

СТВОРЕННЯ КОМП'ЮТЕРНИХ ЗАСОБІВ КЕРУВАННЯ АВТОНОМНИМ ЛІТАЛЬНИМ АПАРАТОМ З ЗАСТОСУВАННЯМ ДИСТАНЦІЙНИХ ХМАРНИХ ОБЧИСЛЕНЬ

В. І. Чигінь

Національний університет "Львівська політехніка",
кафедра електронних обчислювальних машин
vchygin@gmail.com

© Чигінь В.І., 2021

Досліджена можливість створення комп'ютерної моделі керування безпілотним літальним апаратом з застосуванням дистанційних хмарних обчислень за заздалегідь заданими сценаріями з робочого стола користувача. Для цього створена експериментальна установка, яка включає квадрокоптер, персональний комп'ютер з операційною системою Windows, бортовий комп'ютер Raspberry-3 з операційною системою Linux, відеокамеру Pi Camera V2, автопілот Pixhawk. Для моделювання процесів керування і передавання відеозображень на комп'ютері Raspberry-3 мовою Python записано власні програми керування і фотопереслідування. За отриманими результатами запропоновано модель керування безпілотним літальним апаратом з робочого стола персонального комп'ютера користувача через бортовий комп'ютер без використання стандартного пульта керування та оператора.

Ключові слова: комп'ютерна модель керування, квадрокоптер, автопілот, бортовий комп'ютер, робочий стіл користувача

Вступ

Робота є продовженням власних досліджень [1-3] у напрямку створення комплексної комп'ютеризованої системи пасивного виявлення (звуковим, фото- чи іншими засобами) несанкціонованих об'єктів (безпілотних літальних апаратів – БПЛА, зародків пожеж і т.п.), вимірювання координат, напрямку і швидкості польоту БПЛА за допомогою стаціонарної фотоапаратури, а також їх перехоплювання власним БПЛА з бортовим комп'ютером і відеокамерою. Досі не розглядалося питання створення комп'ютерної моделі керування БПЛА за наперед заданими сценаріями з робочого стола персонального комп'ютера користувача через бортовий комп'ютер без використання стандартного пульта керування та оператора.

З практики використання різного типу комерційних БПЛА [4,5] відомо, що сценаріїв для їх польоту може бути багато. Сюди відносяться: отримання відеорядів при облітанні архітектурних об'єктів і туристичних пейзажних місць, охорона державних і військових об'єктів, фотосканування насаджень з метою аналізу і прогнозу врожайності, вимірювання площ сільськогосподарських угідь, охорона фермерських господарств і т.п. При цьому серед споживачів є гостра потреба в організації саме автоматичного (без втручання підготовленого оператора) вильоту БПЛА, польоту за наперед заданим сценарієм, повернення на місце постійної дислокації, а також отримання наживо відеоряду на підручні засоби типу ноутбука, планшета чи смартфона вдалині від місць польотів БПЛА (наприклад, в офісі при польотах над фермерськими полями і тп).

Огляд літературних джерел

У попередніх роботах [2,3] вперше проводились експериментальні дослідження і комп'ютерні моделювання автоматизованих польотів БПЛА коптерного типу при попередньому заданні точок з відомими GPS-координатами, а також при захопленні об'єкта бортовою камерою. Вперше виконані комп'ютерні моделювання обертання квадрокоптера з імітацією фотопереслідування [6].

В українських інтернет-виданнях не виявлено робіт, зв'язаних з використанням бортового комп'ютера для керування польотами за різними сценаріями. Сучасні міні-комп'ютери, які можна використати у ролі бортових для БПЛА, описані на сайтах: Raspberry Pi3 [7], Odroid C2 [8], LattePanda [9]. Комп'ютери Raspberry і Odroid використовують ОС Linux, LattePanda – Windows-10. Відомо, що стандартні програми Mission Planner [10], Litchi [11], Q-GroundControl [12], за допомогою яких можна керувати польотами БПЛА фірми DJI типу Phantom, Mavic і т.п., є обмеженими у плані необхідності тривалого встановлення сценаріїв польотів, ручного задання координат точок траєкторій польотів. Вони призначені, зокрема, для перевірки сенсорів БПЛА перед пуском, заведення моторів і виконання польоту у певному режимі (*Gaided*, *Loiter* тощо). Основний їх недолік є закритість, неможливість створення певних сценаріїв, зокрема, з фотопереслідування і виконання дії знешкодження. Крім того, результати відеознімань розміщуються на карті пам'яті автопілота, а також надаються для спостереження оператору, який знаходиться поряд з точкою вильоту з пультом керування. В окремих роботах розглядаються питання фотозахоплення з БПЛА. Так, нерухома червона куля розпізнається і знешкоджується за допомогою квадрокоптера [13]. За даними [14] коптер DJI Mavic2 Pro може відслідковувати значні за розмірами об'єкти (людину, автомобіль) і летіти на певній відстані від об'єкта (режим *Follow me*). Проте, програми є закритими для модернізації. У роботах [15-17] поміщені стандартні бібліотеки OpenCV, Picamera, DroneKit для розпізнавання об'єкта та опрацювання зображення, зв'язку бортового комп'ютера з бортовою камерою, управління БПЛА. Не виявлено досліджень в області комп'ютерних моделювань польотів з використанням спрощеного задання сценаріїв польоту, безпосереднього перекидання їх параметрів від робочого стола користувача, минаючи пульт керування підготовленого оператора, на борт БПЛА.

Постановка задачі

В роботі поставлена мета створити комп'ютерну модель автоматичного керування безпілотним літальним апаратом за наперед заданими сценаріями з робочого стола стаціонарного комп'ютера користувача. При цьому провести експериментальні дослідження і комп'ютерні моделювання, підібрати адекватне програмне забезпечення для передавання команд керування на бортовий комп'ютер Raspberry з Pi-камерою, минаючи стандартний пульт оператора.

Результати досліджень

Схема комп'ютерної моделі керування БПЛА. На рисунку 1 зображено запропоновану схему комп'ютерної моделі керування БПЛА. Тут 1 – користувач (не обізнаний з деталями складання, відлагодження та експлуатації безпілотних літальних апаратів), 2 – робочий стіл користувача (персональний комп'ютер, ноутбук), 3 – команди на бортовий комп'ютер, 4 – оператор польотів БПЛА (у цій схемі – зайвий), 5 – безпілотний літальний апарат, 6 – бортовий комп'ютер, 7 – автопілот, 8 – бортова відеокамера, 9 – бортовий виконавчий механізм, 10 – траєкторії польоту, піднімання і опускання БПЛА, 11 – відео знімання об'єктів і виконання впливу на об'єкти.

На рисунку 2 зображена схема, використана при дослідженнях за сценарієм фото переслідування. Квадрокоптер підвішено на вертикальних розтяжках для вільного обертання в горизонтальній площині. На ньому встановлений контролер польоту Pixhawk 2, бортовий комп'ютер Raspberry Pi 1, GPS-модуль 7, мотори 5 та акумуляторна батарея 11.

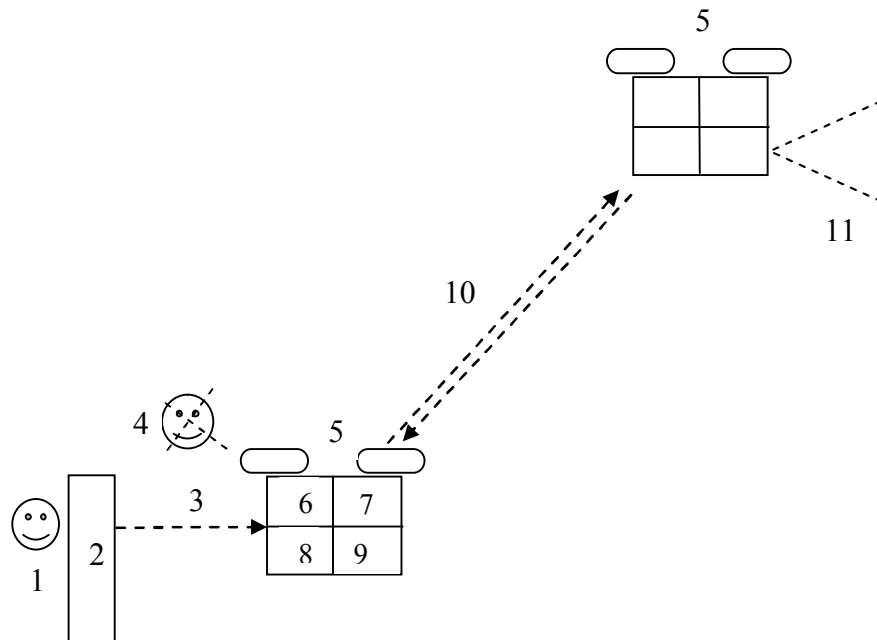


Рис.1. Схема комп'ютерної моделі керування БПЛА. 1 – користувач, 2 – робочий сіл користувача, 3 – команди на бортовий комп'ютер, 4 – зайвий оператор польотів, 5 – БПЛА, 6 – бортовий комп'ютер, 7 – автопілот, 8 – відеокамера, 9 – виконавчий механізм, 10 – траєкторії польоту, 11 – відео знімання об'єктів і виконання дії впливу на об'єкти.

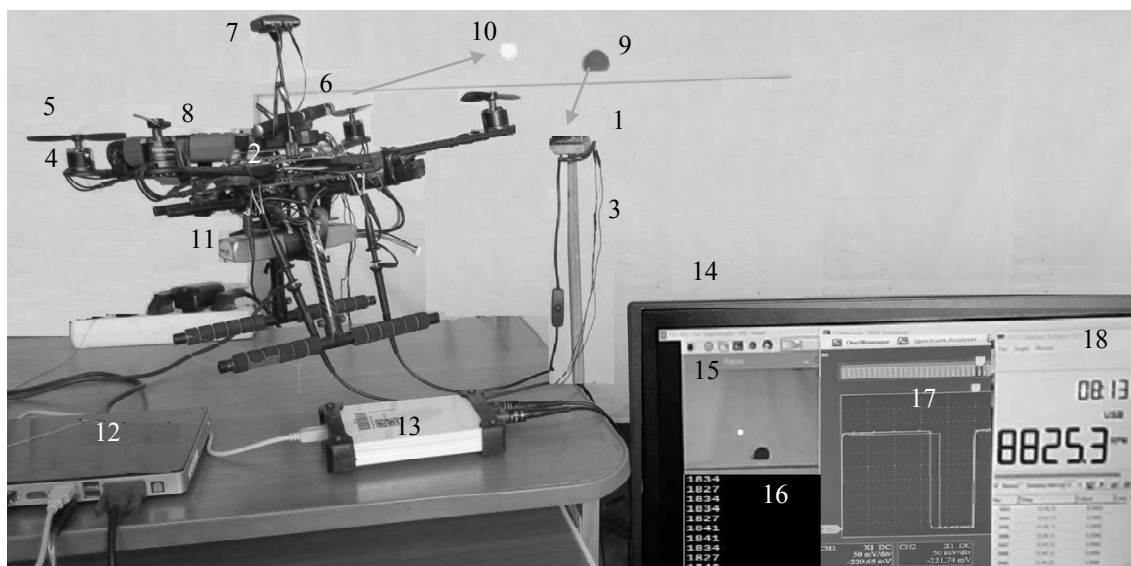


Рис.2. Фотографія установки. 1 – комп'ютер Raspberry Pi, 2 – автопілот, 3 – з'єднання з автопілотом, 4 – мотори, 5 – пропелери, 6 – лазерний ліхтарик, 7 – GPS-приймач, 8 – лазерний тахометр, 9 – червоний диск, 10 – пляма від променя, 11 – акумулятор, 12 – комп'ютер стаціонарний, 13 – осцилографічна приставка, 14 – монітор, 15 – вікно терміналу, 16 – обчислені кількості імпульсів, 17 – вікно з осцилограмами ШІМ-імпульсів, 18 – вікно тахометра зі швидкістю обертання мотора.

За допомогою лазерного ліхтарика 6 вимірювали відхилення квадрокоптера (10 – пляма від лазерного променя) при обертанні у випадку фотозахоплення і переміщення об'єкта переслідування – червоного диска 9. При цьому на вікні терміналу 15 (на моніторі 14 стаціонарного комп'ютера 12) появлялось зображення захопленого об'єкта, а у вікні 16 – обчислені за власною формулою кількості ШІМ-імпульсів, які повинні подаватись на мотори. За допомогою осцилографічної приставки 13 вимірювали фактичні ширини згенерованих ШІМ-імпульсів (17 – вікно з осцилограмами на моніторі). Лазерним тахометром 8 вимірювали швидкість обертання одного із моторів коптера (18 – вікно тахометра на моніторі). Для сценарію фотопереслідування використали ноутбук і WiFi роутер.

З'єднання бортового комп'ютера з автопілотом виконали за схемою на рис.3. На цьому ж рисунку зображено часові залежності виміряних відхилень зображення захопленого об'єкта в горизонтальній площині, а також ширини ШІМ-імпульсів, які подавались на регулятори моторів. Як видно, спостерігається достатня кореляція між цими параметрами у повному часовому діапазоні.

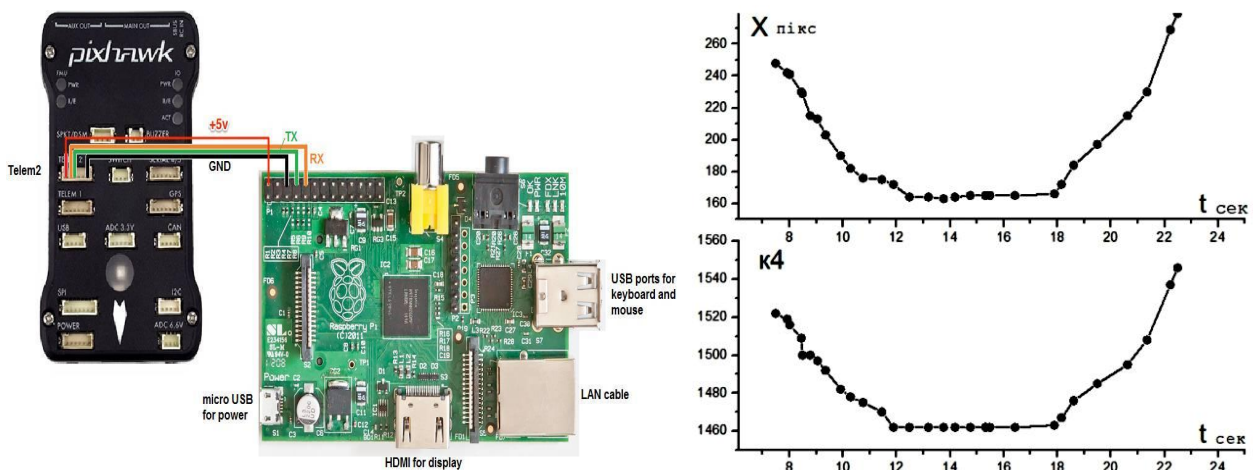


Рис.3. Зліва – фотографія з'єднання автопілота Pixhawk і бортового комп'ютера Raspberry. Справа – часові графіки виміряних залежностей відхилення зображення об'єкта (в пікселях) і виміряної ширини одного з ШІМ-імпульсів.

Алгоритм і програма сценарію фотопереслідування

Опрацьовано алгоритм і створена програма фотозахоплення об'єкта в польоті бортовою апаратурою БПЛА, а також його переслідування з використанням автопілота Pixhawk Px4 (рис.4). Підпрограма опрацювання відеоряду зображень об'єкта використовує такі бібліотеки, як sys, OpenCV, Picamera, RPI.GPIO, datetime, time, errno, platform. До основних слід віднести бібліотеку OpenCV – для аналізу та опрацювання зображення, знаходження та розпізнавання об'єкта, бібліотеку Picamera – для зв'язку бортового комп'ютера з бортовою камерою. Підпрограма складається з одного основного циклу, кожна ітерація якого виконується з появою нового кадру із відеокамери. Використано колір об'єкта, як основну характеристику при розпізнаванні. Застосовується кольорова модель HSV, що базується на трьох основних характеристиках кольору: тоні, насиченості та яскравості. Для кожного кадру проводиться ряд операцій перед виявленням об'єкта. Зокрема, для зменшення деталізації та зашумленості застосовується розмивання Гауса – функція GaussianBlur() з бібліотеки OpenCV. Проводиться бінеризація зображення за допомогою функції inRange(), при якій відбувається перехід від кольорового (або в градаціях сірого) зображення до двокольорового чорно-білого. Програма опрацьовує контури білих фігур і обводиться найбільший з них колом. Встановлюються його параметри – радіус, площу, координати центра X і

У. Всі ці операції виконуються функцією бібліотеки OpenCV – `minEnclosingCircle()`. У випадку, коли отримані координати виходять за межі центра матриці, програма відповідним чином керує механізмами повертання коптером для компенсації цього відхилення. При наближенні БПЛА до об'єкта, коли радіус кола навколо зображення є більшим від заданого, програма дає сигнал на відповідний порт GPIO Raspberry Pi для виконання дії знешкодження.

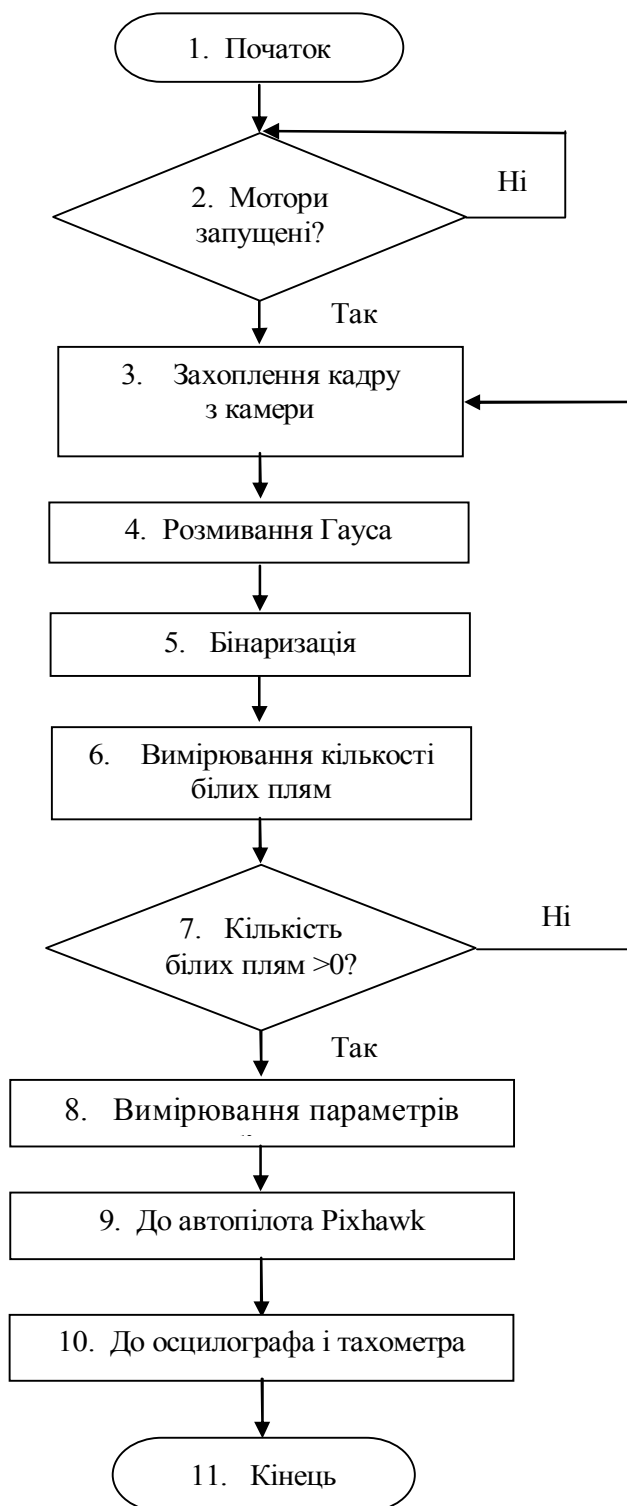


Рис. 4. Схема алгоритму фотопереслідування об'єкта

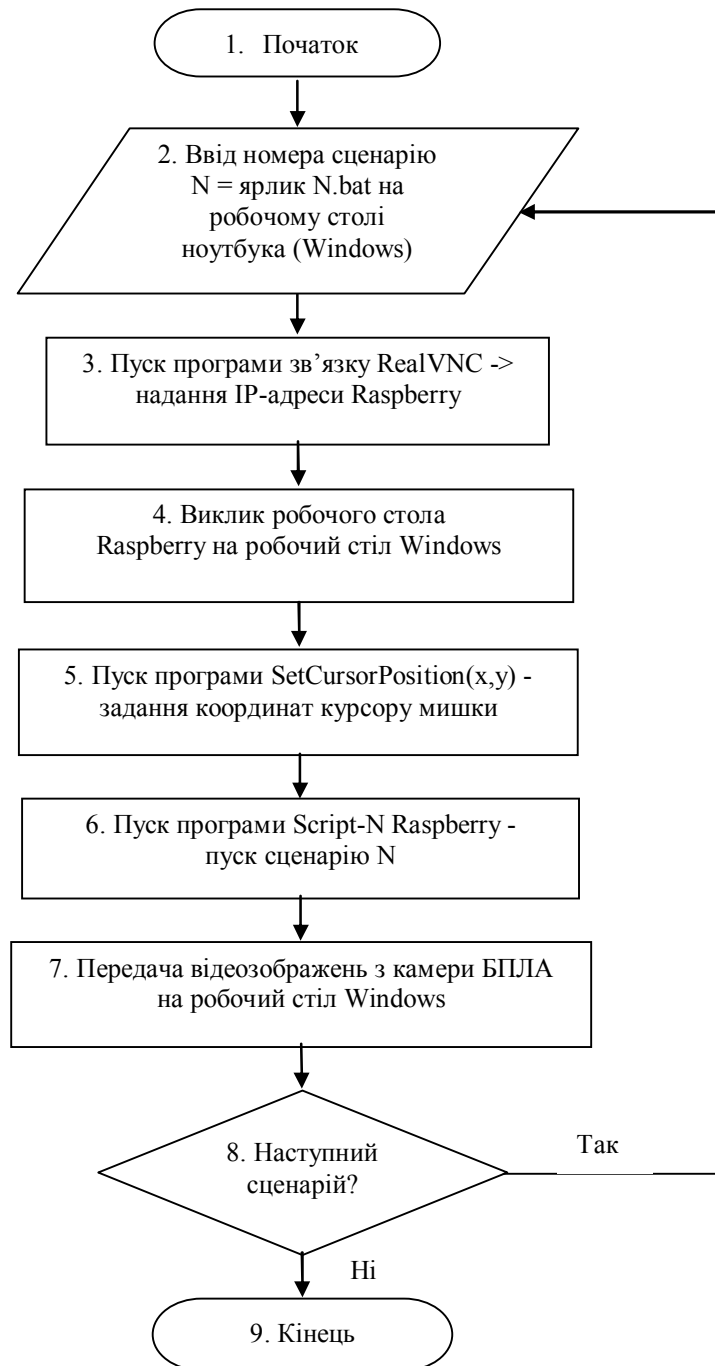


Рис. 5. Схема алгоритму спрощеного виконання сценарію польоту і отримання відеозображень на робочому столі користувача

Результати експериментальних досліджень

За схемою, наведеною на рис.1, опрацьовано алгоритм і створена програма пуску БПЛА з робочого стола користувача та виконання наперед заданих сценаріїв польоту і відеопередавання зображень об'єкта на цей же робочий стіл користувача без використання проміжного стандартного пульта та оператора керування (рис. 5).

Для моделювання використано командний Bat.file і типову програму зв'язку двох комп'ютерів VNC (Virtual Network Computing), в даному випадку – персонального комп'ютера чи

ноутбука з операційною системою Windows і бортового комп'ютера Raspberry з операційною системою Linux, на якому мовою Python записано власні програми керування і фото захоплення та фотопереслідування. RealVNC – це протокол надання доступу до віддаленого комп'ютера у мережі TCP/IP з будь-якого іншого комп'ютера або мобільного пристрою з метою відслідковування (моніторингу) та дистанційного керування (remote control) [18].

Як видно з алгоритму (рис.5), на робочому столі ноутбука попередньо створюються ярлики N.bat (N – номер сценарію) з використанням пакетного файлу з розширенням .bat. Пуск програми зв'язку RealVNC відбувається з попереднім наданням IP-адреси комп'ютера Raspberry. При цьому робочий стіл Raspberry викликається на робочий стіл Windows. Наступним кроком є пуск програми SetCursorPosition(x,y), яку створили для задання координат курсору мишки і відповідного пуску програми Script-N на робочому столі Raspberry, тобто пуску сценарію N, одного із попередньо створених сценаріїв польоту. При його виконанні відбувається пряма передача відеозображень з камери БПЛА на робочий стіл Windows – робочий стіл користувача, що є новим і спрощеним у процесі експлуатації БПЛА, і не вимагає роботи підготовленого оператора з пультом керування.

Висновки

Створено комп'ютерну модель керування безпілотним літальним апаратом з застосуванням дистанційних хмарних обчислень за заздалегідь заданими сценаріями з робочого стола користувача. Для дослідження працездатності цієї моделі створено експериментальну установку на базі персонального комп'ютера з операційною системою Windows, безпілотного літального апарата типу квадрокоптера з використанням автопілота PixHawk PX4 та бортового комп'ютера Raspberry-3 з операційною системою Linux і відеокамерою Pi Camera V2.

Для моделювання процесів керування БПЛА (пуску і виконання різних сценаріїв польоту) і передавання відеозображень на комп'ютері Raspberry-3 мовою Python записано власні програми керування і фотопереслідування.

За отриманими результатами і на основі літературних даних запропоновано модель керування безпілотним літальним апаратом з робочого стола персонального комп'ютера користувача через бортовий комп'ютер Raspberry без використання стандартного пульта керування та оператора. Опрацьовано алгоритм і програму приєднання робочого стола користувача до бортового комп'ютера.

Запропонована комп'ютерна модель дозволяє пряму передачу відеозображень з камери БПЛА на робочий стіл Windows – робочий стіл користувача, що є новим і спрощеним у процесі експлуатації БПЛА, не вимагає роботи підготовленого оператора з пультом керування, та економить час на підготовку і введення сценаріїв польоту в польових умовах.

1. V. Chyhin, M. Protsenko, Yu. Shabatura, M. Bugayov. *Improving the method of detecting unmanned aerial vehicles based on the results of spectral analysis of acoustic signals. Military-technical collection of DIA*, 2019. N20, pp. 58-63. doi:10.33577/2312-4458.20.2019.58-63.

2. Vasyl Chyhin, Pavlo Mykhailishyn. *Experimental unmanned aerial vehicle for photo capture. Bulletin of Khmelnytsky National University*. 2019, № 2 (271), p. 202-206. doi: 10.31891/2307-5732-2019-271-2-202-206.

3. Vasyl Chyhin, Pavlo Mykhailishyn. *Experimental studies of unmanned aerial vehicles during photo capture. Bulletin of Khmelnytsky National University*. 2020, N3 (285), p. 186-188. doi: 10.31891/2307-5732-2020-285-3-28.

4. V. Glotov, A. Gunina, Y. Teleschuk. *Analysis of the possibilities of using unmanned aerial vehicles for military purposes. Photogrammetry, geographic information systems and cartography*. V. 1 (33), 2017. P.139-146. Available at: https://scholar.google.com.ua/citations?view_op=view_citation&hl=uk&user=ML3LP8MAAAAJ&citation_for_view=ML3LP8MAAAAJ:4JMBOYKVnBMC (Accessed: 05 December 2021).

5. M. Lavrovsky. *Development of unmanned aerial vehicles in Ukraine and the world to perform civil defense tasks*. Scientific Bulletin of NLTU of Ukraine, 2017, vol. 27, № 1. P.151-153.
Available at: https://nv.nltu.edu.ua/Archive/2017/27_1/37.pdf (Accessed: 05 November 2021).
6. V. Chyhin, M. Chernenko. *Experimental system and software for the study of photodetection and pursuit of moving objects by unmanned aerial vehicles*. Bulletin of Khmelnytsky National University. 2020, № 4 (287), p. 84-88. doi:10.31891/2307-5732-2020-287-4-84-88.
7. RaspberryPi. Available at: <https://www.raspberrypi.org/products> (Accessed: 05 November 2021).
8. HardKernel. Available at: <https://www.hardkernel.comshopodroid-c2> (Accessed: 05 December 2021)
9. Nastolnye-kompyuterylattepanada. Available at: <https://hotline.uacomputer-nastolnye-kompyuterylattepanada-lattepanada-2g32gb> (Accessed: 05 November 2021).
10. Mission-planner . Available at: <http://www.ardupilot.org/wiki/ardupilot/install-mission-planner.html> (Accessed: 05 December 2021).
11. Flylitchi . Available at: <https://flylitchi.com> (Accessed: 05 November 2021).
12. QGroundControl . Available at: <http://qgroundcontrol.com> (Accessed: 05 November 2021).
13. Ballon Finder [. Available at: <https://www.youtube.com/watch?v=yRmXwRqPesY&feature=youtu.be>.
14. Active track 2.0 на dji mavic 2 pro. Available at: <https://www.youtube.com/watch?v=qEmd5g2fMcE&feature=youtu.be> (Accessed: 05 November 2021).
15. DroneKit . Available at: <https://dronekit.io> (Accessed: 05 November 2021).
16. OpenCV . Available at: <https://opencv.org> (Accessed: 05 November 2021).
17. PiCamera . Available at: <https://picamera.readthedocs.io> (Accessed: 05 November 2021).
18. Virtual_Network_Computing . Available at: https://uk.wikipedia.org/wiki/Virtual_Network_Computing (Accessed: 05 November 2021).

CREATION OF COMPUTER FACILITIES FOR CONTROLLING AN AUTONOMOUS AIRCRAFT WITH THE USE OF REMOTE CLOUD CALCULATIONS

Vasyl Chyhin

Lviv Polytechnic National University,
Computer Engineering Department

© Chyhin V., 2021

The possibility of creating a computer control system for an unmanned aerial vehicle using remote cloud computing according to predefined scenarios from the user's desktop is investigated. For this, an experimental setup was created, which includes a quadcopter, a personal computer with the Windows operating system, an on-board computer Raspberry-3 with the Linux operating system, a Pi Camera V2 camcorder, and a Pixhawk autopilot. To model the control and transmission of video images the own control programs and photo pursuit on a computer Raspberry-3 in Python are recorded. Based on the obtained results, a model of unmanned aerial vehicle control from the desktop of the user's personal computer via the on-board computer without the use of a standard control panel and operator is proposed.

Keywords: computer control model, quadcopter, autopilot, on-board computer, user desktop