International Journal of GEOMATE, April., 2021, Vol.20, Issue 80, pp.82-88
ISSN: 2186-2982 (P), 2186-2990 (O), Japan, DOI: https://doi.org/10.21660/2021.80.GX261
Geotechnique, Construction Materials and Environment

THE NEW ADVANCED PROTOTYPE OF AIRBORNE VISUAL CONTROL OF A GROUND ROBOT

Oleg Kupervasser[1,2*], Hennadii Kutomanov[1], Vitalii Sarychev[2], and Roman Yavich[1]

[1]Department of Mathematics, Ariel University, Israel; [2]Transist Video Llc, Skolkovo, Russia

*Corresponding Author, Received: 30 Oct. 2020, Revised: 13 Jan. 2021, Accepted: 06 Feb. 2021

ABSTRACT: Unmanned autonomous robots will be widely used very soon for land use, treatment, and monitoring. The paper addresses the problem of visual navigation of ground robots using a camera positioned at a certain elevation above the confined area. The main idea is that robot's "eyes" is not located on the robot but are independent autonomous system. As a result, the "eyes" can go up and observe the robot from above. This patented technology of airborne vision-based control of ground robots is already described previously in "International Journal of GEOMATE" and Geomate2019 conference paper (Ehrenfeld et al., 2020). There was described in detail the small toy prototype with very simple algorithms and very shortly the next more complex prototype. In the current paper detailed algorithms for navigation and control of this complex prototype is presented. The system consists of the USB camera on the tower (4 meters) connected to the computer which controls the big ground robot by sending Wi-Fi commands. For navigation (recognition of the ground robot, finding its position and orientation) was used deep learning (artificial neural network). It was demonstrated that this robot can pass two main tests: travelling along some predefined path and random walk inside of some predefined boundary.

Keywords: Vision-based navigation of ground robots, Airborne control from tethered platform, Deep learning convolution network, Time delay of autopilot, Stability of differential equations

1. INTRODUCTION

Unmanned autonomous robots will be widely used very soon for land use, treatment, and monitoring [1-3]. The most popular outdoor robots are currently robot-lawnmowers used for bounded operation area. However, the different very interesting applications for such robots can be found. Let us give yet several examples:

1) It is a remote-controlled slope mower [4]. Currently, remote control of a human operator is used for such mowers. However completely autonomous robots can be made using technology proposed in this paper.
2) Robotic Fruit Harvester - the FFRobot [5]
3) Robots for fire protection [6]
4) Autonomous wheelchair for disabled people [7]

The paper addresses the problem of visual navigation (vision-based navigation) of ground robots using a camera positioned at a certain elevation above the confined area. Vision-based navigation of robots is like human navigation by the help of eyes vision. However, it is not necessary to build eyes into the robot. It is possible to put eyes on a top position and from the top position the robot can see itself and its motion. It means airborne terrestrial robot control [3,8-12]. One of the developments is to take into account delays in the control system.

The described above system may be used for coordination, navigation and control of ground robots (automated transport, automated agricultural machines, municipal and aerodrome vehicles, garden lawnmowers and so on).

The key innovation of the study is the system of navigation and intercoordination. The system includes one or more robots, located on the controlled area; one or more robot tracing devices on the suspended tethered platforms; natural or artificial markings; charger; and central module for robot coordination and orientation detection. The information is transferred to the central module from all the tracing devices. The system central module is equipped with the calculation module located or on the suspended platform, or on the ground, or on the charger or on the robot. The calculation module is possible to calculate coordinates and orientation of robots and to form the control commands, based on the information received from all the above described devices. This visual system can also be used for preventing collision of ground robot with children or animals. Also, this visual system can be used for security purposes.

The technical result is the development of the robot efficient coordination using the devices located on the towers, the aerial apparatuses tracing the robots on the controlled area, supervising their environment including natural and artificial markings. One of the developments is to take into account delays in the control system.

The system (the proof of concept) was defined as

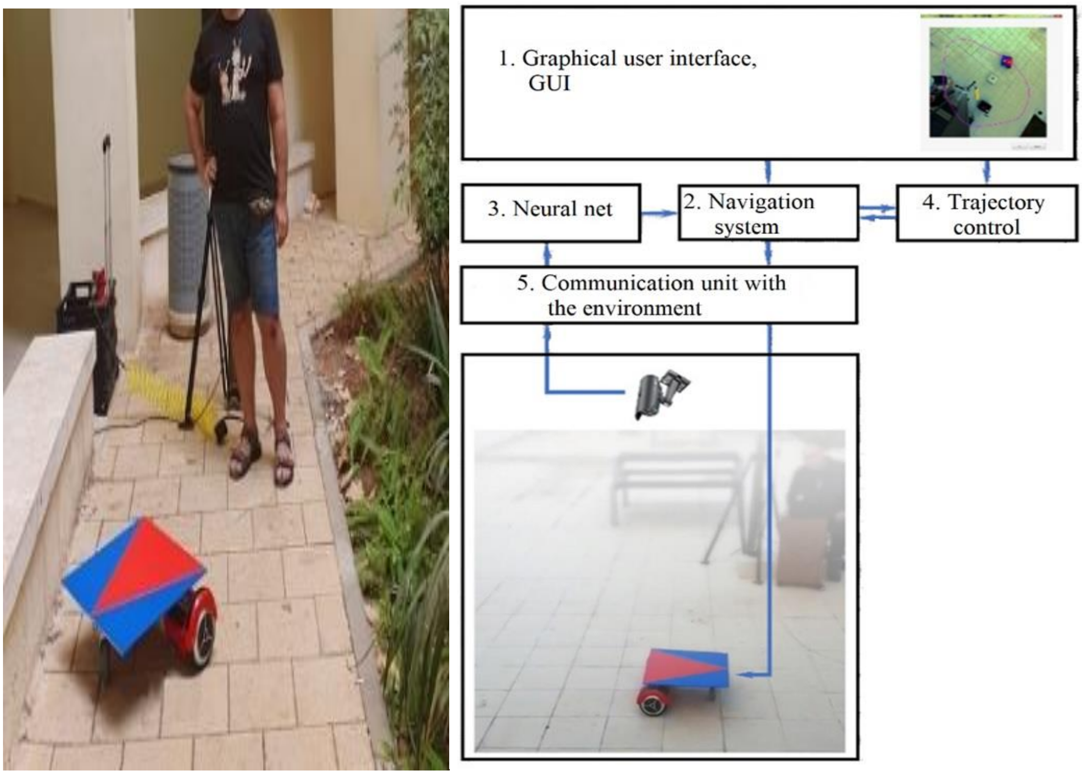

Fig. 1. Airborne terrestrial robot control

following (Fig. 1):

1) The tower (4 meters).
2) The USB camera on the tower connected to computer.
3) The computer which controls the big ground robot by sending Wi-Fi commands, supplied by software for finding its position and orientation using deep learning (artificial neural network) from USB camera images.
4) The ground robot.

It was demonstrated that this robot can pass two main tests: travelling along some predefined path and random walk inside of some predefined boundary. A reader can find these tests here [13,14].

2. CONTROL AND NAVIGATION SYSTEM

Most visual recognition methods use set of correspondent characteristic points [15] to locate the ground robot with respect to an observation camera. In this case, various SIFT functions are usually used to highlight the correspondent characteristic points [15]. These methods require a large database of functions, efficient search methods and a lot of time for calculations.

Using a deep learning convolution neural network can significantly reduce data processing time and increase accuracy.

Despite the massive enthusiasm for neural networks and their active implementation in computer applications, their use to determine the coordinates of objects relative to cameras is not investigated practically.

However, there are a series of interesting works by A. Kendall, M. Grimes and R. Cipolla (University of Cambridge) [16, 17], who used a convolutional neural network to determine camera coordinates, and are pioneers of the use of deep learning methods in navigation. In their works, they determined the coordinates and angles of rotation of the camera using images of Cambridge Landmarks building, made by pedestrian smartphone’s camera.

Unlike the authors of [16,17], who used images of buildings from ground level and using cameras mounted on ground robots, In the current paper it is necessary to find the position of a ground robot relative to a camera in an upper position. The neural network was based on the popular AlexNet network with minor changes in the structure and training mode. AlexNet [18] is the name of a convolutional neural network (CNN), designed by Alex Krizhevsky and published with Ilya Sutskever and Krizhevsky's doctoral advisor Geoffrey Hinton. AlexNet competed in the ImageNet Large Scale Visual Recognition Challenge on September 30, 2012, significantly outperforming other networks. In the original Alex Net, there are only 8 levels (5 convolutional and 3 fully connected). For the current tasks, the last two fully connected layers was removed and replaced with one fully connected layer with 4 outputs (x, y coordinates and trigonometric functions sin() and cos() for the rotation angle α) or 7 outputs (x, y, sin(α) , cos(α), coordinate, angle and total errors) and one regression layer. That is, the pre-trained AlexNet network was used to solve the regression problem: determining two coordinates and course angle of the robot (Fig. 2).

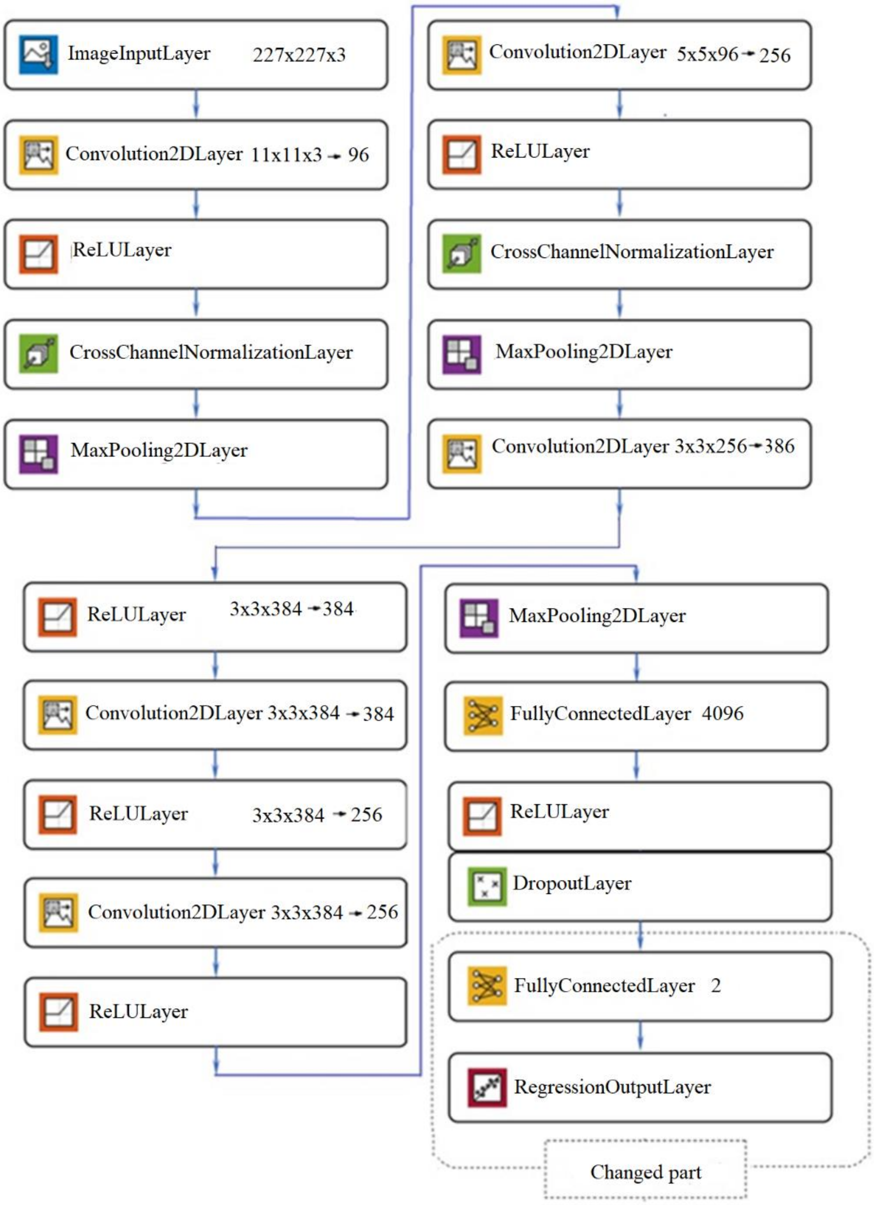

Fig. 2. Transformed AlexNet in Matlab

Two-step network training method was used.

At the first stage, the task of teaching the network to determine the coordinates and course angle of the robot in ideal visibility conditions was solved.

For this, the part of the sample in which the robot is fully open and clearly visible was used. Number of such images in the sample was 9293.

The network containing 4 outputs was built - the two coordinates of the robot in the image, as well as the sine and cosine of its rotation course angle. The direct value of the angle when training the network was not used, since the angle values have a gap at -180: 180, which leads to big errors at angles close to these values.

The sine and cosine Sin_{NN} and Cos_{NN} calculated by the network were used to calculate the angle of rotation of the robot through a weighted average, where the weights of each of the calculated angles is directly proportional to the rate of function change. For example, for Sin_{NN}>0 and Cos_{NN}>0 it can be gotten:

$$\alpha \approx \frac{arcsin(Sin_{NN}) * |Cos_{NN}| + arccos(Cos_{NN}) * |Sin_{NN}|}{|Cos_{NN}| + |Sin_{NN}|} \quad (1)$$

Training was carried out using the solver "adam", training took place in 200 epochs with 500 batch size. This network made it possible to determine the coordinates of the robot with an accuracy of 2.6 pixels (6 cm) in an image of 227 X 227 pixels size and an angle with an accuracy of 4 degrees (see Table 1).

Further, this network was tested on a full sample of 12,000 images, and for each image three values were determined:

- coordinate error

$$err_{coor} = \sqrt{(x_{real} - x_{net})^2 + (y_{real} - y_{net})^2} \quad (2)$$

- angle error

$$err_{angle} = 1000 \cdot \sqrt{(\sin\alpha_{real} - \sin\alpha_{net})^2 + (\cos\alpha_{real} - \cos\alpha_{net})^2} \quad (3)$$

- total error

$$err_{full} = err_{coor} + err_{angle} \quad (4)$$

Finally, the err_{full} is total error value for images, in which the robot is fully visible, is ranged from 0 to 150, for images where the robot is partially visible - from 150 to 1500, and for images where the robot was not visible - from 500 to 4500 (Fig. 3).

At the second stage a network was built, where, in addition to the coordinates and trigonometric functions of the angle, the network was trained to predict err_{full}, err_{coor}, err_{angle}. Thus, by teaching the network to predict the possible error value, the tool that allows us not only to determine the coordinates was gotten, but also to verify the accuracy of their determination. If the robot turned out to be completely or partially covered by environmental objects, it can be seen from the increase of the predicted errors.

This approach allowed us to avoid working with two networks, the first of which would be classifying, and determining the presence of the robot in the image, and the second would be finding its coordinates and angle. The classification problem has been replaced by the regression task of error prediction.

Training was carried out using full sample of 10,000 images and the solver "adam", training took place in 200 epochs with 500 batch size. Thanks to modeling in Unity, the dataset was generated programmatically. At each step, the position and angle of rotation of the robot, as well as the location of the elements of the working platform, were randomly selected.

Lighting simulated sunlight falling from various angles and directions. As a background, 50 different textures were used to achieve independence of determining the coordinates of the robot from background elements (Fig. 4). 12,000 rgb images of size 3036 X 3036 containing robots were generated. Among the images were also shots where the robot was partially or completely covered by environmental objects or the tower of the camera. A file was also prepared with information about the coordinates of the robot and the rotation angle. Then, based on these images, a set of images reduced to 227 X 227 pixels was prepared.

3. AUTOMATIC CONTROL

Let us define the following variables and parameters used in equations of motion for the ground robot (see Fig. 6):

1) For variables describing motion:

x and y– coordinates of ground robot; α - angle of rotation of the robot on the plane; v – translation velocity of the robot; ω - angle velocity of the robot

2) Ground robot parameters:

R – wheel radius; l – distance between wheels

3) Controlling signals:

ω_R and ω_L– angle velocities of rotation of the right and left wheels

It is shown in [19] that forward movement and rotation are described by the system of equations

$$\dot{x} = v\cos\alpha\,;\ \dot{y} = v\sin\alpha\,;\ \dot{\alpha} = \omega; \quad (5)$$

$$v = \frac{R(\omega_R + \omega_L)}{2};\ \omega = \frac{2R(\omega_R - \omega_L)}{l} \quad (6)$$

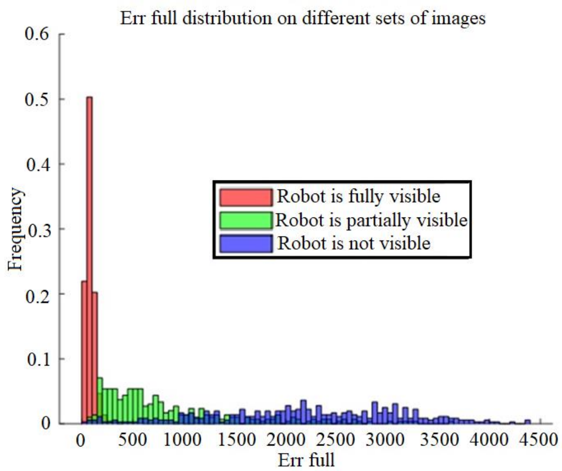

Fig. 3 The total error distribution for different visibilities of the robot

Fig. 4. The different images of the robot from the dataset

Since the system is nonlinear, it's too hard to use those equations to analyze stability. It is necessary to linearize those equations on the premise that the parameters $x(\mathrm{t})$, $y(\mathrm{t})$, $\alpha(\mathrm{t})$, $v(\mathrm{t})$, $\omega(\mathrm{t})$ corresponding with steady flight get small increments $\delta x(\mathrm{t})$, $\delta y(\mathrm{t})$, $\delta\alpha(\mathrm{t})$, $\delta v(\mathrm{t}-\tau)$, $\delta\omega(\mathrm{t}-\tau)$ caused by perturbations action on a path.

Robot trajectory can be estimated by polygonal chain path. This path consists of linear motion along line segments with constant translational velocity and zero angular velocity (Rotation), and rotations in vertices with constant angular velocity and zero translational velocity (Linear motion).

For the case when stationary parameters cannot provide stability of the desirable stationary trajectory themselves, it is necessary to use autopilots (Fig. 5). An autopilot states the controlling parameters $\delta v(\mathrm{t}-\tau)$, $\delta\omega(\mathrm{t}-\tau)$ to be functions of the output-controlled parameters, which are deviations from the desirable stationary trajectory $\delta x(\mathrm{t})$, $\delta y(\mathrm{t})$, $\delta\alpha(\mathrm{t})$. The values of the output parameters can be obtained by autopilot from navigation measurements, for example, from vision-based navigation, inertial navigation, satellite navigation and so on. On the basis these navigation measurements, the autopilot forms controlling signals to decrease undesirable deviation. Unfortunately, there always exists some in getting information about the output-controlled parameters to autopilot for any navigation measurements. So, the problem exists, because of the lack of some necessary information for controlling. In the paper [20], there was demonstrated that even for such conditions with the time delay it is possible to get controlling signal providing a stable path.

Final solution from [20] are following:

For rotation, the stationary solution is

$$\alpha(\mathrm{t})=\omega\mathrm{t}+\varphi;\ v(\mathrm{t})=0;\ \varphi=0 \qquad (7)$$

where control parameters are following:

$$\delta v(\mathrm{t}-\tau) = -2\,a_r\,cos\big(\omega(\mathrm{t}-\tau)\big)\delta x(\mathrm{t}-\tau)$$
$$-2\,a_r\,sin\big(\omega(\mathrm{t}-\tau)\big)\delta y(\mathrm{t}-\tau)$$

$$\text{where } \omega \neq 0, a_r > |\omega|; \qquad (8)$$

$$\delta\omega(\mathrm{t}-\tau) = b_\alpha \delta\alpha(\mathrm{t}-\tau) \text{ where } b_\alpha < 0. \qquad (9)$$

For linear motion, the stationary solution is

$$\alpha(\mathrm{t})=\alpha;\ v(\mathrm{t})=v \qquad (10)$$

where control parameters are following:

$$\delta v(\mathrm{t}-\tau) = -a_l\,cos(\alpha)\delta x(\mathrm{t}-\tau)$$
$$-a_l\,sin(\alpha)\delta y(\mathrm{t}-\tau)$$

$$\text{where } a_l > 0; \qquad (11)$$

$$\delta\omega(\mathrm{t}-\tau) = a\ \sin(\alpha)\ \delta x(t-\tau)$$
$$-a\ \cos(\alpha)\ \delta y(\mathrm{t}-\tau) - 2b\ \delta\alpha(\mathrm{t}-\tau)$$

$$\text{where } av > 0;\ b > \sqrt{av}. \qquad (12)$$

The delay time is following:

$$\tau \le min\left(\frac{1}{2e|b|}, \frac{1}{e|a_l|}, \frac{1}{e|b_\alpha|}, \frac{1}{2e|a_r|}\right) \qquad (13)$$

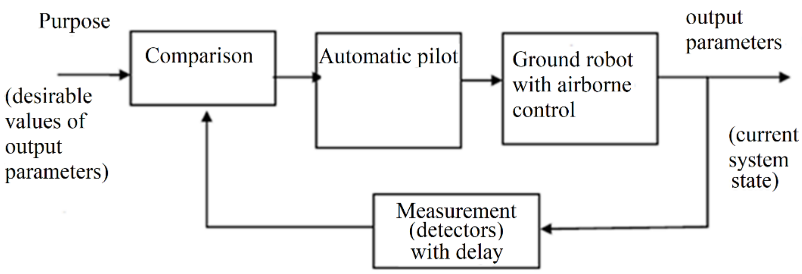

Fig. 5 Automatic control

Fig. 6 Ground robot

4. EXPERIMENTAL RESULTS

This deep learning network is possible to find ground robot angle with an accuracy of 4° and ground robot position with the accuracy of 2.6 pixels (corresponds to 6 cm on the ground) (Table 1) for any 227 X 227 pixels size image.

It is possible to calculate robot recognition error in deep learning network using 3 (from 7 outputs) with position and orientation errors (in 7-outputs network, described above). So, it is possible can conclude that the robot does not exist on an image or is partially occluded if error is very big (more than 150, see Fig. 3).

The robot surface was chosen to be strongly different from background, so recognition quality is almost ideal. Not because of high quality of the network or the diffusion map, but because of choosing the robot surface, which is very different form the background. Indeed, for robust navigation it is necessary to choose such robot surfaces, which are strongly different from the background. The chosen robot surface is optimal for recognition (see Fig. 6).

The software was developed using designed algorithms. It was demonstrated in real experiments that it is possible to carry out two options to drive the ground robot: random movement in user predefined boundaries and user predefined route ([13,14] and Fig. 7). These experiments [13,14] prove that robot indeed was possible to travel inside desirable boundaries in the first case and to carry out stable motion along desirable route in the second case.

Table 1 Error of coordinates and rotation angle for the ground robot for navigation by the help of the artificial neural network, size of the robot is 50cmX50cm, pixel size corresponds to 2,3 cm on the ground

Sets	σ_u	σ_v	$\sigma_{cos(\alpha)}$	$\sigma_{sin(\alpha)}$	σ_α
Validation set	2.6 pix (6 cm)	2.6 pix (6 cm)	0.07	0.07	4°

5. CONCLUSIONS

It was demonstrated that it is possible to carry out two options to drive the ground robot:

a. A random movement in user predefined boundaries
b. A user predefined route

As a result, it was proved that it is possible to maintain stable movement of a ground robot even when time delay exists in transfer information about output control parameters from navigation measurement devices to autopilot.

It is possible to find control parameters for the particular case of visual airborne navigation of ground robot and estimated max possible delay of the system.

It was also demonstrated that the neural network is an effective tool for determining the coordinates of an object: fast, high-precision, insensitive to interference, changes in the environment and lighting.

The quality of determining coordinates allows you to build a high-quality control system for a ground robot.

Obviously, the proposed methods can also be used to solve other problems where it is necessary to determine the coordinates of the observed object.

The future work includes the following:

1) Creation a system for developing possible trajectories

Fig 7. Two experimental tests - random movement in user predefined boundaries and user predefined route

2) Creation a system for inter-coordination of several robots.
3) Recognition of the environment of robots: ground markers, obstacle, moving objects
4) Recognition of robot operation results: how much grass is mowed, which part of the operation area is treated.

However, the developed robot is only prototype. The main future work is creation of working robot for mass-production. The working robot can be:

1) It is a remote-controlled slope mower [4].
2) Robotic Fruit Harvester - the FFRobot [5]
3) Robots for fire protection [6]
4) Autonomous wheelchair for disabled people [7]

6. REFERENCES

[1] Kupervasser O., Sarychev V., Rubinstein A. and Yavich R., Robust positioning of drones for land use monitoring in strong terrain relief using vision-based navigation, International Journal of GEOMATE, Vol.14, Issue 45, 2018, pp. 10-15.

[2] Djaja K., Putera R., Rohman R A. F., Sondang I., Nanditho G. and Suyanti E., The Integration of Geography Information System (GIS) and Global Navigation Satelite System-Real Time Kinematic (GNSS-RTK) for Land use Monitoring, International Journal of GEOMATE, Vol.13, Issue 36, 2017, pp. 31-34.

[3] Ehrenfeld I., Kupervasser O., Kutomanov H.,

Sarychev V. and Yavich R., Algorithms Developed for Two Porotypes of Airborne Vision-Based Control of Ground Robots, International Journal of GEOMATE, 2020, Vol.19, Issue 72, 2020, pp. 217 – 224.

[4] Internet site describing remote-controlled slope mowers, https://www.greenclimberna.com/

[5] Internet site describing Robotic Fruit Harvester: the FFRobot, https://www.insightssuccess.com/ffrobotics-creating-a-new-age-for-fruit-harvesting/

[6] Internet site describing robots for fire protection, https://www.robotpompier.com/en/

[7] Pires G., Honório N., Lopes C., Nunes U., and Almeida A.T., Autonomous wheelchair for disabled people, Conference proceedings, in Proceedings of the IEEE International Symposium on Industrial Electronics, 1997, pp. 797-801.

[8] Ehrenfeld I., Kogan M., Kupervasser O., Sarychev V., Volinsky I., Yavich R. and Zangbi B., Visual navigation for airborne control of ground robots from tethered platform: creation of the first prototype, Conference proceedings, in Proceeding of the IEEE International Conference on New Trends in Engineering and Technology, GRTIET, Tirupathi, Chennai, Tamil Nadu, India, 2018, p. 96, http://grt.edu.in/news/international-conference-on-new-trends-in-engineering-and-technology-icntet-2018/

[9] Kupervasser O.Yu., Kupervasser Yu.I. and Rubinstein A.A., Russian Utility model: Apparatus for Coordinating Automated Devices, Patent № 131276, Russia, Nov 12, 2012.

[10] Kupervasser O.Yu., Kupervasser Yu.I., and Rubinstein A.A., German Utility model: Vorrichtng fur Koordinierung automatisierter Vorrichtungen. Patent Nr. 21 2013 000 225, Germany, Jul 10, 2015.

[11] Kupervasser O. Yu., Franch Utility model application: Coordination System for Ground Movable Automated Devices. Utility model, publication number 3037157, France, Jun 4, 2016, http://bases-brevets.inpi.fr/en/document-en/FR3037157.html?s=1482862654519&p=5&cHash=1f6f99c78cfb2124b5accb92be338388

[12] Kupervasser O.Yu., Russian Invention: Method for coordination of ground moving automated devices with the help of single central control system, Patent № 2691788, Russia, July 05, 2015.

[13] Video-demonstration of navigation of the described in the paper ground-based robot along a trajectory under the control of a neural network, https://www.youtube.com/watch?v=ekxAM9Uytzc

[14] Video-demonstration of navigation of the described in the paper ground-based robot inside a closed loop under the control of a neural network, https://www.youtube.com/watch?v=KYuXCAzTiPE

[15] Lowe D. G., Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, Vol. 60, 2004, pp. 91–110.

[16] Kendall A., Grimes M. and Cipolla R., PoseNet: A convolutional network for real-time 6-dof camera relocalization, Conference proceedings, in Proceedings of the 2015 IEEE International Conference on Computer Vision, Santiago, Chile, 2015.

[17] Kendall A. and Cipolla R., Geometric Loss Functions for Camera Pose Regression with Deep Learning, Conference proceedings, in Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 2017.

[18] Krizhevsky A., Sutskever I. and Hinton G. E., ImageNet classification with deep convolutional neural networks, Communications of the ACM, Vol. 60, Issue 6, 2017, pp. 84–90.

[19] Longoria R.G., Turning Kinematically, presentation of the university lecture, 2015, https://docplayer.net/31417021-Turning-kinematically.html

{20] Domoshnitsky A., Kupervasser O., Kutomanov H. and Yavich R., A method for stabilization of ground robot path controlled by airborne autopilot with time delay, For Chapter in the Book, Functional Differential Equations and Applications, Ed. Domoshnitsky A., Rasin A. and Padhi S., Springer Nature Singapore Pte Ltd, 2020.