DOI: 10.25140/2411-5363-2024-4(38)-282-291
UDC 681.5:004.51

Volodymyr Voytenko[1]***, Maksym Solodchuk***[2]

[1]PhD in Technical Sciences, Docent, associate professor of the Electronics, Automation,
Robotics and Mechatronics Department; researcher of the Department of Automatic Control
Chernihiv Polytechnic National University (Chernihiv, Ukraine); Lund University, (Lund, Sweden)
E-mail: v.voytenko@stu.cn.ua. **ORCID:** http://orcid.org/0000-0003-1490-0600
ResearcherID: F-8698-2014. **Scopus Author ID**: 36167678700

[2]Head of the Department of Military Training
Chernihiv Polytechnic National University (Chernihiv, Ukraine)
E-mail: maxim_solodchuk@stu.cn.ua. **ORCID**: https://orcid.org/0000-0002-1162-6784

SOFTWARE AND HARDWARE COMPLEX FOR EXPERIMENTAL STUDY OF TWO-COORDINATE POSITIONING SYSTEM OF AUXILIARY UAV VIDEO CAMERA

To solve the problem of fatigue of the UAV operator during long-term search and reconnaissance missions, a research hardware and software complex was created, in which the positioning of the auxiliary UAV video camera with a narrow field of view is carried out. Dynamixel Library for MATLAB and Simulink was used to organize the interaction of actuators with the virtual environment, which allows combining positioning and image analysis subsystems in a single virtual environment. The results of an experimental study of the 2DCAM virtual-physical system are given.

Keywords: *unmanned aerial vehicle (UAV); image analysis; computer vision; positioning; MATLAB; Simulink.*
Fig.: 4. Table: 4. References: 28.

Urgency of the research. The development of electronics, information and computer technologies has given a significant impetus to the improvement of long-known unmanned aerial vehicles (UAVs) [1]. This, in turn, made it possible to significantly expand the scope of application of UAVs compared to manned aircraft [2]. However, the traditional tasks of searching for objects, observing the earth's or water surface remain relevant [3]. The use of UAVs to solve such tasks can be more effective compared to the use of a manned device, especially in cases where the safety of the pilot is at risk. Therefore, further research for improving the UAV and its components is important.

Target setting. Remote control of the UAV is mostly carried out based on the image created by the on-board navigation video camera, transmitted using a radio channel and played back on the video monitor of the ground control point as part of the unmanned aircraft complex. During a long-term search mission, the UAV pilot has the additional task of analyzing this image, which requires increased attention. This can cause fatigue and lead to missing the object of interest [4], sometimes with serious consequences.

Actual scientific research and issues analysis. Much attention is paid to the problem of increasing the efficiency of human-operator interaction and technical means during the development of a human-machine interface [5]. In the case of UAVs, the ground control point can be equipped with additional video monitors and other operators can be included in the image analysis [6]. However, such (an extensive) solution to the problem is impossible when the mentioned ground control point must be highly mobile and inconspicuous, and all functions of interaction with the UAV must be performed by one single person. This can be helped by an additional video camera with a narrow field of view [7], the image of which shows a fragment of the frame of the main camera on an enlarged scale. Based on a sufficiently high degree of detailing of the object of interest, the pilot-operator can make a more informed decision about his further actions [8].

The problem is who (or what) should direct this additional camera to the area of interest within the frame of the main (navigation) camera. In [9], a constructive model of the interaction of two cameras is substantiated, as well as a structure that contains a software system for preprocessing and image analysis is proposed. The ultimate purpose of this system is to

---

determine the area of interest and form commands for positioning an additional video camera. There, the key components of one of the electric drives for positioning an additional camera with mechanical links were determined and their simulation was performed.

In order to carry out an experimental research, software and hardware solutions are proposed in [10], which could potentially be used to solve the entire set of tasks, including image processing and control of the electric motors of the positioning of an additional video camera based on the results of the analysis of the image from the navigation video camera, directly on board the UAV and under the conditions of restrictions typical for this case.

In [11], a structural model of the electric drives of the two-coordinate positioning system of the additional video camera of the UAV is presented. On the basis of the formulated specific requirements, typical for work on board the UAV, the choice of brushless direct current (*BLDC*) motors is justified, and a well-known mathematical model has been refined. In particular, the structure of the motor as a link of the automatic control system, transmission functions, as well as the *BLDC* model in continuous time are proposed. It is the basis for the development of an optimal digital regulator. Taking into account that the operation of an ideal optimal digital controller on a UAV is not possible, and the existing hardware and software tools allow deploying a pre-trained neural network on board, the idea of using a quasi-optimal motion controller is proposed. Its training should be organized on the basis of available software tools, originally designed to work as part of an image recognition system.

As a task of the next stage of research, it was also determined that it is necessary to find the transfer function of the traditional optimal controller and compare the performance of two controllers: optimal and quasi-optimal.

Uninvestigated parts of general matters defining. The conducted theoretical studies, as well as simulation modeling, determine the key parameters and indicate promising directions for the practical implementation of the system of two-coordinate positioning of the additional video camera of the UAV. However, these results must have a sufficiently high degree of verification even before schematic diagrams are developed, electronic components are purchased, printed circuit boards are assembled and manufactured. Otherwise, large material and time costs may not be justified. For example, such a problem as choosing an adequate microcontroller, its performance and peripheral devices, and even expensive tools for it, can significantly worsen the development prospects, if this problem is not solved in the best way. Here we also depend significantly on other components, such as the object detector [10] and others. Therefore, it is very important to do one more intermediate step before making the final system, which is usually called the *physical modeling stage*.

It is most convenient to conduct full-scale experiments in the same software environment in which the simulation was carried out. The *MATLAB/Simulink* environment [12] provides wide opportunities not only for simulation modeling of multi-domain systems, but also for connecting to various executive and measuring devices. Tools such as *Image Processing and Computer Vision* [13] will allow in further studies to extend physical modeling also to subsystems related to image processing. Finally, this environment also supports UAV navigation, artificial intelligence and machine learning tools, which makes it possible to solve a much wider range of tasks related to the problems formulated above.

[14] provides a description of a high-level software library that allows you to control *DYNAMIXEL* actuators directly from the *MATLAB/Simulink* environment [15]. This library contains *MATLAB* functions and *Simulink* blocks of general use, which should be supplemented with such capabilities that will allow solving tasks that rely on the subsystem for positioning the additional camera of the UAV.

The research objective. The purpose of the article is to create a research software-hardware complex that, from the *MATLAB/Simulink* environment, allows you to control the positioning of the additional video camera of the UAV by pitch and yaw angles. This allows combining two subsystems – positioning and image analysis – in a single virtual environment to solve the problem of increasing the efficiency of the UAV pilot-operator during long-term search and reconnaissance missions.

In order to achieve the goal, it is necessary to solve such tasks as the analysis of actuators and their control software; refinement and development of blocks that allow you to form the necessary positioning tasks and visualize the current parameters of the two-coordinate positioning system of the auxiliary UAV camera; experimental testing of the complex's performance.

Presentation of the main material

Hardware and software of the complex. Despite the prospects of BLDC [11], this work uses *DYNAMIXEL* actuator systems based on DC motors [16], as it allows you to quickly assess the possibility of solving the problems formulated above. We justify this choice as follows:

1) In these drives, at the mechanical, electronic and algorithmic level, the motor, reduction gear, controller, driver, sensors, as well as network functions are integrated, which allow to significantly reduce the time required for the development of means of communication with a personal computer.

2) A wide range of actuators allows you to choose a device according to the required parameters (supply voltage, torque on the shaft, resolution, etc.). At the time of writing, there are at least 72 actuators available [17] that use both DC and BLDC motors.

3) The software supported by the built-in controller includes *DYNAMIXEL* (*Wizard*; *SDK*; *Workbench*), *Arduino IDE*, *RoboPlus* (*R*+), etc., which allows (including) to implement the daisy-chaining connection of several actuators [18].

4) Availability of libraries [19], which can immediately be used as a basis for further research within the *MATLAB/Simulink* environment.

5) From the point of view of control, both BLDC and DC motor are very close, which makes it possible to further use the currently created prototypes of motion controllers in other actuators.

The simplest actuators of the first generation were taken as a basis:

– *DYNAMIXEL AX*-12*A* [20] for positioning the video camera at the yaw angle (along the horizontal line);

– *DYNAMIXEL AX*-18*A* [21] for positioning the video camera by the pitch angle (along the vertical line).

The advantage of these models of actuators at the beginning of development was small dimensions and weight, low power supply voltage and cost.

The appearance of the hardware part of the experimental complex is shown in *Fig*. 1.

The research used a personal computer with an *Intel*^{R} *Core*^{TM} i7-4770 CPU @ 3.40GHz, 16.0 GB of RAM, on which the *Windows* 10 *Pro* operating system, *MATLAB*$^{®}$ R2024a was installed, and the *Linux Ubuntu* 20.04.4 LTS (*Focal Fossa*) operating system was deployed using *VirtualBox*.

To work with video cameras and object detectors, a platform for developing embedded systems using machine learning *phyBOARD*$^{®}$*-Pollux AI kit* [22] was used. Two video cameras are connected to the single board computer *phyBOARD-Pollux* [23]: *VM*-016-*COL-M-M*12 [24] and *VM*-017-*COL-M-H* [25]. The latter was chosen in the version with a C/CS mount bracket to ensure the possibility of connecting special lenses.

DYNAMIXEL actuators are connected to a personal computer via a *USB* interface using a small *U*2*D*2 converter [26].

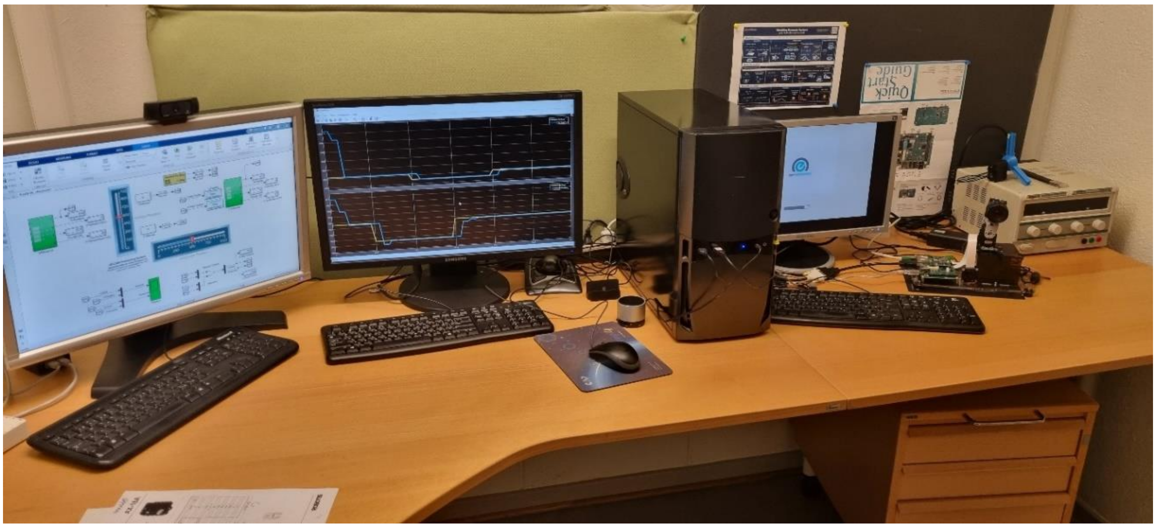

Fig. 1. Software and hardware complex

Source: developed by the authors.

Main functions of *Dynamixel Library for MATLAB and Simulink* and their capabilities

Requirements for the computer platform and a detailed description of the *Dynamixel Library for MATLAB and Simulink* installation are described in [14]. This library, in turn, is based on the dynamic library in the *C* language, which is included in the *DYNAMIXEL SDK software development kit* [19].

The library [14] has a total of 38 *MATLAB* functions. Of them, 15 functions are intended mainly for reading information (*Table* 1), 12 – for writing (*Table* 2).

Table 1 – MATLAB functions of the library [14] *for reading information*

Function	Comments
bulkReadAddParamDxl	Adds parameters for bulk read function of Dynamixel motors
bulkReadDxl	Returns the acquired data from bulk read function for the connected Dynamixel motors
bulkReadTxRxPacket	Executes bulk read for Dynamixel motors
fastSyncReadDxl	Executes fast sync read for Dynamixel motors and returns the selected read parameter
fastSyncReadDxlSim	Executes fast sync read for Dynamixel Motors in Simulink
initBulkReadDxl	Initializes the bulk read function for Dynamixel motors and returns the group numbers
initSyncReadDxl	Initializes the sync read function for Dynamixel motors and returns the group numbers
initSyncReadIndirectDxl	Initializes the sync read function using indirect addresses of Dynamixel motors and returns the group numbers
readDxl	Executes read for Dynamixel motors and returns the acquired data
readDxlSim	Executes read for Dynamixel motors and returns the acquired data
setIndirectReadDxl	Sets indirect addresses for enabling fast sync read for multiple Dynamixel motors
syncReadAddParamDxl	Add parameter for bulk read function of Dynamixel motors
syncReadClearParamDxl	Clears parameters for bulk read function of Dynamixel motors
syncReadDxl	Executes sync read for Dynamixel motors and returns the acquired data
syncReadDxlSim	Executes sync read for Dynamixel Motors in Simulink

Source: developed by the authors.

"Fast" functions allow exchanging information with remote actuators in one operation at the same time, using indirect addressing.

Table 2 – MATLAB functions of the library [14] *for recording information*

Function	**Comments**
bulkWriteAddParamDxl	Adds parameters for bulk write function of Dynamixel motors
bulkWriteDxl	Executes bulk write for Dynamixel motors
fastSyncWriteDxl	Executes fast sync write for Dynamixel motors
fastSyncWriteDxlSim	Executes fast sync write for Dynamixel Motors in Simulink
initBulkWriteDxl	Initializes the bulk write function for Dynamixel motors and returns the group write number
initSyncWriteDxl	Initializes the sync write function using indirect addresses of Dynamixel motors and returns the group numbers
initSyncWriteIndirectDxl	Initializes the sync write function using indirect addresses of Dynamixel motors and returns the group numbers
setIndirectWriteDxl	Sets indirect addresses for enabling fast sync write for multiple Dynamixel motors
syncWriteDxl	Executes sync write for Dynamixel motors and returns the acquired data
syncWriteDxlSim	Executes sync write for Dynamixel motors in Simulink
writeDxl	Executes write for Dynamixel motors
writeDxlSim	Executes write for Dynamixel motors in Simulink

Source: developed by the authors.

Other available *MATLAB* functions can be classified according to the following features:

– initialization functions of both the library itself and the initialization of reading or writing from/to the actuators (7 pcs.);

– functions of synchronous initialization, reading or writing from/to actuators (14 pcs.);

– batch manipulations with actuators (7 pcs.);

– functions of fast synchronous reading or writing from/to actuators (4 pcs.).

It should be noted that this conditional classification helps to quickly navigate the available tools and start using them. Since some of the functions combine several operations mentioned above, that is, they belong to two or more classes at the same time, the total number of functions mentioned here does not coincide with their full number.

Ten unclassified *MATLAB* functions (*Table* 3) allow you to control the communication port, search for and configure found actuators, and check their status and communication errors. Of course, they perform some *I/O* or read or write operations, as do any peripheral access functions.

Table 3 – Unclassified MATLAB library [14] *functions*

Function	**Comments**
openPortDxl	Opens port for communication with Dynamixel motors
setBaudDxl	Sets the baudrate of Dynamixel motor communication
closePortDxl	Closes communication port with connected Dynamixel motors
findserial	Finds and returns connected serial (COM) ports
findDxl	Finds connected Dynamixels, returns their IDs and models
torqueDxl	Enables or disables the torque to Dynamixel motors
driveModeDxl	Changes the drive mode of connected Dynamixel motors
operatingModeDxl	Changes the operating mode of Dynamixel motors
lookupDxl	Searches the control table RAM data of Dynamixel motors
checkErrorDxl	Checks for communication errors and returns the communication status of connected Dynamixel motors

Source: developed by the authors.

MATLAB functions that implement the exchange between actuators directly from *Simulink* should also be highlighted (Table 4).

The library [14] also contains three classes of basic *Simulink* blocks:

– writing, synchronous writing and fast synchronous writing blocks to the *DYNAMIXEL* actuator;

– blocks of reading, synchronous reading and fast synchronous reading of parameters from the *DYNAMIXEL* actuator;

– models of some *DYNAMIXEL* actuators.

Table 4 – MATLAB functions of the library [14] *for communication with Simulink*

Function	Comments
fastSyncReadDxlSim	Executes fast sync read for Dynamixel Motors in Simulink
fastSyncWriteDxlSim	Executes fast sync write for Dynamixel Motors in Simulink
syncReadDxlSim	Executes sync read for Dynamixel Motors in Simulink
syncWriteDxlSim	Executes sync write for Dynamixel motors in Simulink
writeDxlSim	Executes write for Dynamixel motors in Simulink

Source: developed by the authors.

Virtual and physical 2DCAM subsystem

Using the *Dynamixel Wizard* 2.0 [18], the serial port of the computer was identified and the same baud rate was set for both actuators at 1,000,000 *baud*. The actuators used in this study support the exchange protocol version 1.

On the basis of examples [14], a virtual model of the two-coordinate positioning subsystem of the additional video camera of the UAV was created (*Fig*. 2). The model can be used either in manual positioning task mode, or based on previously developed tasks.

The choice of mode is determined using a virtual toggle switch. In the "*Manual Control*" position, the additional video camera of the UAV can be directed in the range of ±150° along the vertical or horizontal, using the appropriate slider controls (*Vertical Position* or *Horizontal Position*). In the "*Scenario*" position, setpoints are used from the numerical file *SetTask.mat*, which is pre-created and adjusted using the *Signal Editor Simulink* block.

To visualize the parameters of both actuators, digital meters of position, positioning speed, as well as moments on the motor shaft and temperature of electric drives are used. For this purpose, each of the *V-Measure* and *H-Measure Simulink* subsystems contains the aforementioned basic *Simulink* blocks for reading certain data from the actuators. Each of these blocks is identical and differs from the others by the parameters of the *C*-function that is called: *Present Position*, *Present Speed*, *Present Load* or *Present Temperature*. Usually, in these *Simulink* blocks, according to the actuators used, the parameters of the *AX*-18*A* actuator are used for vertical positioning, and *AX*-12*A* for horizontal positioning, which are automatically determined at the model initialization stage using the *findDxl* search function (*Table* 3).

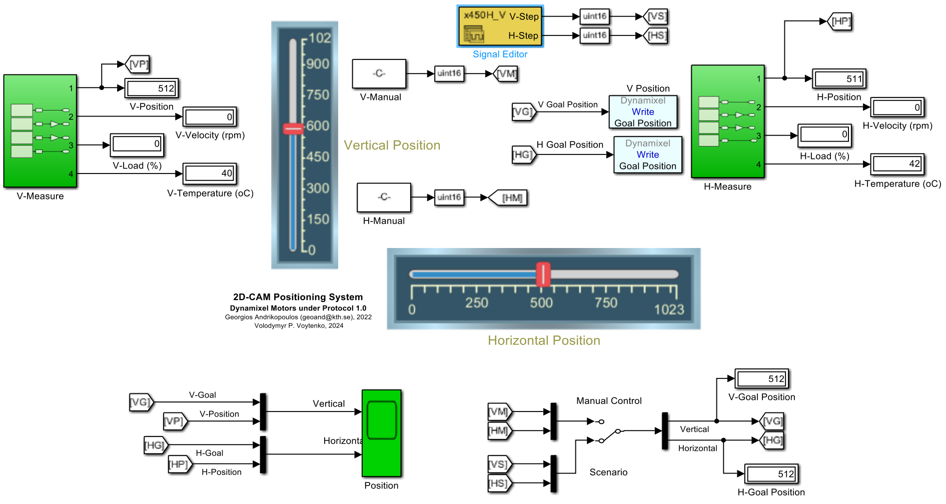

Fig. 2. Virtual part of the 2DCAM subsystem

Source: developed by the authors.

The most important parameters of the system's functioning are the vertical and horizontal positioning setpoint signals, as well as the real values of the position of the UAV's additional video camera obtained through feedback. They are presented visually on the "*Position*" virtual oscilloscope. Blocks of multiplexers and demultiplexers allow all necessary signals to be displayed simultaneously on one screen in two windows.

Of course, the use of a virtual oscilloscope, other measuring devices, as well as measurement devices, blocks (up to individual electronic components) allow to reduce time and reduce the cost of development at the stage of prototyping a complex technical system.

On *Fig*. 3 shows screenshots of oscillograms of transient processes (*N* – ticks from position resolver) in the 2*DCAM* subsystem, which are obtained from the virtual oscilloscope "*Position*".

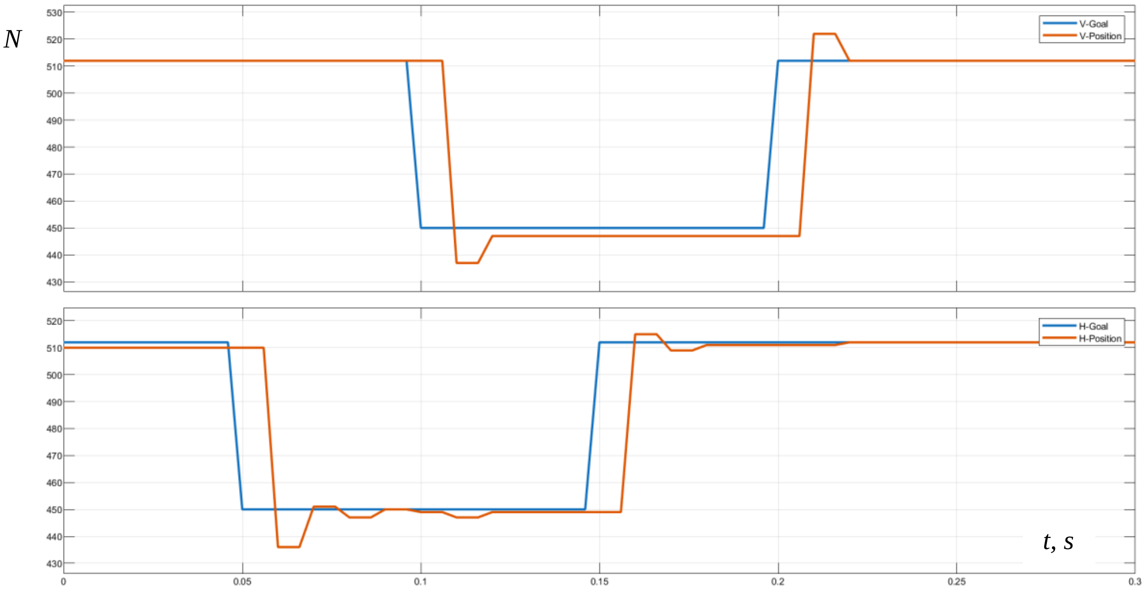

Fig. 3. Transient processes in 2DCAM subsystem

Source: developed by the authors.

As can be seen from *Fig*. 3, the actuators successfully work out the setpoint signals, and the further task is the development and implementation of virtual controllers in the 2*DCAM* subsystem, which will allow to optimize transient processes.

On *Fig*. 4 presents a photo of the hardware part of the 2*DCAM* subsystem. In addition to an electromechanical unit with two video cameras, a *U*2*D*2 interface converter, two actuators for an additional video camera together with a lens with a reduced viewing angle, the photo shows a single-board computer *phyBOARD-Pollux*, as well as other components of a specialized computer system: an auxiliary video monitor and keyboard.

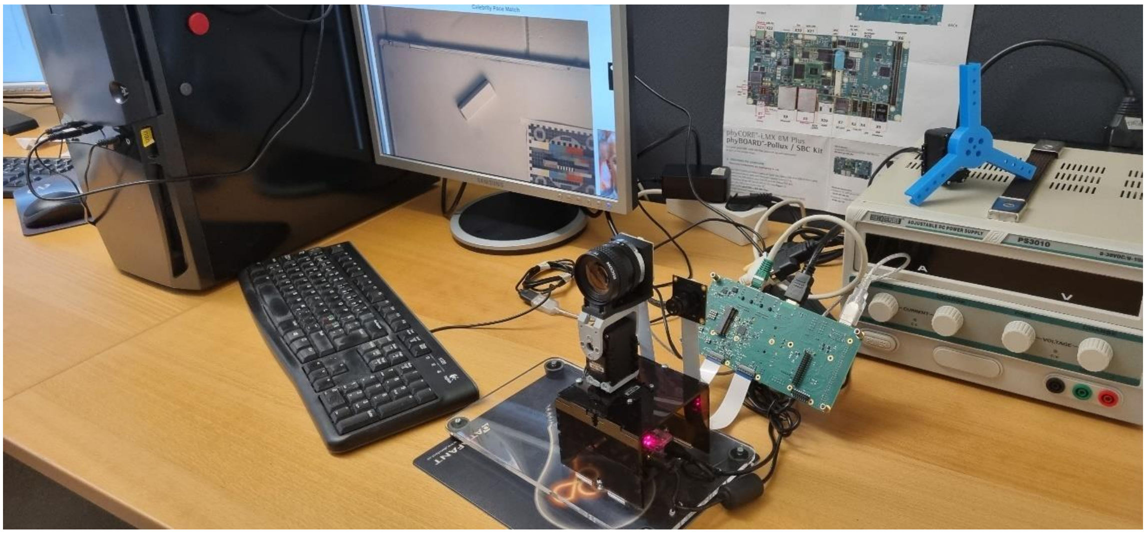

Fig. 4. Hardware part of the 2DCAM subsystem

Source: developed by the authors.

The auxiliary video monitor shows a test image received from the main (navigation) video camera of the UAV, which has undergone digital pre-processing using a processor installed on a single-board computer.

Conclusions. For the first time, a software-hardware complex was developed, which in a single *MATLAB/Simulink* virtual environment allows you to study individual components of a special system for use on board a UAV, as well as their interaction. This system is designed to increase the efficiency of the UAV pilot-operator during long-term search and reconnaissance missions. Components:

– the electromechanical subsystem of the two-coordinate positioning of the additional video camera of the UAV 2*DCAM*;

– software subsystem of image analysis and detection of the area of interest;

– other parts.

During the research, the main efforts were concentrated on the creation of the 2*DCAM* subsystem. For this purpose, an analysis of possibilities and selection of actuators and software control methods was carried out. The relevant software library has been installed and the available *MATLAB* functions have been classified, which facilitates their use. The *Simulink*-blocks were refined and re-created in accordance with the required range of setpoints of the developed system. These blocks made it possible to confirm the correctness of the choice of hardware and software tools used to create the complex.

Experimental verification of the interaction of the virtual component with the available hardware demonstrated the possibility of two-coordinate positioning of the auxiliary UAV camera at pitch and yaw angles of ±150°, which meets the requirements. The obtained results allow further efforts to be concentrated on the optimization of motion controllers in terms of accuracy and speed; on image analysis and area of interest detection using solutions and tools such as *MATLAB and Simulink for Control Systems*, *Image Processing and Computer Vision*, *Embedded Systems*, *Machine Learning*, *Mechatronics* and others.

Careful links to primary sources allow you to use the article, including as a reference, which will quickly help a specialist in the field of robotics and mechatronics to solve a complex of tasks that arise during the development of prototypes of various electromechanical devices. Further research can be done on numerous examples in *MATLAB* in the field of artificial intelligence, including in the *Python* programming language, as well as other, more accurate actuators (for example, *XM*430-*W*210*T*, etc.). This allows us to use the advantages provided by the concept of high-level programming of cyber-physical systems [27].

The availability of a large volume of educational literature and the multi-domain nature of *MATLAB/Simulink* allows the results presented in the article to be used in the educational process as well.

Acknowledgements. The authors are highly grateful to the Crafoord Foundation and Scholars at Risk for providing a grant. We also express our gratitude to Andrii Hrechuk for the manufactured mechanical attachment of the additional video camera.

References

1. Beard, R. W., & McLain, T. W. (2012). *Small unmanned aircraft: Theory and practice*. Princeton University Press.

2. *Different types of drones and uses (2024 full guide)*. (2024, January 3). JOUAV. https://www.jouav.com/blog/drone-types.html.

3. *Drone impact society*. (2019, June 25). Dronescend. https://dronescend.com/blogs/news/impact-dji-drones-on-our-society.

4. Lee, J. D., Wickens, C. D., Liu, Y. & Boyle, L. N. (2017). *Designing for People: An Introduction to Human Factors Engineering*. Createspace Independent Publishing Platform.

5. Kun, A. L. (2018). *Human-machine interaction for vehicles: Review and outlook*.

6. *Drone operators & pilots*. (n.d.). EASA. https://www.easa.europa.eu/en/light/topics/drone-operators-pilots.

7. *WESCAM MX™-10, air surveillance and reconnaissance*. (n.d.). L3Harris® Fast. Forward. https://www.l3harris.com/all-capabilities/wescam-mx-10-air-surveillance-and-reconnaissance

8. Voytenko, V., & Solodchuk, M. (2022). Increasing the speed of analysis of images obtained from unmanned aerial vehicle. *Technical Sciences and Technologies*, (2(28)), 127-137. https://doi.org/10.25140/2411-5363-2022-2(28)-127-137.

9. Voytenko, V., Denisov, Y., Yershov, R., Solodchuk, M. (2023). A Conceptual Model for Increasing the Speed of Decision-Making Based on Images Obtained from UAVs. In: Shkarlet, S., et al. *Mathematical Modeling and Simulation of Systems. MODS 2022. Lecture Notes in Networks and Systems, vol 667*. Springer, Cham. https://doi.org/10.1007/978-3-031-30251-0_23.

10. Voytenko, V., Olofsson, B., Solodchuk, M., & Denisov, Y. (2023). Components of a system for automatic detectionof a zone of interest in images obtained from a UAV. *Technical Sciences and Technologies*, (2(32)), 300-312. https://doi.org/10.25140/2411-5363-2023-2(32)-300-312.

11. Voytenko, V., Solodchuk, M., Denisov, Y., Pisarevskiy, A. (2024). Controllers for Two-Coordinate Positioning of the UAV Auxiliary Video Camera. In: Kazymyr, V., et al. *Mathematical Modeling and Simulation of Systems. MODS 2023. Lecture Notes in Networks and Systems, vol 1091*. Springer, Cham. https://doi.org/10.1007/978-3-031-67348-1_29.

12. *Products and Services*. (n.d.). MathWorks. https://mathworks.com/products.html?s_tid=nav_products.

13. *Image Processing and Computer Vision.* (n.d.). MathWorks. https://mathworks.com/solutions/image-video-processing.html.

14. Andrikopoulos, G. (2024). Dynamixel Library for MATLAB and Simulink. https://mathworks.com/matlabcentral/fileexchange/125545-dynamixel-library-for-matlab-and-simulink.

15. All-in-One Actuator DYNAMIXEL. (n.d.). https://en.robotis.com/shop_en/category.php?ca_id=20.

16. What is DYNAMIXEL? (n.d.). https://www.dynamixel.com/whatisdxl.php.

17. DYNAMIXEL Selection Guide. (n.d.). https://en.robotis.com/service/selection_guide.php.

18. Software. (n.d.). *Robotis E-manual*. https://emanual.robotis.com/docs/en/software/.

19. DYNAMIXEL SDK. (n.d.). *Robotis E-manual*. https://emanual.robotis.com/docs/en/software/dynamixel/dynamixel_sdk/api_reference/matlab/matlab_porthandler/#matlab.

20. AX-12A. (n.d.). *Robotis E-manual*. https://emanual.robotis.com/docs/en/dxl/ax/ax-12a/.

21. AX-18A. (n.d.). *Robotis E-manual*. https://emanual.robotis.com/docs/en/dxl/ax/ax-18a/.

22. *I.MX 8M plus EVK for AI and embedded vision | PHYTEC*. (n.d.). PHYTEC Messtechnik GmbH. https://www.phytec.eu/en/produkte/development-kits/phyboard-pollux-ki-kit.

23. *I.MX 8M plus Cortex-A53 / M7 single board computer | PHYTEC*. (n.d.). PHYTEC Messtechnik GmbH. https://www.phytec.eu/en/produkte/single-board-computer/phyboard-pollux/.

24. *MIPI CSI-2 camera module for embedded vision | PHYTEC*. (n.d.). PHYTEC Messtechnik GmbH. https://www.phytec.eu/en/produkte/embedded-imaging/kameramodule/vm-016-phycam-m/.

25. *5 Mp MIPI CSI-2 camera module for embedded vision | Phytec*. (n.d.). PHYTEC Messtechnik GmbH. https://www.phytec.eu/en/produkte/embedded-imaging/kameramodule/vm-017-phycam-m/.

26. U2D2. (n.d.). *Introduction*. ROBOTIS e-Manual. https://emanual.robotis.com/docs/en/parts/interface/u2d2/.

27. Voytenko, V. & Yershov, R. (2021). Ultra-high-level programming of the system of electric drives of quadcopters and autonomous robots. *Technical sciences and technologies*, (4(26)), 129-139. https://doi.org/10.25140/2411-5363-2021-4(26)-129-139.

Отримано 20.12.2024

УДК 681.5:004.51

***Володимир Павлович Войтенко*[1]*, Максим Олександрович Солодчук*[2]**

[1]кандидат технічних наук, доцент, доцент кафедри електроніки, робототехніки, автоматики та мехатроніки;
дослідник кафедри автоматичного керування
Національний університет «Чернігівська політехніка» (Чернігів, Україна); Лундський університет (Лунд, Швеція)
E-mail: v.voytenko@stu.cn.ua. **ORCID:** http://orcid.org/0000-0003-1490-0600
ResearcherID: F-8698-2014 **Scopus Author ID:** 36167678700

[2]начальник кафедри військової підготовки
Національний університет «Чернігівська політехніка» (Чернігів, Україна)
E-mail: maxim_solodchuk@stu.cn.ua. **ORCID**: https://orcid.org/0000-0002-1162-6784

ПРОГРАМНО-АПАРАТНИЙ КОМПЛЕКС ДЛЯ ЕКСПЕРИМЕНТАЛЬНИХ ДОСЛІДЖЕНЬ СИСТЕМИ ДВОКООРДИНАТНОГО ПОЗИЦІОНУВАННЯ ДОДАТКОВОЇ ВІДЕОКАМЕРИ БПЛА

БПЛА протягом тривалого часу використовуються для вирішення завдань, коло яких постійно розширюється завдяки досягненням у галузі електронних систем, комп'ютерного зору, штучного інтелекту та ін. Тому подальші дослідження з метою удосконалення БПЛА та його складових мають важливе значення. Однією із суттєвих проблем, пов'язаних із використанням БПЛА, є напруга пілота-оператора під час виконання довготривалих пошукових і розвідувальних місій. Можливе вирішення передбачає встановлення додаткової до навігаційної відеокамери з вузьким кутом зору. Збільшене зображення області інтересу, яке створюється додатковою відеокамерою, допомагає оператору, проте потребує відповідного позиціонування додаткової відеокамери всередині поля зору основної відеокамери. Це позиціонування може здійснюватися за командами, які формує детектор об'єктів.

Мета статті – створення дослідницького програмно-апаратного комплексу, який із середовища MATLAB/Simulink дозволяє керувати позиціонуванням додаткової відеокамери БПЛА за кутами тангажу та рискання. Це дозволяє об'єднати дві підсистеми – позиціонування та аналізу зображення – в єдиному віртуальному середовищі для вирішення проблеми підвищення ефективності роботи пілота-оператора БПЛА під час виконання тривалих пошукових і розвідувальних місій.

У роботі обґрунтовано вибір актуаторних систем на базі двигунів постійного струму DYNAMIXEL, які підключено до персонального комп'ютера через інтерфейс USB. Для реалізації концепції надвисокорівневого програмування актуаторів та використання інструментів середовища MATLAB/Simulink для подальших досліджень використано бібліотеку Dynamixel Library for MATLAB and Simulink, функції якої класифіковані за ознаками, що полегшує їхнє використання. Розроблено віртуально-фізичну підсистему двокоординатного позиціонування додаткової відеокамери БПЛА 2DCAM, результати експериментальних досліджень якої підтверджують справедливість вибору апаратних та програмних засобів, використаних для створення комплексу.

Стаття може бути корисною для розробників кібер-фізичних систем, а також використана в освітньому процесі.

***Ключові слова:** безпілотний літальний апарат (БПЛА); аналіз зображень; позиціонування; MATLAB; Simulink.*

Рис.: 4. Табл.: 4. Бібл.: 28.

Voytenko, V., Solodchuk, M. (2024). Software and hardware complex for experimental study of two-coordinate positioning system of auxiliary UAV video camera. *Technical sciences and technologies,* (4(38)), 282-291.