The Development of Laboratory Work on the Topic: Pre-processing of Information from CVS Sensors of a Mobile Rescue Robot in Smoke Conditions

Konstantin S. Bogushev[1] and *Vasily I.* Rubtsov[1*]

[1]Bauman Moscow State Technical University, 2nd Baumanskaya str., 5/1, 105005, Moscow, Russia

Abstract. The solution to the problem of processing long-range and television information received by the sensors of a mobile rescue robot in a smoke-filled environment is considered. A selection of budget sensors is made among those available in the free sale and having open-source software. The selected sensors are linked into a single information field in the free ROS software package using open-source libraries. The first stage of processing is the calibration of sensors to reduce the effect of distortion, as well as comparing the color image of the television camera with the readings of the rangefinder. The second stage is the analysis of existing solutions for image filtering in smoke conditions and the selection of the best according to the criteria for reducing the number of "smoke-filled" pixels and speed of response. In this paper, an algorithm is selected based on an atmospheric physical model with image analysis in the YCrCb space. The operation of this algorithm is demonstrated and a method for approximating a long-range image using a filtered color image is proposed to restore information from a rangefinder and further construct a model of the environment. Suggestions were made for further analysis and improving the accuracy of the algorithm. Based on this decision, laboratory work was formed in the course “RS designing”.

1 Introduction

The current pace of robotics development shows that the creation of autonomous mobile rescue robots is only a matter of time. The relevance of creating such robots is justified by the significant reduction in risks for professional rescuers, as well as the potential physical superiority of robots over people: partial or complete immunity to fire, smoke or cold, the ability to develop great powers or moments, etc. All this makes them preferable to use in case of fire, the collapse of buildings and debottleneck, work in the avalanche zone.

However, the introduction of such robots into the actions of rescuers is difficult due to the difficult situation in the emergency zone: non-deterministic environment, reduced visibility, etc. The solution to this problem lies primarily in the development of algorithms for the computer vision system (CVS) of a mobile robot [1-2].

* Corresponding author: rubtsov@mail.ru

Conducting rescue operations under the fire conditions a particular difficulty for the CVS of the rescue robot: in addition to direct open fire, which can affect the operation of the robot, a large amount of smoke is emitted in burning rooms, which leads to a sharp deterioration in the visibility conditions and a decrease in the efficiency of the CVS due to following factors:

1) The visual picture taken by the television cameras is “smoke-filled”, i.e. has an incorrect color display and distorts the shapes of objects, or the object is only partially observed or not observed at all;
2) The rays of laser rangefinders are reflected from smoke particles and either simply do not return to the receiver or provide false information about the distance to objects;
3) The thermographic cameras also give an incorrect display of the external environment due to the presence of foci of an open flame.

2 Problem statement

This paper sets out the option of restoring and approximating long-range and visual information through the use of filters and algorithms that improve visibility in smoke conditions for further use on pattern recognition neural networks.

The tasks in the current work are:

1) The choice of a laser rangefinder for receiving long-range information and the choice of a television camera for obtaining visual information;
2) The connecting of the selected sensors to ROS (Robot Operating System) and receiving information in the form of ROS messages;
3) The integration of sensors for comparing long-range and visual information;
4) The study and application of filtering algorithms on the visual image from the television camera;
5) The correction of the rangefinder data according to the filtered visual information.

3 Sensor selection

As a prototype of the CVS, it is advisable to use a sensor that combines a television camera and a rangefinder. This is due to the following reasons:

1) The primary task is to check the operability of the algorithm, therefore, it is not necessary to choose sensors that are expensive and high-precision;
2) Connecting one sensor instead of two reduces the load on the control system, i.e. only one connection port is needed;
3) The general format for sending frames, and the general calibration, reduce the time for setting up the sensor and connecting it to the control system.

In the budget price category of sensors, which provide both visual and long-range images, those developed for interactive games are the most popular. The seriality of the release and low price lead to the presence of a large number of software libraries for working with these sensors. Of the most common devices in this area, Kinect Sensor 2.0. and Asus Xtion Pro Live were selected [3]. Comparative characteristics in which these sensors differ are presented in table 1:

Table 1. Comparative characteristics of Kinect Sensor 2.0. and Asus Xtion Pro Live.

Sensor	**Kinect Sensor 2.0.**	**Asus Xtion Pro Live**
Power	Additional power supply through a special adapter	USB powered
Weight	450 g	200 g
Dimensions	25x6x6 cm	18x3.5x5 cm
Long-range image	Torn, a lot of "black spots" - lack of readings from the sensor	Steady

Based on the results of the preliminary comparison, the Asus Xtion Pro Live sensor (hereinafter referred to as Xtion) with the characteristics presented in Table 2 and the device appearance shown in Fig. 1 was selected as the final version:

Table 2. Technical documentation for the Asus Xtion sensor.

Parameter	Value
Power consumption	less than 2.5 watts
Scan distance	from 0.8 m to 3.5 m
Viewing angles	58° H, 45° V, 70° D
Sensors	color video camera, depth sensor, 2 microphones
Depth sensor resolution	VGA (640x480): 30 fps QVGA (320x240): 60 fps
Camera resolution	SXGA (1280x1024)

Fig. 1. The appearance of the selected sensor.

4 Sensor connection and calibration

The Robot Operating System (ROS) was chosen as the medium for processing information from Xtion sensors. ROS provides developers with libraries and tools for building robotics applications, provides hardware abstraction, offers device drivers, libraries, visualizers, messaging, package managers, and much more. ROS is released under the terms of the BSD license and is open-source.

To connect the Xtion sensor, the OpenNI2 package was used, which contains libraries for connecting general-purpose long-range cameras, for example, Asus Xtion or Microsoft PrimeSense, which use the OpenNI source code [4].

Before using the sensor, its camera was calibrated using the camera_calibration package [5]. The essence of the calibration was to remove the distortion effect, which causes distortion of straight lines in the image. For this, sheet A3 with a chess markup printed on it, that shown in Fig. 2 was used.

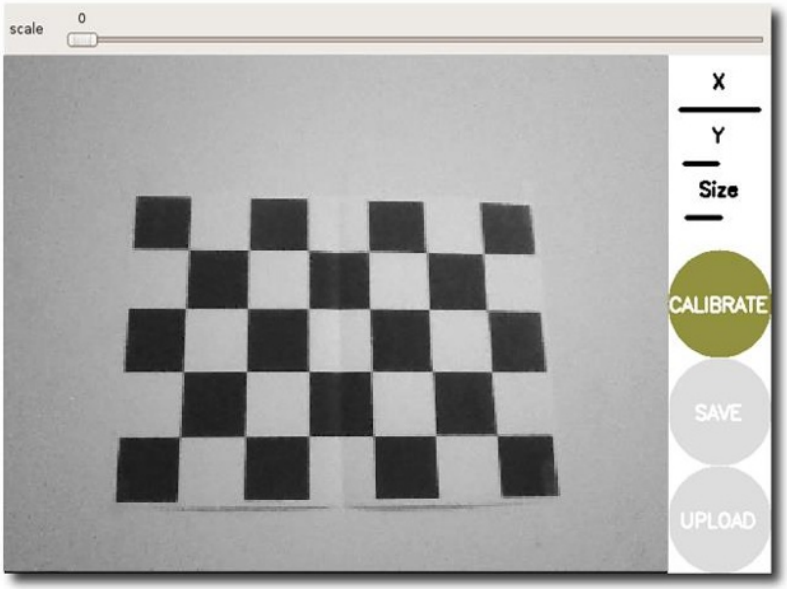

Fig. 2. The window interface of the camera calibration program.

5 Receiving and converting information from sensors

To process images from the Xtion sensor, the data must be converted into a format that is understandable to the programming languages in which the processing algorithms are written. In this work, the programming language is Python, and the library of algorithms is the OpenCV open library. To do this, data from the ROS message format are converted to arrays of the numpy.array type.

Next is the integration of two images. The integration of long-range and television images is carried out in two stages. At the first stage, for each elementary long-range measurement (long-range pixel), there is a correspondence in the space of the video frame, and for adjacent measurements, the geometric and color distances are determined, taking into account which, at the second stage, approximating elementary faces of the wireframe model will be constructed. The correspondence between the pixels of the video frame and the measurements of the rangefinder can be found by simply moving and rotating the coordinate system of the rangefinder into the coordinate system of the video camera [7-8]:

$$\begin{bmatrix} x_{cam} \\ y_{cam} \\ z_{cam} \end{bmatrix} = A(\Delta\alpha, \Delta\beta, \Delta\gamma) \begin{bmatrix} X_{LD} \\ Y_{LD} \\ Z_{LD} \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix} \quad (1)$$

Obviously, the integration uses only the mutual intersection of the viewing areas of the long-range and television sensors.

The next step is the processing of the image in the form of the use of smoke filters on a visual image, followed by the conversion of long-range data. After that, the arrays of changed data are converted back to the corresponding ROS message formats.

At the second stage, a three-dimensional wireframe model is built from the long-range image and the corresponding television frame is “pulled” onto it. The result is a data format such as a cloud of point, which is demonstrative during visualization, and also allows the mobile robot to get an idea of the environment.

The relationship graph of the described ROS message path is shown in Fig. 3:

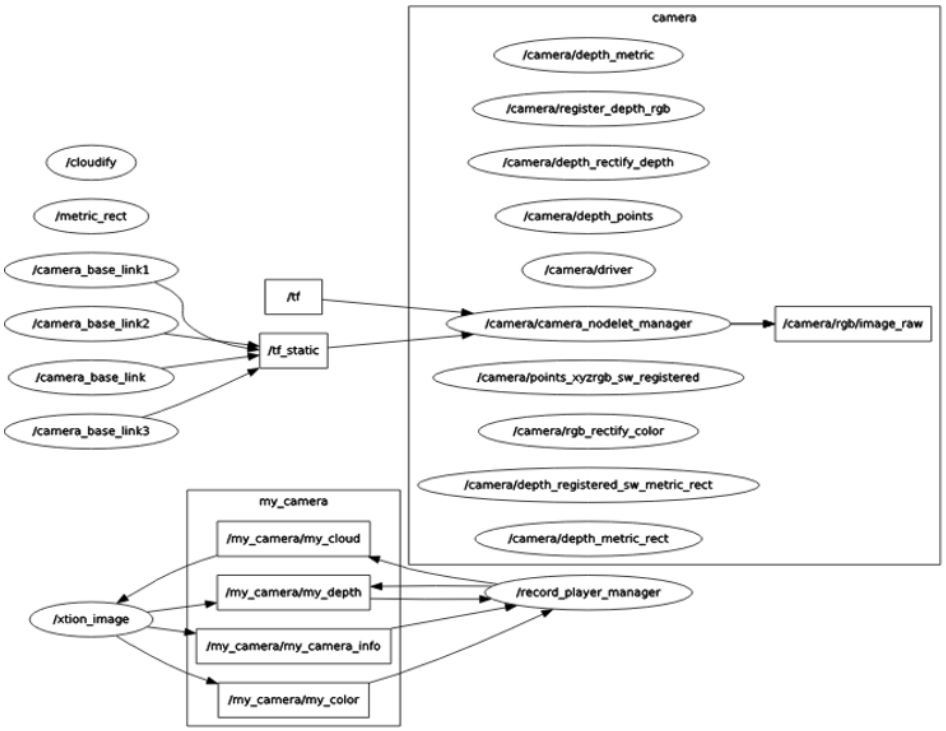

Fig. 3. The graph of dependencies between ROS nodes.

6 Image filtering

As noted earlier, the use of the rangefinder without preliminary processing of the received information in smoke conditions is impossible due to the effect of smoke particles on the laser measuring rays. In the work on the creation of the SmokeBot robot, the authors indicate two types of exposure to smoke on the rays of the laser rangefinder [9]:

1) The rays are reflected from the smoke particles and return incorrect readings about the long-range information in the observation scene;
2) The rays are absorbed by smoke, not returning to the receiver of the rangefinder, which leads to the absence of long-range information in principle.

A schematic representation of these smoke effects is shown in Fig. 4.

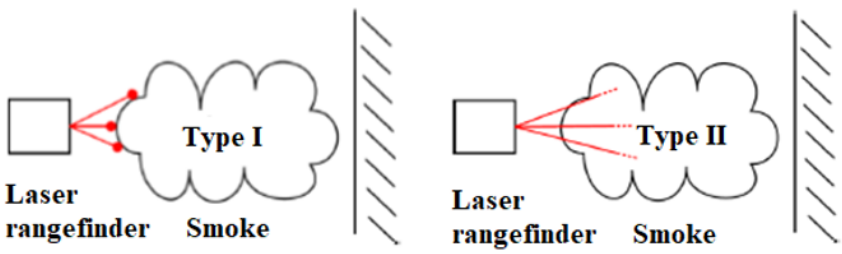

Fig. 4. The effect of smoke on the readings of the rangefinder.

To process information in such conditions, the development team used sophisticated algorithms, including the integration of sensors such as 3D LiDAR, radar, and a high-definition television camera. This approach gives the best results on the criterion of accuracy and correctness of obtaining environmental information, however, it requires the presence of expensive sensors and large computing power to process all incoming information and apply algorithms.

In the case of filtering exclusively on a television picture, two main methods are proposed:

1) Improving image quality by aligning the histogram;
2) Construction of an atmospheric physical model.

Work with a histogram involves averaging a certain value, taken as the basis for filtering, expanding the boundaries on the histogram, or cutting off an uninformative part of it.

The construction of a physical model occurs according to the following formula:

$$I(x,y) = t(x,y)J(x,y) + A\big(1 - t(x,y)\big) \quad (2)$$

where I – is a smoke-filled image; J – represents the intensity of the light reflected from the object (i.e., a "clean" image, which must be restored); A – is the global intensity of illumination in the scene; t – is the transparency coefficient;

Figure 5 shows a generalized diagram of an atmospheric physical image model.

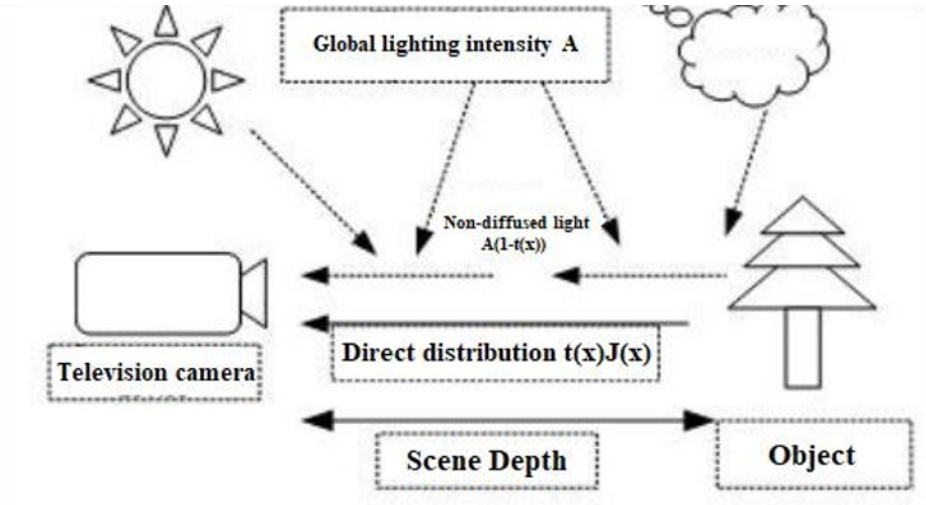

Fig. 5. A generalized diagram of the atmospheric physical image model.

The first method is used by CohuHD company in their cameras of the Helios line [10]. According to the company's documentation, the algorithm works in real-time and includes two stages: assessing the smoke content of the image and then applying the histogram transformations. An example of the algorithm is presented in Fig. 6.

Fig. 6. Image processing by CohuHD 3960HD Camera.

However, the company does not provide free-access processing algorithms, and the description in the official documents does not contain specific methods, but only indirectly indicates the order and applications.

From sources with an open-source or a detailed description of the filtration steps for the first method, the papers [11–13] were selected, and using the second method, the papers [15–17].

All selected algorithms were previously compared according to two criteria: the percentage of image improvement (the number of pixels that restored color from the total number of "smoke-filled" pixels) and the processing time of a single frame.

To check the percentage of improvement, the images were previously converted to HSV format (Hue - tone [0..360], Saturation - saturation [0..1], Value - brightness [0..1]). In this paper, "smoke-filled" pixels are accepted whose saturation is less than 0.2. Recovered color - those pixels whose saturation is more than 0.4. A visual representation of the HSV color palette is shown in Fig. 7.

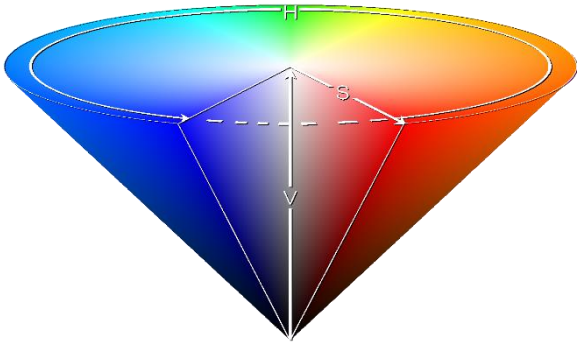

Fig. 7. Visual representation of the HSV space.

The transition from the RGB space to the HSV space is carried out by the following transformations:

Let R, G, B be measured from 0 to 1, and MAX and MIN are the maximum and minimum values among R, G, B, respectively. Then:

$$H = \begin{cases} 0, \text{if } MAX = MIN \\ 60 * \frac{G-B}{MAX-MIN} + 0, \text{if } (MAX = R) \cap (G \geq B) \\ 60 * \frac{G-B}{MAX-MIN} + 360, \text{if } (MAX = R) \cap (G < B) \\ 60 * \frac{G-B}{MAX-MIN} + 120, \text{if } MAX = G \\ 60 * \frac{G-B}{MAX-MIN} + 240, \text{if } MAX = B \end{cases} \quad (3)$$

$$S = \begin{cases} 0, \text{if } MAX = 0 \\ 1 - \frac{MIN}{MAX}, \text{if } MAX \neq 0 \end{cases} \quad (4)$$

$$V = MAX \quad (5)$$

Based on the selected criteria, it was decided to use an algorithm based on the YCbCr space using an averaging filter [14]. YCbCr is a family of color spaces that are used to transmit color images in component video and digital photo graphics. Y - is the brightening component, C_B and C_R - are the blue and red color-difference components.

The transition formulas from RGB to YCrCb that used in the paper:

$$\begin{cases} Y = 0.257R + 0.504G + 0.098B + 16 \\ Cr = -0.148R - 0.291G + 0.439B + 128 \\ Cb = 0.439R - 0.368G - 0.071B + 128 \end{cases} \quad (6)$$

According to the results of transformations for the filtered image, the following dependencies were obtained:

$$\begin{cases} Y_J = \frac{Y_I - A(1-t(x))}{t(x)} + 16 \\ Cr_J = \frac{Cr_I - A(1-t(x))}{t(x)} + 128 \\ Cb_j = \frac{Cb_I - A(1-t(x))}{t(x)} + 128 \end{cases} \quad (7)$$

Based on the physical model:

$$A\big(1 - t(x,y)\big) \leq I(x,y) \rightarrow t(x,y) \geq 1 - \frac{I(x)}{A} \quad (8)$$

Next, an averaging filter is applied to channel Y:

$$M_{ave}(x,y) = Average_{Sd}\big(I(x,y)\big) \quad (9)$$

where Sd – is the size of the filter window.

Then the average value of the transparency coefficient:

$$t_{ave}(x,y) = 1 - \frac{M_{ave}(x,y)}{A} \quad (10)$$

The true value of t, according to the model, is:

$$t_{actual}(x) = \frac{A - I(x)}{A - J(x)} \quad (11)$$

Thus, the averaged coefficient is less than the true one; therefore, it is necessary to add a term to compensate for the difference:

$$t_{ave}(x,y) = 1 - \frac{M_{ave}(x,y)}{A} + \varphi \frac{M_{ave}(x,y)}{A} \ (0 < \varphi < 1) \tag{12}$$

$$\omega = 1 - \varphi \tag{13}$$

$$t_{ave}(x,y) = 1 - \omega \frac{M_{ave}(x,y)}{A} \tag{14}$$

Too large values of ω lead to small values of t, i.e. very dark image, too small values lead to an overly whitened image. Then you need to enter an adaptive tunable coefficient:

$$\omega = \delta M_{ave}(x,y) \tag{15}$$

After taking into account all the equations, the formula for the optimized coefficient:

$$t_{ave}(x,y) = 1 - \min(\delta M_{ave}(x,y), 0.95) * \frac{M_{ave}(x,y)}{A} \tag{16}$$

The global illumination in the frame is taken as the maximum of the average value for channel Y:

$$A = \max\big(M_{ave}(x,y)\big) \tag{17}$$

An example of the application of the algorithm from the paper is presented in Fig. 8.

Fig. 8. An example of an algorithm based on the YCrCb space.

The idea of applying the filtering algorithm is as follows: Filters and algorithms are applied to the image received from the television camera, reducing the number of "smoke-filled" pixels, increasing the contrast and clarity of the picture. Thus, the "smoke-filled" pixels are repainted in the colors of objects that are supposedly behind the smoke. Further, for points corresponding to "smoke-filled" pixels (Saturation values are less than 0.2), the depth values are reassigned based on the depth values of the neighboring pixels to the repainted ones. The number of neighbors is determined by the size of the comparison window λ, which will be applied to the image. The "new" depth for a pixel should be defined as the largest value of the long-range data for pixels that fall into the comparison window.

7 Result of the filtering algorithm

According to the results of the algorithm, a filtered visual image and a more accurately reconstructed 3D-picture of the environment of the robot based on the readings of the rangefinder are obtained. During the experiment, 50 pairs of images were processed under various conditions of smoke, 37 of them were successfully processed (a filter bar of more

than 20% of “smoke-filled” pixels was reached). Figures 9-11 show the pairs of images before and after filtering, respectively:

Fig. 9. Example No. 1 of the algorithm.

Fig. 10. Example No. 2 of the algorithm.

Fig. 11. Example No. 3 of the algorithm.

8 Laboratory work

Based on this study, laboratory work was formed for students of the Special Engineering 7 “Robotics and Mechatronics” department at the course "RS designing", which is aimed at:

1) Studying the ROS package in working with real sensors;
2) Receiving and processing the readings of the rangefinder and the television camera;
3) Integration and visualization of the data;

4) Application of image filtering algorithms.

Necessary material and technical support:

1) Asus Xtion Pro Live Sensor;
2) A computer with preinstalled software;
3) Marked chess sheet with a format of at least A3;
4) A set of images with smoke-filled scenes.

Required Software:

1) ROS is not lower than the Indigo version;
2) OpenNI2 package corresponding to the installed version of ROS;
3) Python 3.x.;
4) OpenCV 4.x.x.

9 Conclusion

As a result of this work, a sensor was selected that combines a rangefinder and a television camera, the basics of working with the ROS operating system, connecting sensors to it and reading and processing information received from them were studied.

Among the filters that reduce the smokiness of the image, the most optimal efficiency/speed of response ratio was selected; on all tested images the algorithm showed an improvement of 20% or more pixels exposed to smoke. An algorithm for adjusting the long-range information according to the filtered visual image is also implemented.

Thus, the algorithm will allow the mobile rescue robot to interact much more accurately with environmental objects and people, even in smoke conditions. The application of the algorithm is aimed at the evacuation of people and the analysis of local debottlenecks that create obstacles to movement.

The following theses were identified for further improvement of the algorithms, which will be taken into account at the next stage of the final work:

1) The implementation of a more effective selection of tunable parameters δ and λ;
2) The verification of the possibility of working in real-time on stationary computers with a modern generation processor.

References

1. A.V. Vazaev, V.P. Noskov, I.V. Rubtsov, S.G. Tsarichenko, *Integrated CVS in the fire robot control system*, Bulletin of the Southern Federal University, Technical science (2017).
2. V.N. Kazmin, V.P. Noskov, *Formation of geometric and semantic models of environmental media in motion control problems*, Bulletin of the Southern Federal University, Technical science, (2015).
3. Asus Xtion Pro Live: Specifications. URL: https://www.asus.com/en/3D-Sensor/Xtion_PRO_LIVE/specifications/ (accessed: 11/17/2019).
4. Openni2 package for connecting long-range cameras. URL: http://wiki.ros.org/openni2_launch (accessed: 11/10/2019).
5. Camera calibration to minimize distortion. URL: http://wiki.ros.org/camera_calibration/Tutorials/MonocularCalibration (accessed: 11/15/2019).
6. Processing of long-range images in ROS. URL: http://wiki.ros.org/depth_image_proc (accessed: 11/10/2019).

7. V.N. Kazmin, V.P. Noskov, *Isolation of geometric and semantic objects in long-range images for navigating robots and reconstructing the external environment*, Bulletin of the Southern Federal University, Technical science, No.10 (171), pp. 71 – 83 (2015).
8. A.V. Vazaev, V.P. Noskov, I.V. Rubtsov, *An integrated computer vision system in a robot operating system with attachments*, Bulletin of the Tula State University, Technical science, iss. 3 (2018).
9. P. Fritsche, *Mobile Robots with Novel Environmental Sensors for Inspection of Disaster Sites with Low Visibility* (2018).
10. Examples of the operation of Defog algorithms on CohuHD high-resolution cameras. URL: https://www.cohuhd.com/Videos/CohuHD-Video-Gallery/edgpid/28/edgmid/462 (accessed: 11/20/2019).
11. Using the defogging library for image processing. URL: https://pypi.org/project/defogging/ (accessed: 11/20/2019).
12. R.C. Gonzalez, R.E. Woods, *Digital Image Processing Second Edition*, Published by Pearson Education, Inc., pp. 55-62 (2002).
13. D. Liu, W. Liu, Q. Zhao, B. Fei, *Image defog algorithm based on open close filter and gradient domain recursive bilateral filter*, Proc. SPIE 10605, LIDAR Imaging Detection and Target Recognition 2017, 106053T (2017).
14. Sh. Lu, X. Yang, *The Fast Algorithm for Single Image Defogging Based on YCrCb Space*, 2016 3rd International Conference on Smart Materials and Nanotechnology in Engineering (2016).
15. R. Mondal, S. Santra, B. Chanda. *Image Dehazing by Joint Estimation of Transmittance and Airlight using Bi-Directional Consistency Loss Minimized FCN*, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop, pp-920-928 (2018).
16. M.J. Yu, H.F. Zhang, *Single-image dehazing based on dark channel and incident light assumption*, Journal of Image and Graphics, Papers 19(12), 1812-1819 (2014).