	5
19	0.5	0.5	50%	$> 5\%$	1	10
20	0.5	0.5	50%	$> 5\%$	1	25

According to the chosen method, the segmentation error probability of the entire image is determined as follows:

$$p_{err} = p(b|o)p(o) + p(o|b)p(b). \tag{22}$$

Here: $p(o)$ is the a priori probability that a randomly selected pixel of the original image belongs to the object (the ratio of the area of the object to the area of the entire image); $p(b)$ is the a priori probability that a randomly selected pixel of the original image belongs to the background (the ratio of the background area to the area of the entire image); $p(o|b)$ is the probability that a pixel belonging to the background is erroneously assigned to the object during segmentation (the ratio of the part of the background area erroneously assigned by the algorithm to the object to the entire background area). In the theory of statistical hypotheses, such an error is called an error of the first kind; $p(b|o)$ is the probability that a pixel belonging to an object will be erroneously assigned to the background (the ratio of the part of the object's area erroneously assigned by the algorithm to the background to the area of the object) — an error of the second kind.

So $p(o|b)$ and $p(b|o)$ are cumulative segmentation errors.

In this case, the area is understood as the number of pixels of a given area.

For cases when the desired objects are small in size relative to the entire image (occupying less than 5% of the image area), i.e.,

$$\#(o) \ll \#(\mathcal{D}),$$

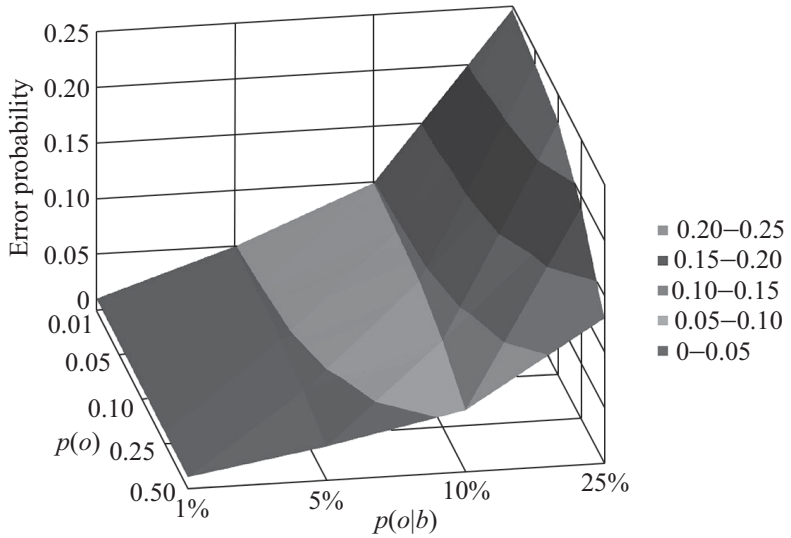

Fig. 7. Probability of the error of the algorithm calculated by the method (22) depending on the a priori probability (22) depending on the a priori probability $p(o)$ and a posteriori probability $p(b|o)$.

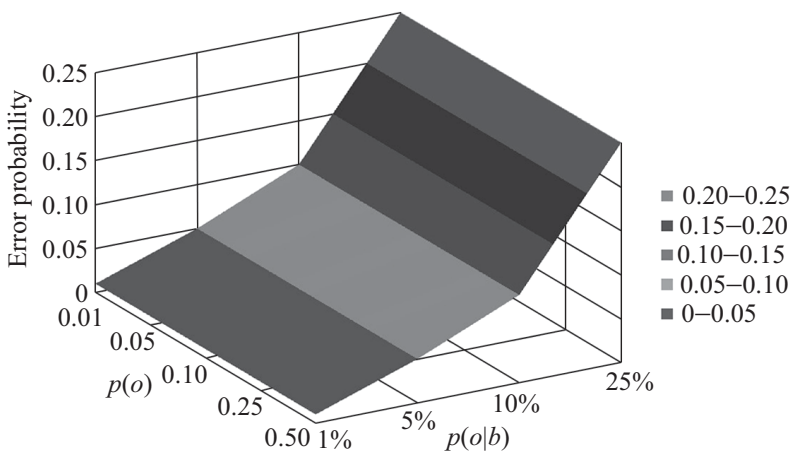

Fig. 8. Probability of an algorithm error calculated by method (23) depending on a priori probability $p(o)$ and a posteriori probability $p(b|o)$.

where $\#(o)$ is the power of the set of pixels belonging to the desired object, and $\#(\mathcal{D})$ is the power of the set of pixels of the entire image, it is advisable to use estimate (22) instead of formula (23):

$$p_{err} := \frac{\#(o_s) - \#(o)}{\#(\mathcal{D})}, \tag{23}$$

where o_s is the set of found objects as a result of segmentation.

To determine the optimal way to calculate the algorithm's accuracy, we conducted 20 independent experiments on recognizing circles in images. Among them, in 8 images, the object of interest occupied 5% or less of the area, and in 12 images, it occupied more than 5%. After image processing by the developed algorithm, posterior recognition errors $p(o|b)$ and $p(b|o)$ were obtained. A priori and a posteriori experimental data are presented in Table 1.

Then, based on the obtained data, the accuracy was calculated according to the methods (22) and (23). The results are presented in Figs. 7–10.

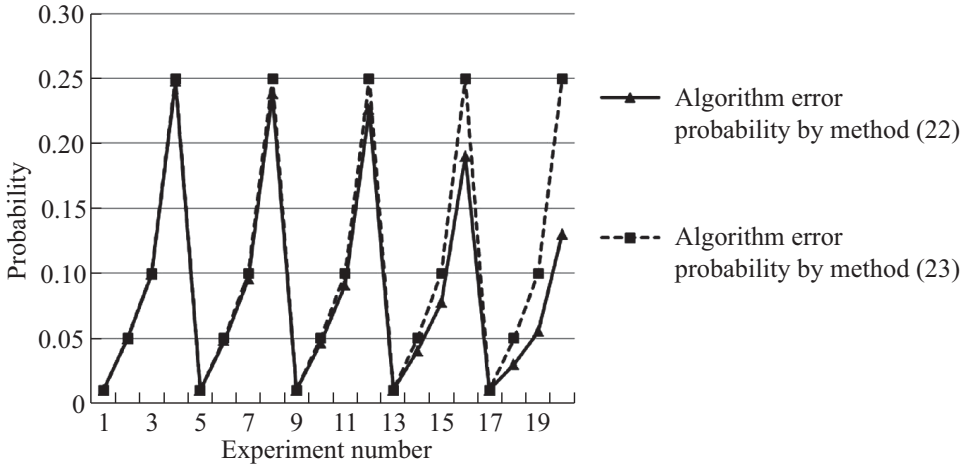

Fig. 9. Comparative efficiency of methods for estimating the algorithm's accuracy.

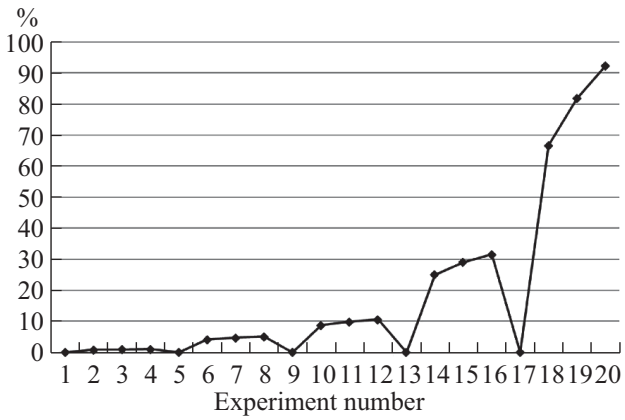

Fig. 10. Difference in accuracy of method (22) relative to method (23).

The results show that for objects with an area of up to 5% of the area of the entire image, inclusive, the correctness estimates calculated by both methods with a fixed type 2 error level of 0.01, generally have minor differences. However, if the desired object occupies an area of more than 5% of the area of the entire image, it is advisable to apply the estimate according to the Eq. (22). Thus, method (22) is preferred.

For all further studies, the accuracy was calculated according to (22).

4.2. Influence of Radial Symmetry Transformation Parameters

The dependence of the circle detection accuracy on the radial stiffness parameter α is studied. We generated and examined 100 images of circles with radii from 1 to 100 pixels on the 2400 × 8400-pixel original image. ext, we conducted experiments with a change in the stiffness parameter from $\alpha = -3$ to $\alpha = +7$. The results are presented in Fig. 11a. It can be seen that with an increase in the radial stiffness parameter α to a value of +2, the detection accuracy reaches 75% (0.75). With a further increase in α an accuracy of 95% is achieved at a value of +4 and then asymptotically tends to 100%. On Fig. 11b hows the influence of the standard deviation on the accuracy of detecting circles at the value of the radial stiffness parameter α equal to 2. It can be seen that this parameter has an insignificant effect on the result under these conditions.

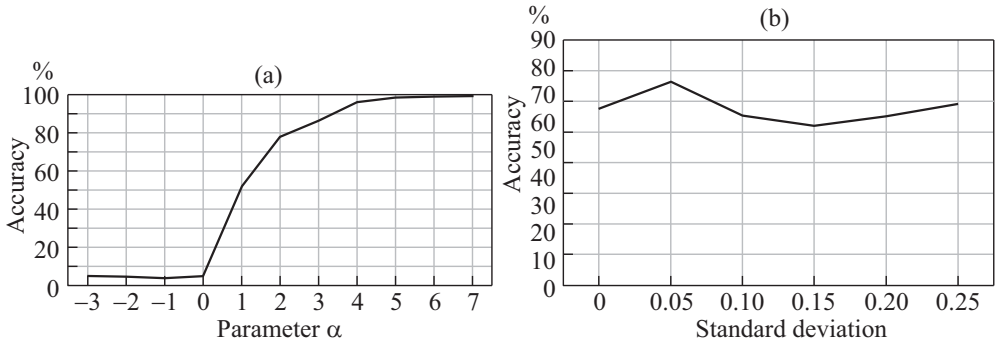

Fig. 11. Influence of algorithm parameters on the accuracy of circle detection: (a) radial stiffness parameter; (b) standard deviation.

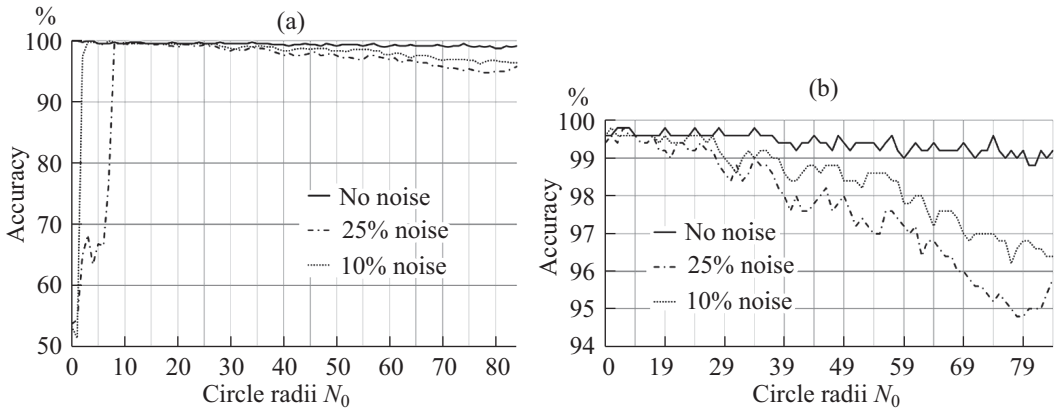

Fig. 12. Influence of impulse noise on the accuracy of circle recognition for different radii: (a) for all circles; (b) for circles with a radius greater than 10 pixels.

4.3. Influence of Noise on Recognition Accuracy

The turbidity of the water, the presence of slight organic and inorganic inclusions, and distortions caused by the refraction of light between the water and the camera lens are mathematically described as additive noise. In particular, a refined solid suspension in water is modeled as an impulse noise of the "salt" and "pepper" types. The effects of distortion and turbidity were modeled using masks as Gaussian and "blurring" noises. The impact on the recognition of each type of noise is discussed below.

4.3.1. Influence of Impulse Noise

To determine the effect of impulse noise on the algorithm's operation, 84 images of circles with radii from 1 to 84 pixels were examined. The original image was noisy, with noise at 25% intensity and 10% of the image area. The results of the algorithm are shown in Fig. 12.

It can be seen that impulse noise has a significant effect on objects of small sizes. Its influence on the result of the recognition of larger objects is less significant.

4.3.2. Influence of Gaussian Noise

To study the effect of Gaussian noise on the operation of the algorithm, we imposed noise on the original image by filling image pixels with normally distributed random numbers with mathematical expectation $\mu = 0$ and standard deviation $\sigma = 5$. Then the image was processed by

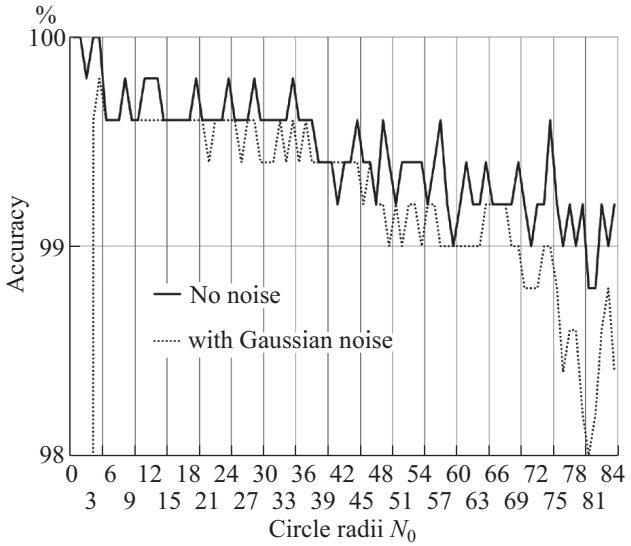

Fig. 13. Influence of Gaussian noise on recognition accuracy circles for various radii.

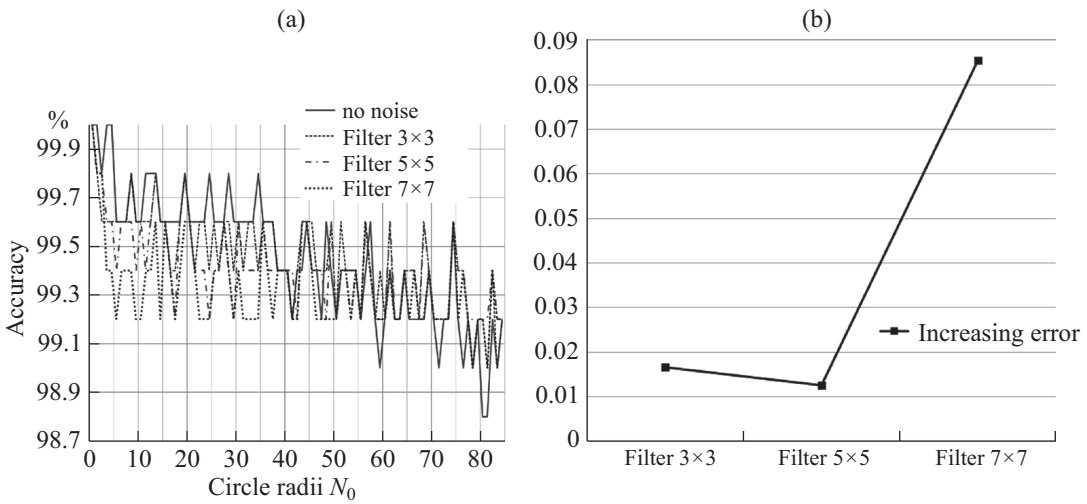

Fig. 14. Influence of image blur by filters on circle recognition accuracy: (a) comparative accuracy for various blur filters; (b) the relative average increase in the recognition error for various filters compared to the undistorted image.

a radial symmetry detector with a radial stiffness parameter α, equal to 2. The results are presented in Fig. 13.

Gaussian noise has a significant effect on small objects. For large objects, the effect of this kind of noise is insignificant.

4.3.3. Influence of Blur-Type Noise

To study the influence of noise of the "blur" type, we subjected the original image to blurring with filters of dimensions 3×3, 5×5 and 7×7. The results of changing the accuracy of circle recognition are shown in Fig. 14.

The blur-type noise practically does not affect the result of object detection. The loss of accuracy is no more than 0.09%.

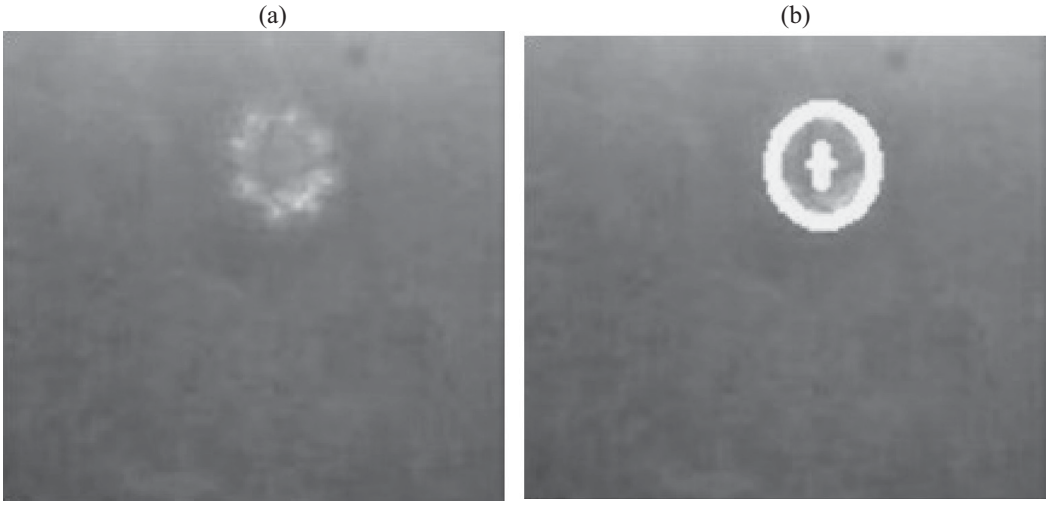

Fig. 15. Light Marker Detection Example: (a) the original image of the marked ring of the docking station at an angle; (b) a recognized circle with a calculated center.

An interesting fact is the increase in recognition accuracy in the presence of "blur" type noise compared to a non-noisy image in several experiments. This is due to the problem of false detection when the algorithm erroneously refers to the found circles of a given radius and also closely spaced circles of other, close in value, radii. With blur noise applied, the probability of such a false positive detection is reduced in some cases.

The obtained results of studying the influence of various types of noise on the accuracy of recognition of radial images show that it is not required to apply additional sharpening algorithms since the developed method is sufficiently resistant to noise within the framework of the problem being solved.

4.4. Image Detection

The image of a light docking ring in turbid water refers to images with a high degree of blurring distortion and, to a lesser extent, Gaussian and impulse noise. Thus, for objects with a radius of more than 10 pixels (large objects), the developed algorithm provides high resistance to all types of noise, and according to the main distorting criterion, blur, it provides stability for recognizing an underwater docking luminescent ring of all sizes.

An example of the image of a light marker in turbid water at an angle of 16° o the camera with a recognized docking center is shown in Fig. 15.

In the course of numerical experiments, both the circles themselves were recognized, and their centers were found, the coordinates of which can potentially serve as the coordinates of the final optimal position of the unmanned vehicle during automatic control. For the image presented in Fig. 15, in particular, we have:

Original image parameters:

(a) a color image with a dimension of 448 × 448 pixels in *.jpeg format;

(b) the number of desired circles is 1, the desired radius is 85 pixels;

(c) the set of acceptable colors — shades of yellow;

(d) allowable mismatch between the positions of the centers of the circles — 5 pixels;

found a circle centered at a point:

- $x_{target} = 282$;
- $y_{target} = 298$.

Table 2. Comparison of recognition accuracy for radial objects with a uniform increase in radii

Minimum distance between circle centers, pix.	The actual number of circles in the image	Found by the Hough method	Found by the developed algorithm
5	7	21	8
10	7	7	7
15	7	7	7
20	7	7	7
25	7	7	7
30	7	7	7
35	7	7	7
40	6	7	6
45	7	8	8
50	6	6	6

Table 3. Comparison of recognition accuracy for radial objects with a random change in radii

Minimum distance between circle centers, pix.	The actual number of circles in the image	Found by the Hough method	Found by the developed algorithm
5	8	152	12
10	8	53	8
15	8	30	8
20	8	22	8
25	8	14	8
30	8	10	8
35	8	10	8
40	8	10	8
45	8	8	8
50	8	6	8

4.5. Comparison of the Efficiency of Algorithms

In this section, we will be comparing the accuracy and speed of the developed algorithm with the Hough and fast radial symmetry algorithms that were used separately. It's worth noting that methods such as neural network recognition require additional computational costs and memory costs of the computing device at the training stage. This is mainly because of the need to have a relevant training dataset that includes all possible combinations of input parameters to obtain an adequate model. Due to this, the use of such machine-learning algorithms is unsuitable for low-performance computing devices that are installed on board HUAUVs or similar unmanned vehicles.

4.5.1. Comparison of Accuracy of Radial Objects Detection

To illustrate the method's effectiveness for detecting radial objects in images based on the proposed algorithm, we conducted two computational experiments: images of circles with uniform and random changes in radii.

Tables 2 and 3 show the results of comparing the accuracy of the algorithms when varying the minimum distance between the centers of the circles. An image is considered correctly recognized if the deviation of the found centers from the real ones does not exceed 15 pixels.

Table 4. Characteristics of image datasets for experiments comparing the speed of algorithms

Dataset name	Short description
LISA Traffic Light Dataset	100 images from a set of size 1280×960 in JPG format
Bosch Small Traffic Lights Dataset	PNG-format images from a set of various sizes
Selective set from Kaggle	200 PNG-format images from a set of various sizes

Table 5. Comparison of the speed of recognition of radial objects of different ranges of circles

Range of circles for recognition	FRST algorithm running time, ms	Proposed algorithm running time, ms	Speed gain of the developed algorithm, times
1–10	89 988	50 569	1.8
1–20	171 902	79 753	2.1
1–30	253 435	111 020	2.3
1–50	465 673	128 970	3.6

The comparison results show that the accuracy of the developed algorithm is, on average, 35% higher than the accuracy of classical methods used separately.

4.5.2. Comparison of the Speed of Radial Objects Detection

For a relative comparison of the speed of the developed algorithm, we used a computer with the following characteristics:

- processor: Intel Core i5-3230M 2.60 GHz;
- RAM: 8 GB;
- operating system: Windows 10, 64-bit.

The experiments were carried out for three test sets of images: 2 complete datasets, "LISA Traffic Light Dataset" and "Bosch Small Traffic Lights Dataset" of the Kaggle open type dataset, as well as 200 sample images from various Kaggle datasets relevant to the study. A description of the test set of images from these databases is given in Table 4.

We conducted computational experiments on circle recognition in four ranges: from 1 to 10 pixels, from 1 to 20 pixels, from 1 to 30 pixels, and from 1 to 50 pixels. Table 5 shows the results of estimating the speed of the algorithm proposed by us in comparison with the primary method based on the radial symmetry transformation FRST.

It is evident that with the increase in the range of desired circles, the advantage of the proposed method's speed increases compared to the basic one.

It should be noted that the computer used for the experiments had excess power for the practical implementation of the proposed algorithm and was used solely to compare the author's algorithm with the classical one. In natural conditions, the power of HUAUV onboard computers is sufficient for the algorithm due to the simplicity of the implemented calculations.

5. CONCLUSION

We have proposed a method that can be used to detect and localize objects of a given shape and color in images of light markers. For example, our method can be used to detect an object of interest in an unmanned underwater or aerial underwater vehicle. The described method is easy to implement and resistant to interference. It can be supplemented with various image processing operations at all stages to improve performance, vital for detecting an object in an aquatic environment with reduced visibility.

In addition, it should be noted that the procedure for generating a multiscale image weight model has the properties of natural parallelism. This determines the possibility of increasing the processing speed through hardware and software tools for high-performance parallel computing, essential for effective real-time control of an unmanned vehicle and reducing power consumption.

The limitation of the algorithm's applicability in water turbidity conditions is the size of the detected radial object. In particular, objects with a radius of less than 10 pixels are difficult to detect by the developed algorithm. Larger objects are recognized with a high accuracy of over 95%. In this case, the radial stiffness parameter α should equal +4. For effective detection of objects, the level of impulse noise in images should not exceed 25%, while Gaussian noise has little effect on detecting objects whose radius exceeds 5 pixels. The developed algorithm is resistant to blurring noise with averaging filters. Because of the preceding, the algorithm does not require additional methods to improve image clarity, which is its undeniable advantage over analogs.

This advantage is obtained through the joint application of several basic algorithms, such as the Hough method, the fast radial symmetry transformation, and the construction of the Gaussian pyramid. The resulting unified model outperforms each of the basic algorithms individually and, when applied in pairs, in performance and computing resource requirements.

REFERENCES

1. Liguo, T., Shenmin, S., Xiaoyan, Y., and Jianwen, S., An Overview of Marine Recovery Methods of UAV for Small Ships, *J. Harbin Institute Technol.*, 2019, vol. 51, no. 10, pp. 1–10.
2. Chamola, V., Kotesh, P., Agarwal, A., Naren, Gupta, N., and Guizani, M., A Comprehensive Review of Unmanned Aerial Vehicle Attacks and Neutralization Techniques, *Ad Hoc Networks*, 2021, vol. 111, p. 102324. https://doi.org/10.1016/j.adhoc.2020.102324
3. State Standard 56829-2015. Intelligent transport systems. Terms and Definitions. Moscow, Standartinform Publ. 10 december 2015. 10 p. (In Russian)
4. Albeaino, G., Gheisari, M., and Franz, B., A Systematic Review of Unmanned Aerial Vehicle Application Areas and Technologies in the AEC Domain, *J. of Inform. Technol. Construct.*, 2019, vol. 24, pp. 381–405.
5. Hajiyev, C., Soken, H.E., and Vural, S.V., Navigation Systems for Unmanned Aerial Vehicles, *State Estimation and Control for Low-cost Unmanned Aerial Vehicles*, 2015, pp. 25–49. https://doi.org/10.1007/978-3-319-16417-5_3
6. Moiseev, V.S., Shafigullin, R.R., and Gushchina, D.S., Rational Placement and Required Number of Information Unmanned Aerial Systems for On-Line Monitoring of Large Territories, *Russian Aeronautics*, 2012, vol. 55, pp. 223–229. https://doi.org/10.3103/S1068799812030014
7. Neira, J., Sequeiros, C., Huaman, R., Machaca, E., Fonseca, P., and Nina, W., Review on Unmanned Underwater Robotics, Structure Designs, Materials, Sensors, Actuators, and Navigation Control, *J. Robot.*, 2021, pp. 1–26. https://doi.org/10.1155/2021/5542920
8. Kolesnikov, M.P., Martynowa, L.A., Pashkevich, I.V., and Shelest, P.S., Method of AUV Positioning While Homing to the Docking Station, *Izvestiya Tula State University. Technical science*, 2015, no. 11–2, pp. 38–49. (In Russian)
9. Qiu, S. and Cui, W., An Overview on Aquatic Unmanned Aerial Vehicles, *Ann. Rev. Res.*, 2019, vol. 5, no. 3, p. 555663. https://doi.org/10.19080/ARR.2019.05.555663
10. Popescu, D. and Ichim, L., Image Recognition in UAV Application Based on Texture Analysis, *Advanced Concepts for Intelligent Vision Systems. ACIVS 2015. Lecture Notes in Computer Science*, 2015, vol. 9386. https://doi.org/10.1007/978-3-319-25903-1_60
11. Samadzadegan, F., Dadrass Javan, F., Ashtari Mahini, F., and Gholamshahi, M., Detection and Recognition of Drones Based on a Deep Convolutional Neural Network Using Visible Imagery, *Aerospace*, 2022, vol. 9, no. 1, p. 31. https://doi.org/10.3390/aerospace9010031

12. Fujiyoshi, H., Hirakawa, T., and Yamashita, T., Deep Learning-Based Image Recognition for Autonomous Driving, *IATSS Res.*, 2019, vol. 43, no. 1, pp. 244–252. https://doi.org/10.1016/j.iatssr.2019.11.008

13. Sevostyanov, I.E. and Devitt, D.V., Sual Positioning System for Multi-Rotor Drones for High-Precision Autonomous Landing, *Science Time*, 2021, no. 90, pp. 38–42. (In Russian)

14. Stepanov, D.N., Methods and Algorithms for Determining the Position and Orientation UAV Using on-Board Videocameras, *Software Products and Systems*, 2014, no. 1, pp. 150–157. (In Russian)

15. Deltheil, C., Didier, L., Hospital, E., and Brutzman, D.P., Simulating an Optical Guidance System for the Recovery of an Unmanned Underwater Vehicle, *IEEE J. Ocean. Engineer.*, 2000, vol. 25, no. 4, pp. 568–574. https://doi.org/10.1109/48.895364

16. Guo, D., Bacciaglia, A., Simpson, M., Bil, C., and Marzocca, P., Design and Development a Bimodal Unmanned System, *AIAA Scitech 2019 Forum*, 2019, pp. 1–7. https://doi.org/10.2514/6.2019-2096

17. Pinheiro, P.M., Neto, A.A., Grando, R.B., Silva, C.B. da, Aoki Vivian, M., Cardoso, D.S., Horn, A.C., and Drews, P.L.J., Trajectory Planning for Hybrid Unmanned Aerial Underwater Vehicles with Smooth Media Transition, *J. Intelligent Robot. Syst.*, 2022, vol. 104, no. 46. https://doi.org/10.1007/s10846-021-01567-z

18. Lock, R.J., Vaidyanathan, R., Burgess, S.C., and Loveless, J., Development of a Biologically Inspired Multi-Modal Wing Model for Aerial-Aquatic Robotic Vehicles through Empirical and Numerical Modelling of the Common Guillemot, Uria Aalge, *Bioinspirat. Biomimetics*, 2010, vol. 5, no. 4, pp. 1–15. https://doi.org/10.1088/1748-3182/5/4/046001

19. Wu, Y., Li, L., Su, X., and Gao, B., Dynamics Modeling and Trajectory Optimization for Unmanned Aerial-Aquatic Vehicle Diving into the Water, *Aerospace Sci. Technol.*, 2019, vol. 89, pp. 220–229. https://doi.org/10.1016/j.ast.2019.04.004

20. Liu, S., Ozay, M., Okatani, T., Xu, H., Sun, K., and Lin, Y., Detection and Pose Estimation for Short-Range Vision-Based Underwater Docking, *IEEE Access.*, 2019, vol. 30, no. 7, pp. 2720–2749. https://doi.org/ 10.1109/ACCESS.2018.2885537

21. Cowen, S., Briest, S., and Dombrowski, J., Underwater Docking of Autonomous Undersea Vehicles Using Optical Terminal Guidance, *Oceans '97. MTS/IEEE Conference Proceedings*, Halifax, NS, Canada, 1997, vol. 2, pp. 1143–1147. https://doi.org/10.1109/OCEANS.1997.624153

22. Negre, A., Pradalier, C., and Dunbabin, M., Robust Vision-Based Underwater Homing Using Self-Similar Landmarks, *J. Field Robot.*, 2008, vol. 25, no. 6–7, pp. 360–377. https://doi.org/10.1002/rob.20246

23. Ghosh, S., Ray, R., Vadali, S.R.K., Shome, S.N., and Nandy, S., Reliable Pose Estimation of Underwater Dock Using Single Camera: A Scene Invariant Approach, *Machine Vision Appl.*, 2016, vol. 27, no. 2, pp. 221–236. https://doi.org/10.1007/s00138-015-0736-4

24. Li, Y., Jiang, Y., Cao, J., Wang, B., and Li, Y., AUV Docking Experiments Based on Vision Positioning Using Two Cameras, *Ocean Engineer.*, 2015, vol. 110, pp. 163–173.

25. Chichkarev, E., Sergienko, A., and Balalaeva, E., Using Machine Learning Models and Deep Learning Networks to Recognize Handwritten Numbers and Letters of the Russian and Latin Alphabets, *Inter-Conf.*, Prague, Czech Republic, 2021, pp. 363–380. https://doi.org/10.51582/interconf.21-22.11.2021.044 (In Russian)

26. Sirota, A.A., Mitrofanova, E.Yu., and Milovanova, A.I., Analysis of Algorithms for Searching Objects in Images Using Various Modifications of Convolutional Neural Network, *Proceedings of Voronezh State University. Series: Systems Analysis and Information Technologies*, 2019, no. 3, pp. 123–137. https://doi.org/10.17308/sait.2019.3/1313 (In Russian)

27. Zou, Z., Shi, Z., Guo, Y., and Ye, J., Object Detection in 20 Years: A Survey, *Proceedings of the IEEE*, 2019, no. 111, pp. 257–276.

28. Kletenik, D.V., *Sbornik zadach po analiticheskoi geometrii* (Collection of Problems in Analytic Geometry), Moscow: Nauka, 1998.

29. Gonzalez, R.C. and Woods, R.E., *Digital Image Processing*, 3rd ed., London: Pearson, 2007.

30. Hough, P.V.C., Machine Analysis of Bubble Chamber Pictures, *2nd International Conference on High-Energy Accelerators and Instrumentation, HEACC 1959*, CERN, Geneva, Switzerland, 1959, pp. 554–558.

31. Lyasheva, S., Shleymovich, M., and Shakirzyanov, R., The Image Analysis Using Fast Radial Symmetry Transform in Control Systems Base on the Computer Vision, *International Multi-Conference on Industrial Engineering and Modern Technologies, FarEastCon*, 2019, pp. 1–6. https://doi.org/10.1109/FarEastCon.2019.8934298

32. Shakirzyanov, R.M., Detection of Traffic Signals Using Color Segmentation and a Radial Symmetry Detector, *Bulletin of the Voronezh State Technical University*, 2020, vol. 16, no. 6, pp. 25–33.

33. Adelson, E., Burt, P., Anderson, C., Ogden, J.M., and Bergen, J., Pyramid Methods in Image Processing, *RCA Engineer.*, 1984, vol. 29, no. 6, pp. 33–41.

34. O'Malley, R., Jones, E., and Glavin, M., Rear-Lamp Vehicle Detection and Tracking in Low-Exposure Color Video for Night Conditions, *IEEE Transactions on Intelligent Transportation Systems*, 2010, vol. 11, no. 2, pp. 453–462. https://doi.org/10.1109/TITS.2010.2045375

35. Druki, A.A., Algorithms for Selecting Faces on Static RGB Images and in a Video Stream, *Izvestiya TPU*, 2012, no. 5, pp. 65–69. (In Russian)

36. Darge, A., Rajendran, R.S., Zerihum, D., and Chung, P.Y.K., Multi Color Image Segmentation using L*A*B* Color Space, *Int. J. Advanced Engineer., Management Sci.*, 2019, vol. 5, pp. 346–352. https://doi.org/10.22161/ijaems.5.5.8

37. Forsyth, D. and Ponce, J., *Computer Vision: A Modern Approach*, London: Pearson, 2012.

38. Droogenbroeck, V.M. and Barnich, O., Design of Statistical Measures for the Assessment of Image Segmentation Schemes, *Proceedings of 11th International Conference on Computer Analysis of Images and Patterns (CAIP2005), Lecture Notes in Computer Science*, Rocancourt, France, 2005, vol. 3691, pp. 280–287.

39. Koltsov, P.P., Osipov, A.S., Kutsaev, A.S., Kravchenko, A.A., Kotovich, N.V., and Zakharov, A.V., On the Quantitative Performance Evaluation of Image Analysis Algorithms, *Computer Optics.*, 2015, vol. 39, no. 4, pp. 542–556. (In Russian)

This paper was recommended for publication by O.A. Stepanov, a member of the Editorial Board