

THE VISION SYSTEM SOFTWARE TO ENABLE MANEUVERING OF UNDERWATER VEHICLE

Ivan Vasilyev¹, Alexander Popov¹, Ekatherina Smirnova¹,
Oleg Shmakov² & Vasily Tselouyko¹

¹ Russian State Scientific Centre for Robotic and Technical Cybernetics, St. Petersburg, Russia

² Peter the Great St. Petersburg Polytechnic University, St. Petersburg, Russia



This Publication has to be referred as: Vasilyev, I[van]; Popov, A[lexander]; Smirnova, E[katerina]; Shmakov, O[leg] & Tselouyko, V[asily] (2018). The Vision System Software to Enable Maneuvering of Underwater Vehicle, Proceedings of the 29th DAAAM International Symposium, pp.0540-0547, B. Katalinic (Ed.), Published by DAAAM International, ISBN 978-3-902734-20-4, ISSN 1726-9679, Vienna, Austria
DOI: 10.2507/29th.daaam.proceedings.078

Abstract

High-risk underwater operations at industrial facilities requires for underwater robots. Many works (inspection, docking with manifolds, etc.) are associated with operations in an environment with obstacles. While developing the corresponding control of the underwater vehicle (UV), it is important to take into account the dynamic parameters of the UV and ensure the accuracy of its positioning. In the article the UV for work on manmade objects is considered. For the positioning we consider artificial markers and television vision system (VS). The safe maneuvering of the UV requires rapid processing of the video stream. Limited computing resources of UV in combination with the complexity of the task determine the increased requirements for the organization of the computing process. The architecture of the software offered by the authors provides flexibility in the implementation of different scenarios and effective computational procedures. In addition to television cameras, VS of UV includes sonars, echo sounders, etc. The proposed architecture provides a unified approach to the sensors. Also are described the mathematical model of UV, been developed for debugging of the software and results of computational experiment, which simulated UV bringing in a target point which coordinates are defined by means of VS.

Keywords: underwater vehicle; robotic operations; technical vision; software

1. Introduction

Maintenance, inspection of the technical condition of the equipment, monitoring and other similar operations at the sea-based industrial facilities are associated with the work of personnel in an environment of increased danger. A natural consequence of the growth in the volume of such work is the increase in the number of underwater robots. Many works under water are associated with operations in limited and cramped conditions.

Such operations include: inspection of underwater structures, docking with manifolds, etc. In these conditions it is necessary to synthesize control of the underwater vehicle taking into account safety of its maneuvering. In the algorithms of safety and maneuvering are decisive on the one hand, the account of the dynamic parameters of the control object, on the other hand, the accuracy of its positioning relative to other objects.

A number of solutions to the problem of positioning an Autonomous robot with respect to obstacles are known, when technical vision systems (VS) based on different physical principles are used as the main sensor [1], [2], [3], [4], [5]. When carrying out underwater operations on man-made objects, it can be assumed that the robot performs operations in a prepared environment containing markers and other auxiliary means. In this case, the main sensor of technical vision are television cameras that distinguish markers at a distance of 20-25 meters.

The most difficult to implement are those tasks where the VS data are used directly in the control loop when driving in an environment with obstacles. Taking into account the relatively low speed of the UV when maneuvering among obstacles, to ensure the safe movement of the UV requires on-line processing of the video stream at a speed of no more than 5-8 fps. However, the limited computing resources of UV in combination with the complexity of the algorithms determine the increased requirements for the organization of the computing process. The architecture of the software of VS should provide both flexibility in the implementation of various scenarios and effective implementation of computational procedures.

In addition to television cameras, VS of UV (UVVS) may include sonar sensors (SS), echo sounders and some other sensors. The software architecture should provide a unified methodological approach to all sensors, which allows to develop the software complex of UV technical vision on the basis of unified solutions.

2. Structure of the UVVS technological operations

The general approach to the development of the UVVS software architecture is based on the robot operation frame architecture [6]. This approach provides both the possibility of operational formation of scenarios for the use of VS, and the quality of debugging of individual typical technological operations (TOP).

We list the following VS TOPs, which together provide all the necessary functions:

- Sensor settings. When working with TV cameras, the TOP "TV settings" allows you to adjust the camera sensor parameters (shutter speed, gain, resolution, automatic exposure, color mode), the TOP "SS settings" allows to select the mode of sonar operation. Similar procedures are created for sounders and other VS sensors available on the UV.
- Accumulation of data. When working with TV cameras, the TOP "TV Accumulation" records TV frames in jpg format, records a video file in avi format and stores metadata (geographical coordinates, depth, date, time, etc.) while receiving frames. TOP "SS Accumulation" records the data obtained from SS and metadata to the file in xtf format. . Similar procedures are created for other VS sensors available on the UV.
- Positioning. TOP "Positioning" calculates the relative position of UV in the required coordinate system, for example, in the coordinate system of markers placed on the underwater object. The calculation involves video data received from TV cameras and data of the UV navigation system.
- Search. TOP "Search for elongated objects" searches the image of elongated objects (underwater cables, pipelines). TOP "Search for point objects" searches relatively compact underwater objects, for example, manifold units. TOP "Search for a given image" searches for an object by its image from the database.

3. UVVS input and output data

Consider a typical case when the input data of the VS are formed by television cameras, sonar (SS) and navigation system of the UV. When using VS in the control loop of the UV, the results of the data processing algorithms may be the transition matrix from the system of coordinates (SC) of the UV into SC of another object (e.g. manifold), geographical coordinates of detected objects, the direction of the detected elongated object, etc. In addition, the output data may be the data stored on a solid-state drive: TV frames, video, sonar data with reference to the time and geographical coordinates of the UV

4. UVVS software architecture

The software is built on a modular principle, which provides expandability and flexibility of the system. Data processing is performed in parallel in two or more channels, which allows one to receive and process data from different devices with different frequencies. For example, one channel processes data received from TV, the second one processes data received from SS, etc. The structure of the top-level modules is shown in figure 1.

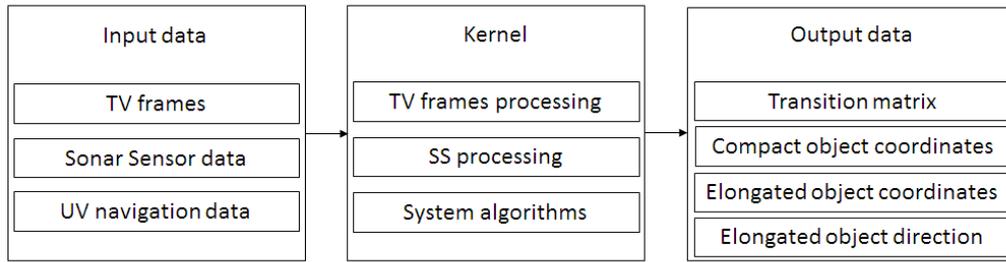


Fig. 1. The structure of top-level UVVS software modules

5. Description of the computing kernel operating

A computing core is a Software package for the development of vision and decision-making systems with dynamically changing modular architecture [4], adapted to the problem under consideration. The main tasks of the computing core are to ensure independent operation of data processing channels and control of data processing modules.

When the system is initialized, the kernel builds two independent computational models that include all the necessary data processing modules. Depending on TOP, the core establishes communication between the modules and starts the cycle of data processing in the necessary channel. The computational kernel allows the simultaneous execution of all the TOP, which is a consequence of the lack of mutually exclusive modules. Launching a new TOP will add links to the model, activate the calculation of new modules and increase the counter of the TOP performed in a particular channel. When you receive a signal about the need to stop the execution of any TOP, the activation of the calculation of the used modules is reset, provided that they are not used during the execution of other active TOP.

The data processing cycle will end if the counter of the running TOP is zeroed or in case of an emergency situation. In case of an emergency, an error signal will be sent to the decision-making system. A key feature of the kernel is its independence, which allows you to debug the algorithms, regardless of the presence of the device. The same code is used on board the UV and in the simulator simulating the operation of the device. No less important task of the kernel is to log the work of information processing modules. In debug mode, the kernel is able to store structured information about the data transmitted within the system, namely the sequence of input and output data of each data processing module involved in the calculation in the order of their activation. This feature significantly increases the speed of debugging due to the rapid localization of the module in which the calculation error occurred.

6. Software modules structure description

Consider the structure of software modules that implement the functions of the TOP, which were described above, in section 4. Structure to perform the Sensors setting and Data accumulation is trivial. It consists of communication modules with appropriate sensors and recording modules to the file system. The most difficult task of the VS is to determine the spatial position of the UV relative to another object. The structure of the modules for the TOP Positioning is shown in figure 2.

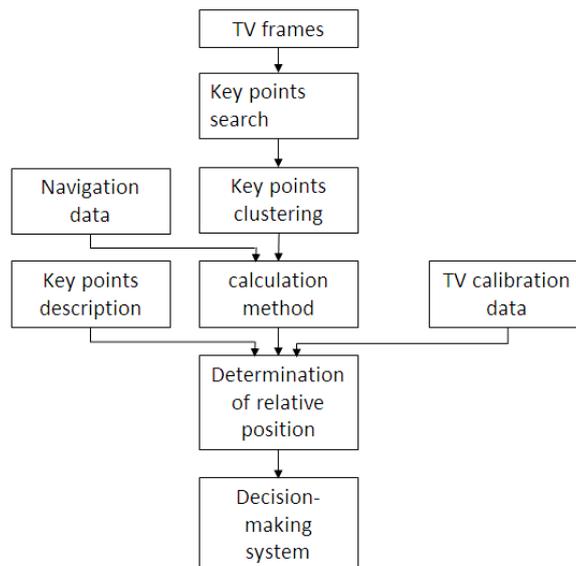


Fig. 2. The structure of the TOP Positioning modules

The peculiarity of this structure is the use of the calculation method selection module, which allows to expand the operating range of the system. It is necessary due to the different location of key points on various elements of the underwater infrastructure and the different location of the cameras on different UV. For example, to determine the relative position of the UV, it is possible to implement a method, using two key points on the TV frame and known angles of UV roll and trim, or a method, using a set of points on the TV frame (at least 4) or other methods of combining visual and navigation information [7]. The proposed structure of the software allows to fulfill the TOP Positioning, using both one and several TV cameras.

Calibration data carry information about the internal and external calibrations of the cameras. External calibration is calculated from the geometry of the UV, the internal one is calculated in accordance with standard algorithms for the calibration of television cameras.

The structure of the sonar data processing modules is shown in figure 3. Dashed lines indicate the connection with the modules, responsible for the execution of various TOP. The elongated object search and tracking software modules are instances of the same class that differ only in their parameters. During the search missions UV navigation system data are used as well (e.g. the distance from the bottom and the UV speed).

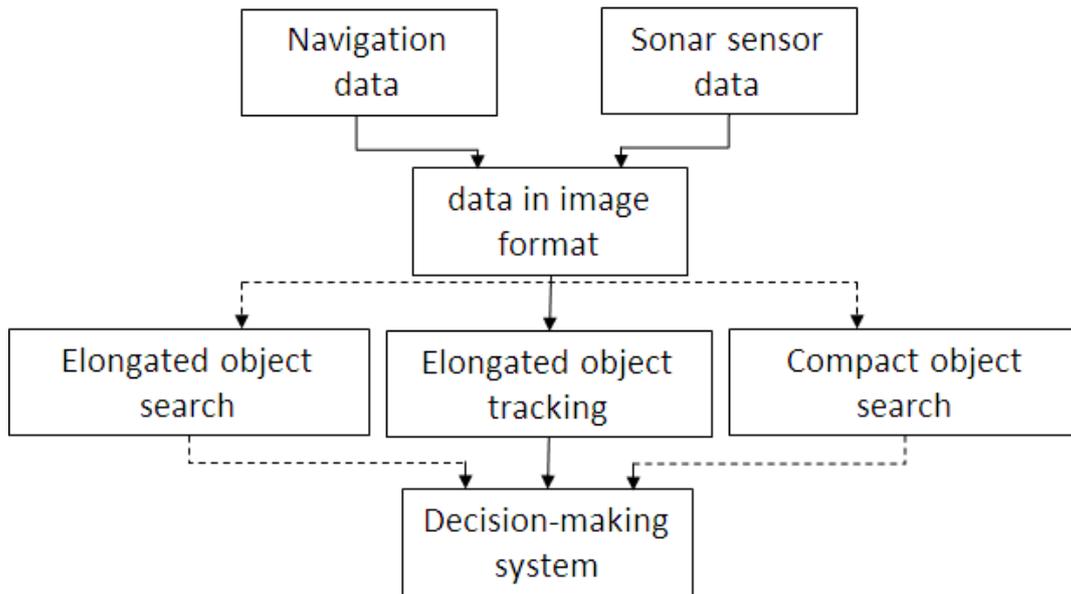


Fig. 3. The structure of the sonar data processing modules

7. Data structure description

The following is a description of the input and output format shown in figure 1. Television frames are represented as a three – dimensional array of bytes with dimension $h \times w \times c$, where h is the height of the image, w is the width of the image, c is the number of color channels. The elements of the array take on values from 0 to 255.

Sonar data are presented as a one – dimensional array of bytes with dimension n , where n is the number of sent/received signals of the sonar, which depends on the SS operation mode. The elements of the array take on values from 0 to 255. The data of the navigation system are the UV roll and trim angles, the UV speed, the distance to the bottom, the speed of sound in the environment, the UV geographical coordinates. All data is represented as signed floating-point numbers.

The transition matrix from UV SC to the SC of another object (e.g. manifold) has a dimension of 4×4 and its values are represented as signed floating-point numbers. The coordinates of a compact object are its geographic coordinates as two floating-point numbers. The coordinates of an elongated object are the geographical coordinates of its current calculated center in the form of two floating-point numbers. The course of the elongated object is its direction relatively to the UV current direction. The course is represented as a sign number with a floating point.

8. UV modeling and computational experiment

A simplified computer model of highly maneuverable UV, similar to the apparatus, been described in [8], was developed for the experiments. A specific feature of such UVs is the absence of rudders, which were replaced with thrusters. Figure 4 shows the forces applied to such a device: thrust force, inertia force, resistance force, buoyancy force and gravity.

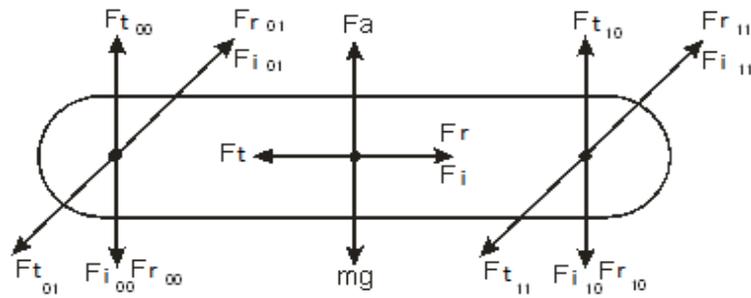


Fig. 4. Forces applied to the UV

Buoyant force is denoted F_a , the force of gravity - mg , the thrust force - F_t , the resistance force - F_r , the force of inertia - F_i . Meaning of indices: 00 - vertical forces of the bow thruster, 01-horizontal forces of the bow thruster, 10-vertical forces of the stern thruster and 11 – horizontal forces of the stern thruster .

Using theoretical mechanics methods let's find the total force and the total moment of forces, applied to the UV (see figure 5). In order not to clutter up the picture, the thrust force of the main marching propulsion is not shown.

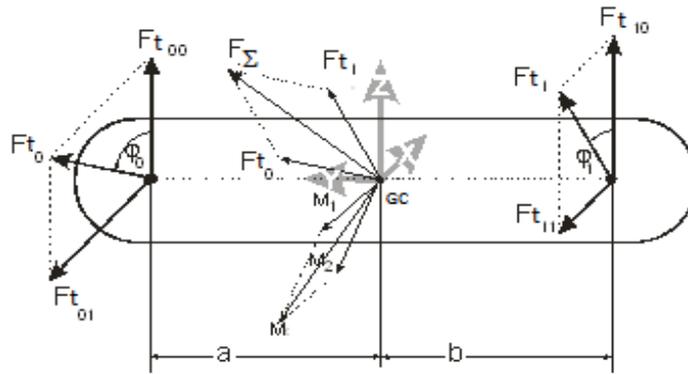


Fig. 5. Scheme of forces and moments summation

Therefore, we can write the sum of forces and moments in vector form:

$$\begin{aligned} \vec{F}_{\Sigma} &= \vec{F}_{t00} + \vec{F}_{t01} + \vec{F}_{t10} + \vec{F}_{t11} + \vec{F}_t + \vec{F}_A \\ \vec{M}_{\Sigma} &= \vec{a} \times \vec{F}_{t00} + \vec{a} \times \vec{F}_{t01} + \vec{b} \times \vec{F}_{t10} + \vec{b} \times \vec{F}_{t11} \end{aligned} \quad (1)$$

From here we obtain projections of forces and moments:

$$\begin{aligned} F_{\Sigma x} &= F_{t01} + F_{t11} \\ F_{\Sigma y} &= F_t \\ F_{\Sigma z} &= F_{t00} + F_{t10} \\ M_{\Sigma x} &= a F_{t00} - b F_{t01} \\ M_{\Sigma y} &= 0 \\ M_{\Sigma z} &= a F_{t01} - b F_{t11} \end{aligned} \quad (2)$$

The found forces and moments are required for substitution in the equation of body hydrodynamics forces:

$$A \dot{y} = U - C y^2, \quad y(t_0) = y_0 \quad (3)$$

where

- A - analogue of the inertial coefficient (mass and inertia moment);
- y - analogue of linear and angular velocities, hereinafter referred to as the generalized velocity;
- U - control action (force and torque);
- C - analogue of the hydrodynamic resistance coefficient (for forward and rotational movements).

Equation (1) has an analytical solution, but it is more convenient to use numerical methods to solve this differential equation.

Let us now consider the simulation of movement by a predetermined distance s . To fulfill this movement the following four conditions are required to be met:

1. the time integral of the velocity must be equal to the given displacement s ;
2. the speed and acceleration of the UV must be zero at the start and the end of movement;
3. the speed of the drive effort change shall not exceed the maximum during the entire period of movement;
4. the drive force must be zero at the start and the end of movement.

Other conditions for approximate modeling are not imposed, therefore, any function satisfying these four conditions can be chosen as the law of velocity change. Obviously, the graph of the velocity function should be bell-shaped (see figure 6). For this purpose, it is proposed to use a fourth-order polynomial of the generalized velocity versus time:

$$y(t) = \frac{b}{a^4} ((t - a)(t + a))^2 \quad (4)$$

As a result of applying the above four conditions to this equation and substituting it in (1), we obtain an expression for the control action:

$$U(t) = \frac{b^2(-225 \cdot s^2 + 256 \cdot b^2 t^2) \cdot (51840000 \cdot A \cdot b \cdot s^4 t - c \cdot (225s^2 - 256 \cdot b^2 t^2)^3)}{2562890625 \cdot s^8} \quad (5)$$

This expression is universal because it contains both a hydrodynamic coefficient C and an inertial coefficient A . Integer coefficients are the consequence of choosing the 4th degree polynomial.

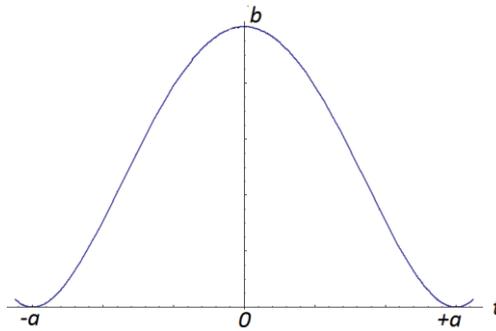


Fig. 6. Graph of the generalized velocity function when moving the UV at a distance s

Consider the UV acceleration to a given speed. It requires the following five simple conditions to be met, which are similar to the conditions, required to met for movement:

1. the velocity and its derivative must be equal to zero at the start;
2. the speed must have given value and the acceleration must be equal to zero at the end of acceleration;
3. the force and its derivative must be equal to zero at the start;
4. force must correspond to the speed at the end of acceleration;
5. the force derivative must not exceed the maximum value.

We write an expression of derivative that satisfies the first two conditions: $t^2(t - a)^2$. For $t = 0$ and $t = a$ this expression is strictly zero.

Let's integrate the derivative (see the graph in figure 7):

$$\int t^2(t - a)^2 dt = \frac{a^2 t^3}{3} - \frac{a \cdot t^4}{2} + \frac{t^5}{5} \quad (6)$$

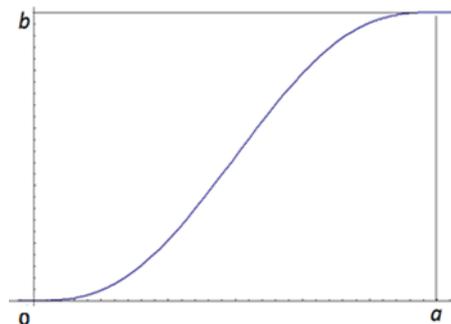


Fig. 7. The graph of speed when accelerating the UV

Then substitute this function in (1) and get an expression for the control action:

$$U(t) = \frac{30b(a^5 A \cdot (a - t)^2 t^2 + \frac{1}{30} b \cdot c \cdot t^6 (10a^2 - 15a \cdot t + 6t^2)^2)}{a^{10}} \quad (7)$$

Derive in a similar way control formula for braking. In contrast to the acceleration and movement processes, the braking may start with arbitrary initial values of speed and drive efforts.

Let us formulate the conditions of braking:

1. there is a non-zero initial velocity and a non-zero derivative of the velocity at the start of braking;
2. the speed, acceleration, force must be equal to zero at the end of braking;
3. the force derivative shall not exceed the maximum permissible values while braking;

For the velocity derivative let us consider an equation of second degree:

$$\dot{v}(t) = aa \cdot t^2 + bb \cdot t + cc \quad (8)$$

Performing all the mentioned conditions and substituting the results in (1), we obtain the control formula for braking:

$$U(t) = \frac{A(a-t)(-6b \cdot t + a(a-3t)vs0)}{a^3} + \frac{c(a-t)^4(2b \cdot t + a(b+t \cdot vs0))^2}{a^6} \quad (9)$$

Computer experiments were carried out, using the described model. Synthesized images of markers indicating the target point of the PA trajectory were used to determine the UV current position. An example of the synthesized image of the marker described in [3] is shown in figure 8.

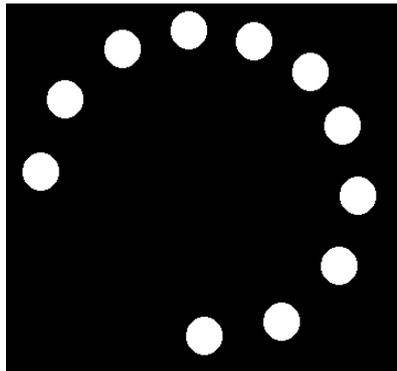


Fig. 8. An example of a synthesized marker for UV maneuvering according to VS data

During the computational experiment, the following parameters were set: the start distance for bringing the UV to the target point - 20 meters, the initial positioning accuracy - 1.5 m at a distance of 20 meters, the final positioning accuracy - 0.1 m at a distance of 1 m. Image processing was carried out on a single board computer based on Intel Atom processor with a clock frequency of 1.66 GHz. Due to the above-described structural solutions, data processing speed of at least 8-10 fps was achieved. At the maximum set speed of 0.2 m/sec, this ensured a 100% successful UV bringing to the target point on the model.

9. Conclusion

The considered architecture of the VS software provides all the necessary operations to solve functional problems of UVVS. The developed software prototype was tested on a computer model of UV and confirmed its applicability to solve the problem of UV positioning when working on man-made offshore facilities, where constant maneuvering is required. The simulation results confirmed the possibility of successful solution of this problem with acceptable speed (about 8 fps) and the required accuracy.

In the future, the proposed architecture and the developed software prototype can be used to solve other problems of underwater vehicles FTZ due to their flexibility and extensibility.

10. Acknowledgments

The results were obtained in the framework of the state task of the Ministry of education and science of Russia. Project 8.13276.2018/12.1

11. References

- [1] Inzartsev, A.V. and others (2015) Monitoring of the seabed with the use of intelligent data processing technologies of search devices on board the AUV //Underwater research and robotics. № 2(20). C. 20-27.
- [2] Vorontsov, A.V. and others (2009) Some algorithms of bringing a small AUV to the seabed dock-station in the near zone, based on visual information processing// Technical problems of the world ocean development: materials of the third all-Russian scientific and technical conference Vladivostok: Dalnauka, 2009. C. 399-404.
- [3] Boreiko, A.A. and others (2010) Video processing algorithms for solving some problems of control and navigation of Autonomous unmanned underwater vehicles //Underwater research and robotics. 2010. № 1(9). C. 29-39.
- [4] Park, Jin-Yeong & Jun, Bong-Huan & Lee, Pan-Mook & Oh, Junho. (2009). (2008) Experiments on vision guided docking of an autonomous underwater vehicle using one camera. Ocean Engineering. 36. 48-61. 10.1016/j.oceaneng.2008.10.001.
- [5] Eustice R., Pizarro O., Singh H. (2008) Visually Augmented Navigation for Autonomous Underwater Vehicles//IEEE J. Oceanic Eng. 2008. Vol. 33, No. 2. P. 103-122.
- [6] Lopota, A.V. and others (2013) Results and promising areas of research in the field of navigation and control of mobile robotic systems // Space crafts and technology 2013. №2 (4). URL: <https://cyberleninka.ru/article/n/osnovnye-rezultaty-i-perspektivnye-napravleniya-issledovaniy-v-oblasti-navigatsii-i-upravleniya-mobilnymi-robototekhnicheskimi> (11.09.2018)
- [7] Bakhshiev, A.V., Michailov, V.V. (2016) Software package for development of vision and decision-making systems with dynamically changing modular architecture// Scientific and Technical Journal «Robotics and Technical Cybernetics» №4(13), 2016. – P. 38-42
- [8] Valavanis, K.P. and others (1997) “Control architecture for autonomous underwater vehicles,” IEEE Control Systems, 17(6), 1997. - P. 48-64
- [9] Smirnova, E.; Stepanov, D. & Goryunov V. (2015) A Technique of Natural Visual Landmarks Detection and Description for Mobile Robot Cognitive Navigation // Annals of 26th DAAAM International Symposium on Intelligent Manufacturing and Automation, DAAAM 2015.