

УДК 617.57-77

МАНОЙЛЕНКО О. П., КАЛІНІНА К. Р.,
ЛЕБЕДЕНКО Ю. О., ДІДИК М. А.

Київський національний університет технологій та дизайну, Україна

ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ УПРАВЛІННЯ МЕХАТРОНИМИ СИСТЕМАМИ ТИПУ "СОБОТ" З ВИКОРИСТАННЯМ ML-ТЕХНОЛОГІЙ

Мета. Роботи є дослідження та розроблення конструкції руки-маніпулятора, його мехатронної системи та програмного керування, з особливим акцентом на розгорнутій кінематиці біонічної кисті руки типу "Cobot" [1, 2]. Вдосконалення методів управління мехатронними системами та розроблення алгоритмів системи керування для створення інноваційної системи управління рукою-маніпулятором типу "Cobot", заснованої на ML-технологіях машинного зору та платформі Arduino.

Розробка керуючих алгоритмів на мові програмування Python та їх інтеграція з Arduino, використовуючи стандартні фреймворки: "CV2", "MediaPipe", "Numpy" та "Serial" [3–6]. Кодування біомеханічних суглобів для застосування в комп'ютерному імітаційному моделюванні та дистанційному керуванні з використанням технологій штучного зору.

Методика. Методика дослідження включала в себе використання математичних методів для аналізу кінематики біонічної кисті руки та розрахунку координат та кінематики механічної системи. Розробка програмних алгоритмів керування базувалася на мові програмування Python. Для оптимізації коду застосовані стандартні фреймворки: "CV2", "MediaPipe", "Numpy" та "Serial". Також використовувалася інтеграція програмного забезпечення з мікроконтролером Arduino для управління мехатронічною системою.

Результати. Розроблено інноваційну систему керування на базі руки-маніпулятора типу "Cobot", яка реагує на рухи та жести кисті руки перед web-камерою. Розроблена блок-схема взаємодія електронних та електромеханічних пристроїв блоку керування кисті руки-маніпулятора "Cobot". Реалізовано керуючі алгоритми, які можуть застосовуватися в різних галузях машинобудування, включаючи медицину, промисловість та область доповненої та віртуальної реальності. Представлені алгоритми для розпізнавання жестів, розширюючи можливості використання методів машинного навчання в різних галузях. Наведені залежності кінематики пальців біонічної кисті руки типу "Cobot".

Практична значимість. Отримані результати вказують на потужний потенціал використання інтелектуальних систем управління мехатронними пристроями. Система може бути застосована в медицині, промисловості та області доповненої та віртуальної реальності. Робота сприяє розвитку наукових платформ і є корисною для подальших досліджень у сфері мехатроніки та інтелектуального управління.

Ключові слова: штучний інтелект; мехатронні системи; комп'ютерний зір; машинний зір; рука-маніпулятор; дистанційне керування; Arduino; мова програмування Python.

Вступ. В сучасному світі технологічний прогрес надзвичайно швидко розвивається, сьогоднішній світ не можна представити без робототехнічних пристроїв, Інтернет речей (IoT-технологій) [7], AR/VR/MR-технологій [8], штучного інтелекту (AI-технологій) [9] та інших технологій Індустрії 4.0 [10]. Розвиток вказаних технологій обумовлений стрімкими науковими досягненнями в IT-галузі, впровадженням робототехнічних, мехатронних систем, а також інших систем з дистанційним керуванням з застосуванням вищезгаданих технологій.

Сучасні засоби керування робототехнічними пристроями різноманітні і включають в себе голосове керування, використання різних сенсорів і датчиків, віддалене керування через Інтернет, жести та маніпуляції, інтеграцію з віртуальною реальністю та навіть мозковий інтерфейс так звані BCI-системи [11]. Вибір конкретного методу залежить від призначення пристрою та завдань, які він повинен виконувати.

Але в сучасному світі все більш поширюється використання сучасних технологій, зокрема штучного інтелекту. Ця технологія вже стала відомою і доступною для широкого кола користувачів.

Розробка технічних засобів для контролю технологічних машин на основі машинного зору є актуальною і перспективною галуззю досліджень і практичного впровадження, оскільки вона сприяє підвищенню автоматизації та якості виробництва, зменшенню людського втручання та ризику помилок в технологічних процесах.

На сьогодні широке застосування все більше знаходять маніпулятори, що копіюють рухи людини. Ця технологія використовується в різних сферах, таких як медицина, промисловість, віртуальна реальність та інші. У медицині, наприклад, хірургічні роботи можуть виконуватися з великою точністю завдяки маніпуляторам, які повторюють рухи лікаря [12]. У віртуальній (VR) та доповненій реальності (AR) ця технологія дозволяє користувачам взаємодіяти з віртуальними об'єктами, використовуючи свої рухи [10].

Маніпулятори, що копіюють рухи людини, також знаходять своє застосування в промисловості, де вони можуть використовуватися для автоматизації завдань, які вимагають високої точності, координації рухів та застосовуються в небезпечному для людини середовищі. Ця технологія допомагає підвищити продуктивність та знизити ризик пошкодження матеріалів, готових виробів та зберегти здоров'я, або навіть життя людині [13].

Постановка завдання. Метою цієї роботи є створення доступної системи керування, яка базується на технологіях машинного зору та платформі Arduino. Основний акцент у роботі ставиться на розробці руки-маніпулятора типу "Cobot" [1, 2] з системою керування, побудованою на технологіях машинного зору. Поєднання штучного інтелекту, зокрема комп'ютерного зору, з роботизованою рукою відкриває можливість створення інноваційного методу керування, який має численні переваги, включаючи зручність та безпечність в користуванні, ефективність та економічність. Додатковим важливим аспектом є зменшення потреби в активній взаємодії з робочою поверхнею елементів керування (пульт ДК, джойстик, консолі керування та ін.) [14].

Результат дослідження.

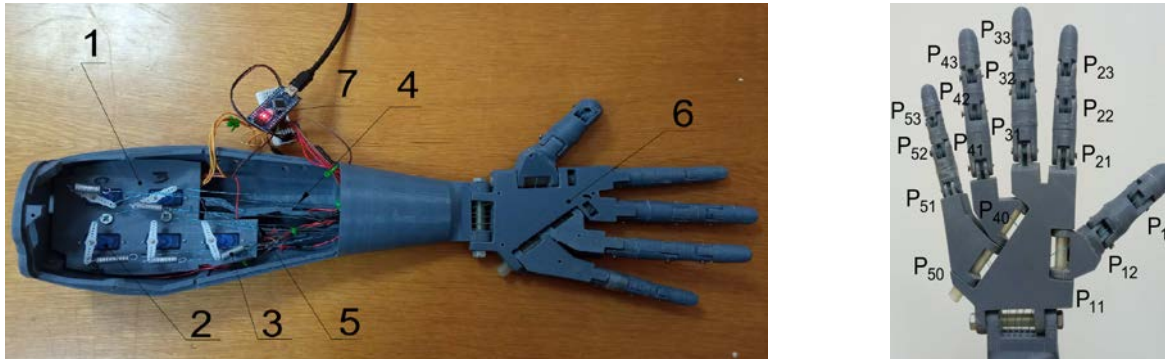
Механічна складова руки-маніпулятора. Рука-маніпулятор типу «Cobot» розроблена та виготовлена шляхом 3D друку на базі конструкції 3D моделі [15]. Привід біонічної кисті руки-маніпулятора містить блок приводу 1 (рис. 1а), який побудований на 5-ти міні сервоприводах типу SG90 – 2 з моментом пуску 2 кг/см. Кожне коромисло сервоприводу 2 з'єднане через пружний елемент 3 (виконаний у вигляді пружин 6x26x0,8) з верхньою тягою 4 та безпосередньо з нижньою тягою 5, які виконані з металевого дроту. Тяги 4 та 5 (рис. 2) через систему нитконапрямників $G_{i,j}^{B,H}$ з'єднані з сегментами (фалангами) I-III – біомеханічних пальців 1–5. Кожний сегмент I-III з'єднаний між собою шарнірами $P_{i,j}$ (рис. 1а, 2) (де i -тому пальцю відповідають j -ті шарнірні з'єднання конкретного елемента, відповідно, I – проксимальний (ближчий до долоні), II – середній та III – дистальний (найближчий до кінця пальця)). При цьому пружні елементи 3 та тяги 4, 5 біонічної кисті руки-маніпулятора відіграють функції «Актонів» [16] біологічної кінцівки, забезпечуючи, тим самим певне їх положення. Кожний з сервоприводів 2 (рис. 2) системою тяг 4 та 5 приводить в рух окремий палець кисті 6 руки-маніпулятора та керується платою Arduino Nano – 7.

Таким чином механізм контролю пальців реалізується через натягнуті дроти тяги, які з'єднують сервоприводи 2 з кінцями сегментів (фалангів пальців) – III.

Функція положення суглобів n -го суглоба, яка враховує його положення відносно попередніх суглобів в системі координат X та Y може бути визначена виразом:

$$P(x, y) = \sum_{i=1}^n [l_i \cdot \cos(\sum_{j=1}^i \theta_j), l_i \cdot \sin(\sum_{j=1}^i \theta_j)], \quad (1)$$

де $l_{i,j}$ – довжина j -го сегменту (суглобу) певного i -го пальця;
 $\theta_{i,j}$ – кут повороту- j -го сегменту (суглобу) певного i -го пальця.



а) компоновка складових частин

б) позначення шарнірів кисті
 руки-маніпулятор

Рис. 1. Конструкція руки-маніпулятор типу "Cobot"

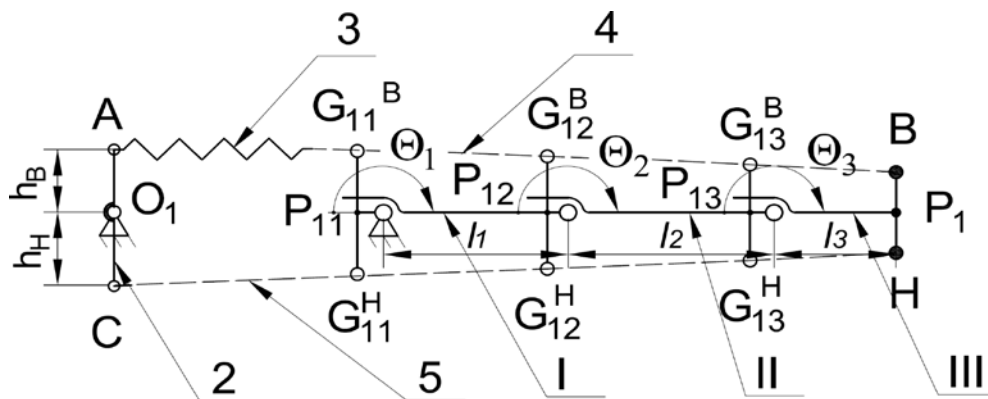


Рис. 2. Кінематика біомеханічного «великого пальця» руки-маніпулятор типу "Cobot"

Для кожного шарнірного з'єднання (суглоба) $P_{i,j}$ можна визначити силу згинання (розгинання) $F_{зг,i,j}$, як функцію прикладеної сили та кута повороту (згинання) $\theta_{i,j}$ в кожному $P_{i,j}$ шарнірі (суглобі).

Момент сил згинання $M_{зг,i,j}$ в кожного шарнірі (суглобі) буде дорівнювати:

$$M_{зг,i} = F_{зг,i} \cdot h_{i,j}. \quad (2)$$

Тоді узагальнене рівняння рівноваги моментів для всієї системи $P_{i,j}$ шарнірів (суглобів) матиме вигляд:

$$\sum_{j=1}^{i,j} M_{зг,i,j} = \sum_{j=1}^{i,j} R_{i,j} \cdot l_{i,j} \cdot \cos(\alpha_{i,j}), \quad (3)$$

де $M_{зг,i,j}$ – згинаючий момент рівноваги сил j -го сегменту (відносно суглобу) певного i -го пальця;

$F_{зг,i,j}$ – приведена сила до механічної фаланги пальця;

$R_{i,j}$ – значення приведеної реакції сил в $P_{i,j}$ шарнірі (суглобі);

$\alpha_{i,j}$ – кут прикладання між вектором реакції та осью ліній елемента.

Наведені залежності (1–3) можуть бути використані для визначення кінематичних характеристик руки-маніпулятора та раціонального вибору крутного моменту сервоприводу.

Керування та програмне забезпечення. Блок керування механічних пальців кисті руки-маніпулятора "Cobot" має вбудовану електроніку, яка побудована на платформі Arduino [17] та комп'ютерне програмне забезпечення керування, що включає алгоритми штучного інтелекту на базі технологій машинного зору. Схема взаємодії електронних та електромеханічних пристроїв зображена на рис. 3.

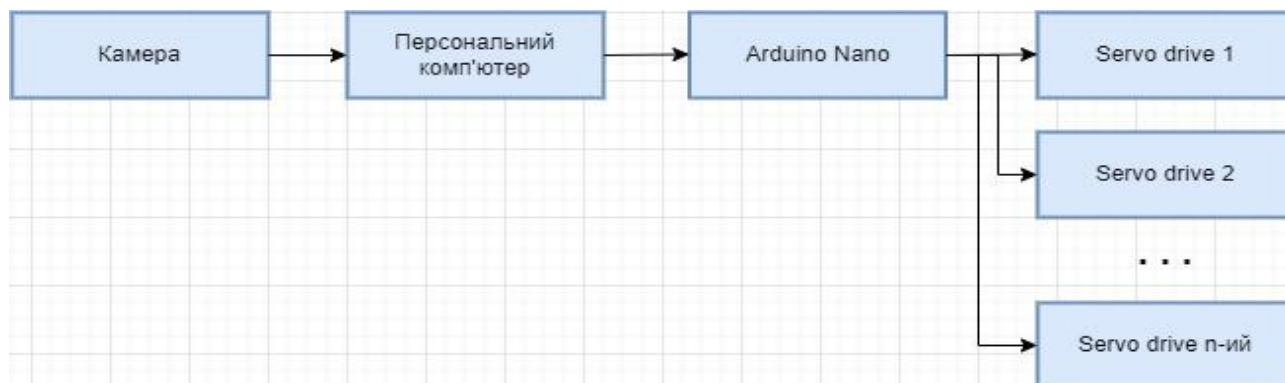


Рис. 3. Блок-схема взаємодія електронних та електромеханічних пристроїв блоку керування кисті руки-маніпулятора "Cobot"

Блок керування пальців кисті 1 (рис. 4) руки-маніпулятора "Cobot" 2 включає внутрішню web-камеру 3, ноутбука (далі ПК) 4 таке поєднання не потребує додаткового підключення, та забезпечує передачу сигналу через внутрішні шини інтерфейсу до операційної системи ПК. Операційна система ПК 4 пов'язана через контроллер USB 5 з контроллером зовнішньої плати керування Arduino Nano 6 та блоку 7 серводвигунів. Алгоритм керування блоку 7 сервоприводів базується на сигналі, який подається від web-камери 3 та базується на коді та її IP-адресі.

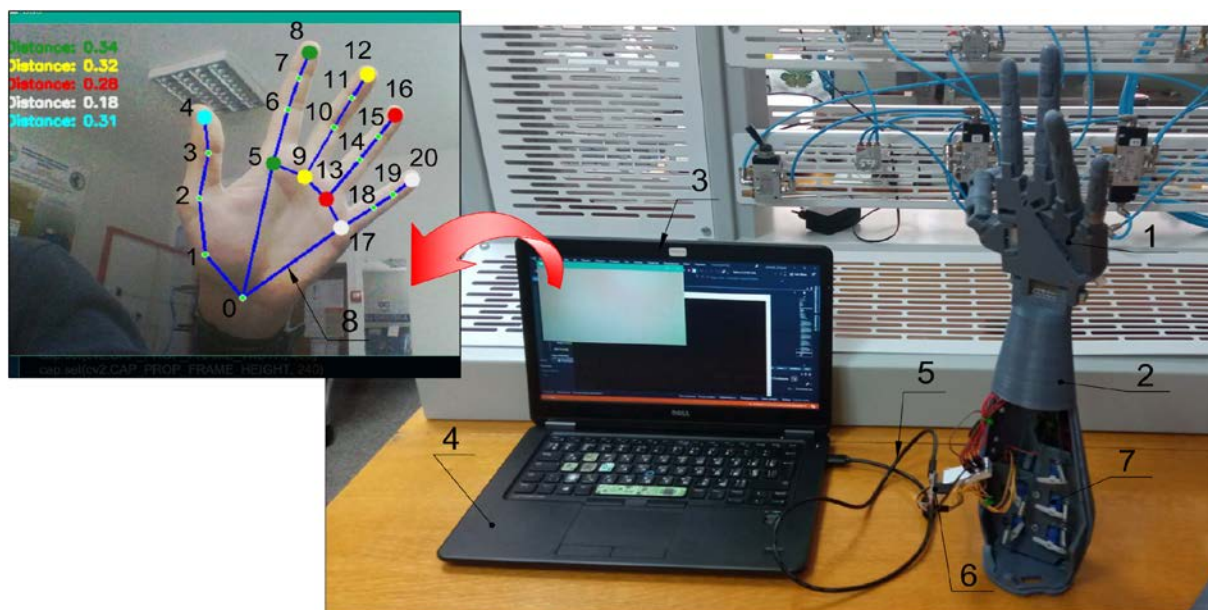


Рис. 4. Схема підключення руки-маніпулятора та сегментація руки оператора з накладеною «маскою кисті»

Принцип роботи блока керування базується на телеметрії поточної геометрії «маски кисті» 8 (рис. 4) за допомогою web-камери 3. Тобто оператор підносить розкриту долоню своєї руки до web-камери 3, ПК 4, таким чином, щоб зображення пальців повністю потрапило в кадр. Після цього керуюча програма автоматично накладає візуальний скелет – «маску кисті» 8 на долоню та пальці руки оператора. Певним точкам 1-20 «маски кисті» 8 (рис. 5) відповідають $P_{i,j}$ (рис. 1б) суглоби біомеханічної кисті руки-маніпулятора, т. 0 (рис. 4, 5) прийнято за початок локальної системи координат.

Після завантаження програми на мікроконтролер Arduino Nano, система може розпізнавати рухи руки оператора, такі як згинання та розгинання пальців або повторювати конкретні жести. Сформований керуючий сигнал, що базується на матриці відносних координат кожної точки від 1 до 20 (рис. 4), передається відповідній команді та через послідовний порт надсилається мікроконтролеру Arduino. Останній, у свою чергу, використовує цей керуючий сигнал для керування відповідним серводвигуном.

Наприклад, коли оператор згинає великий палець перед web-камерою, це призводить до зміни координат точок 1–4 (рис. 4) та трансформації «маски кисті» 8. У результаті цього процесу система розпізнає вказаний жест і автоматично відправляє відповідну команду на Arduino Nano. Мікроконтролер Arduino Nano шукає внутрішню програму для цієї команди і активує відповідний сервопривод, щоб виконати відповідний рух великого пальця.

Програмне забезпечення. Керуюча програма написана з використанням мови програмування, зокрема Python [18–21] та Arduino. Для програмування web-камери та взаємодії з контроллером Arduino [22], було використано програмне середовище Python.

У процесі розробки програмного забезпечення використовувалися стандартні фреймворки Visual Studio 2019 (64-біт), Arduino IDE 2.2.1 та Python 3.9. Завантаження нестандартних фреймворків здійснювалося вручну через командний рядок Windows.

В програмному коді Python використано 4 головні фреймворки, які реалізують наступне керування (рис. 5):

1. Фреймворк "CV2" (повна назва – OpenCV) забезпечує роботу з записом та обробкою зображень/відео [1].
2. Фреймворк "MediaPipe" забезпечує роботу з комп'ютерним зором, виявлення рук/обличчя, розпізнавання жестів тощо [2].
3. Фреймворк "Numpy" забезпечує роботу з числовими даними [3].
4. Фреймворк "Serial" забезпечує роботу з передачею даних через послідовний порт (COM-порт) [4].

```
import cv2 # робота з зображення відео/зображення
import mediapipe as mp # комп'ютерний зір, виявлення руки
import numpy as np # числові дані
import serial # під'єднання к Arduino
import math # стандарт
import time # час, таймер тощо
```

Рис. 5. Підключені фреймворки для роботи з кодом

Наведені фреймворки забезпечують взаємодію оператора з візуальним скелетом – «маскою руки» (рис. 6а), та забезпечують програмування для виконання конкретних операцій. Кожна точка від 0 до 20 має свою унікальну та складну назву (рис. 6б), але для зручності їх позначено простими цифрами від 0 до 20.

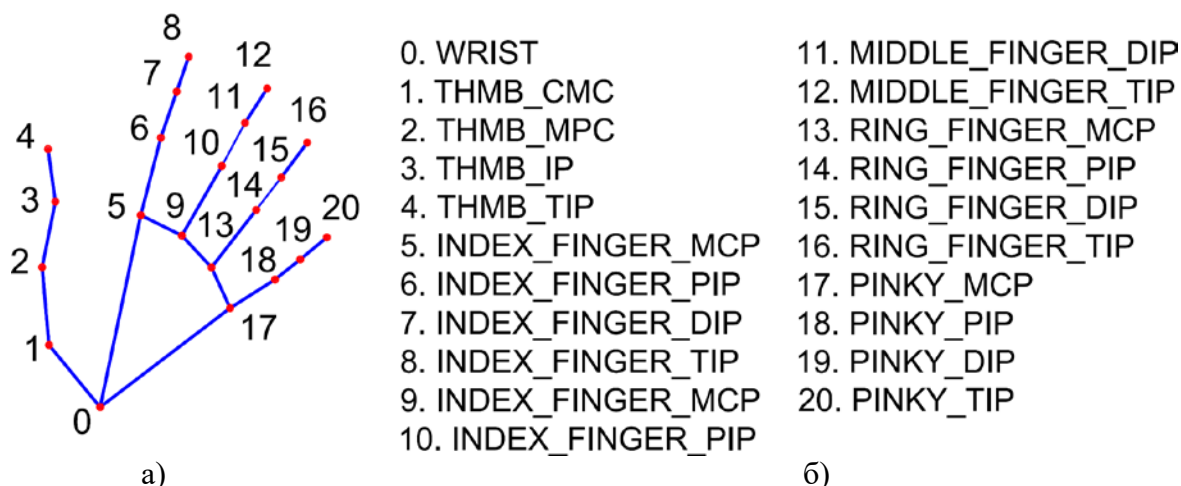


Рис. 6. Інформаційна модель «маски кисті»: а – схематичне представлення скелета «маски кисті»; б – анатомічні маркери характерних точок $P_{i,j}$

Для комп'ютерної симуляції «маски кисті» руки-маніпулятора були використані наступні точки, які були об'єднані у відрізки і пов'язані з рухомими частинами скелету (пальцями): $8=5$, $12=9$, $16=13$, $20=17$ та $4=13$. Кожна з програмованих точок має визначені координати X та Y.

Для обчислення відстані між двома точками певних відрізків на площині здійснювалося за методикою [23, 24]. Шляхом зміни відстані між точками, користувач може імітувати різні жести кисті руки-маніпулятора.

Наведений код фреймворка "MediaPipe" (рис. 7) здійснює визначення та фіксацію координат конкретного базового вузла $P_{i,j}$ пальця руки.

```
index_finger_mcp = hand_landmarks.landmark(mp_hands.HandLandmark.INDEX_FINGER_MCP)
index_finger_tip = hand_landmarks.landmark(mp_hands.HandLandmark.INDEX_FINGER_TIP)

image = cv2.circle(image, (int(index_finger_tip.x * image.shape[1]), int(index_finger_tip.y * image.shape[0])),
radius=8, color=(32, 117, 26), thickness=-1)
image = cv2.circle(image, (int(index_finger_mcp.x * image.shape[1]), int(index_finger_mcp.y * image.shape[0])),
radius=8, color=(32, 117, 26), thickness=-1)
```

Рис. 7. Фрагмент коду для вилучення координат потрібного маркера $P_{i,j}$ сегмента

Стабільний зв'язок з платою Arduino Nano здійснено прописаним кодом, який наведений на рис. 8.

```
arduino_port = 'COM7'
baud_rate = 9600
ser = serial.Serial(arduino_port, baud_rate)
cap = cv2.VideoCapture(0)
```

Рис. 8. Рядки коду підключення плати Arduino Nano

Стабільність роботи Arduino Nano забезпечується фрагментом коду (рис. 8) за умови:

1. Відповідності порту підключення, керування рукою-маніпулятор (в нашому випадку "COM 7").
2. Встановлення швидкості передачі даних відповідно до параметрів, визначених у середовищі розробки Arduino IDE.

3. Забезпечення фізичного з'єднання комп'ютера USB та плати Arduino Nano (обов'язкова умова для роботи з кодом Python).

4. Для використання коду Arduino слід врахувати, що нахили сервоприводів не фіксуються прямо у коді. Використовується команда "Serial.parseInt()", яка зчитує відстань з коду Python і автоматично перетворює її на необхідні нахили відповідно до конкретної логіки.

Висновок. У результаті дослідження і розробки інноваційної системи керування рукою-маніпулятором "Cobot" отримано суттєві результати в області інтелектуального управління мехатронними системами. Розроблені алгоритми проектування ML-систем на базі машинного зору з застосуванням мови програмування Python та її інтеграцію з Arduino за допомогою фреймворків "CV2", "MediaPipe", "Numpy", "Serial". Розроблено реактивну систему керування, здатну взаємодіяти з рухами кисті перед web-камерою.

Наведені загальні характеристики кінематики та силових розрахунків можуть бути корисні при подальшому дослідженні подібних робототехнічних систем типу "Cobot".

Результати підтверджують потенціал застосування в галузях медицини, промисловості та VR/AR технологій.

References

1. Peshkin, M. A., Colgate, J. E., Wannasuphprasit, W., Moore, C. A., Gillespie, R. B., Akella, P. (2001). Cobot Architecture. *IEEE transactions on robotics and automation*, P. 377–390.
2. Pavlenko, T. I., Shylo, N. Yu. (2019). Analiz kolaboratyvnykh robotiv [Analysis of collaborative robots]. *Automation and Development of Electronic Devices*, Part 2, P. 179–182 [in Ukrainian].
3. OpenCV-Python Tutorials. URL: https://docs.opencv.org/3.4/d6/d00/tutorial_py_root.html.
4. MediaPipe official documentation: Gesture recognition task guide. URL: https://developers.google.com/mediapipe/solutions/vision/gesture_recognizer#configurations_options.
5. NumPy official documentation. URL: <https://numpy.org/devdocs/user/>
6. PySerial official documentation. URL: <https://pyserial.readthedocs.io/en/latest/>
7. Pradyumna Gokhale, Omkar Bhat, Sagar Bhat (2018). Introduction to IoT. *International Advanced Research Journal in Science, Engineering and Technology*, pp. 41–44.
8. Radu, Iu., Joy, T., Schneider, B. (2021). Virtual Makerspaces: Merging AR/VR/MR to Enable Remote Collaborations in Physical Maker Activities. *CHI EA '21: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*, pp. 1–5.
9. Alsheibani, S., Cheung, Y., Messom, C. (2018). Artificial Intelligence Adoption: AI-readiness at Firm-Level. *Twenty-Second Pacific Asia Conference on Information Systems*, pp. 1–12.

Література

1. Peshkin M. A., Colgate J. E., Wannasuphprasit W., Moore C. A., Gillespie R. B., Akella P. Cobot Architecture. *IEEE transactions on robotics and automation*. 2001. P. 377–390.
2. Павленко Т. І., Шило Н. Ю. Аналіз колаборативних роботів. *Automation and Development of Electronic Devices*. 2019. Ч. 2. С. 179–182.
3. OpenCV-Python Tutorials. URL: https://docs.opencv.org/3.4/d6/d00/tutorial_py_root.html
4. MediaPipe офіційний довідник: Gesture recognition task guide. URL: https://developers.google.com/mediapipe/solutions/vision/gesture_recognizer#configurations_options.
5. NumPy official documentation. URL: <https://numpy.org/devdocs/user/>
6. PySerial official documentation. URL: <https://pyserial.readthedocs.io/en/latest/>
7. Pradyumna Gokhale, Omkar Bhat, Sagar Bhat. Introduction to IOT. *International Advanced Research Journal in Science, Engineering and Technology*. 2018. P. 41–44.
8. Radu Iu., Joy T., Schneider B. Virtual Makerspaces: Merging AR/VR/MR to Enable Remote Collaborations in Physical Maker Activities. *CHI EA '21: Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021. P. 1–5.

9. Alsheibani S., Cheung Y., Messom C. Artificial Intelligence Adoption: AI-readiness at Firm-Level. *Twenty-Second Pacific Asia*

10. Lasi, H., Kemper, H.-G., Fettke, P., Feld, T., Hoffmann, M. (2014). Industry 4.0. *Business & Information Systems Engineering*, pp. 239–242.
11. Martini, M. L., Oermann, E. K., Opie, N. L., Panov, F., Oxley, T., Yaeger, K. (2020). Sensor Modalities for Brain-Computer Interface Technology: A Comprehensive Literature Review. *Neurosurgery*, 86 (2), E108, DOI: 10.1093/neuros/nyz286.
12. Dvorakevych, A. O., Huraievskiy, A. A., Stasyshyn, A. R., Huraievskiy, D. A., Shevchuk, D. V., Kalinchuk, O. O. (2022). Pershyi dosvid zastosuvannya robot-asystovanoi khirurgii v dytiachomu vitsi v Ukraini [First experience of robot-assisted surgery in childhood in Ukraine]. *PEDIATRIC SURGERY. UKRAINE*, pp. 91–95 [in Ukrainian].
13. Krainiuk, O. V., Buts, Yu. V., Barbashyn, V. V., Severynov, O. V. (2022). Vykorystannia tekhnolohii virtualnoi ta dopovненоi realnosti dlia zabezpechennia bezpeky pratsi [The use of virtual and augmented reality technologies to ensure occupational safety]. *Komunalne hospodarstvo mist = Communal management of cities*, pp. 165–172 [in Ukrainian].
14. Kalinina, K. R., Skidan, V. V., Lebedenko, Yu. O. (2023). Prohramno-aparatna platforma dlia keruvannia mekhatronnymy systemamy za dopomohoiu zhestiv [Software and hardware platform for controlling mechatronic systems using gestures]. *Materialy Mizhnarodnoi naukovo-praktychnoi konferentsii "Nauka, osvita, tekhnolohii i suspilstvo v umovakh hlobalizatsii" = Materials of the International Scientific and Practical Conference "Science, Education, Technology and Society in the Conditions of Globalization"* (10.06.2023, Bila Tserkva), pp. 19–20 [in Ukrainian].
15. Website InMoov. URL: <https://inmoov.fr/inmoov-stl-parts-viewer/?bodyparts=Right-Hand>.
16. Morcecki, A., Ekiel, J., Fidclus, K. (1971). *Bionics of movement*. Warsaw.
17. Website Arduino.cc official documentation. URL: <https://docs.arduino.cc/learn/>
18. John Wiley & Sons, Inc. (2018). *Beginning Programming with Python For Dummies*. 2nd Edition. 379 p.
19. Packt Publishing (2018). *Clean Code in Python Refactor your legacy code base*. 311 p.
20. Website Python official documentation. URL: <https://www.python.org/>
10. Lasi H., Kemper H.-G., Fettke P., Feld T., Hoffmann M. *Industry 4.0. Business & Information Systems Engineering*. 2014. P. 239–242.
11. Martini M. L., Oermann E. K., Opie N. L., Panov F., Oxley T., Yaeger K. *Sensor Modalities for Brain-Computer Interface Technology: A Comprehensive Literature Review. Neurosurgery*. 2020. No. 86 (2). E108. DOI: 10.1093/neuros/nyz286.
12. Дворакевич А. О., Гураєвський А. А., Сташишин А. Р., Гураєвський Д. А., Шевчук Д. В., Калінчук О. О. Перший досвід застосування робот-асистованої хірургії в дитячому віці в Україні. *PAEDIATRIC SURGERY. UKRAINE*. 2022. С. 91–95.
13. Крайнюк О. В., Буц Ю. В., Барбашин В. В., Северинов О. В. Використання технологій віртуальної та доповненої реальності для забезпечення безпеки праці. *Комунальне господарство міст*. 2022. С. 165–172.
14. Калініна К. Р., Скідан В. В., Лебеденко Ю. О. Програмно-апаратна платформа для керування мехатронними системами за допомогою жестів. *Матеріали Міжнародної науково-практичної конференції "Наука, освіта, технології і суспільство в умовах глобалізації"* (10.06.2023, Біла Церква). С. 19–20.
15. Website InMoov. URL: <https://inmoov.fr/inmoov-stl-parts-viewer/?bodyparts=Right-Hand>.
16. Morcecki A., Ekiel J., Fidclus K. *Bionika ruchu*. Warszawa, 1971.
17. Website Arduino.cc Arduino.cc official documentation. URL: <https://docs.arduino.cc/learn/>
18. *Beginning Programming with Python For Dummies*. 2nd Edition. John Wiley & Sons, Inc., 2018. 379 p.
19. *Clean Code in Python Refactor your legacy code base*. Packt Publishing, 2018. 311 p.
20. Website Python official documentation. URL: <https://www.python.org/>

21. Putivnyk movoiu prohramuvannia Python [Guide to the Python programming language]. URL: <https://pythonguide.rozh2sch.org.ua/>
22. Website Real Python: Arduino With Python: How to Get Started. URL: <https://realpython.com/arduino-python/>
23. Lyksherstov, D. O., Lebedev, D. Yu. (2022). Porivnialnyi analiz shliakhiv vyznachennia zobrazhennia na videoriadi zasobamy mashynnoho navchannia [Comparative analysis of methods for determining an image on a video sequence using machine learnin]. *Zapysky TNU imeni V. I. Vernadskoho = Notes of TNU named after V. I. Vernadskyi*, pp. 20–26. URL: https://tech.vernadskyjournals.in.ua/journals/2022/5_2022/3.pdf.
24. Website OnlineMSchool. URL: https://ua.onlinemschool.com/math/library/analytic_geometry/point_point_length/
21. Путівник мовою програмування Python. URL: <https://pythonguide.rozh2sch.org.ua/>
22. Website Real Python: Arduino With Python: How to Get Started. URL: <https://realpython.com/arduino-python/>
23. Лихошерстов Д. О., Лебедев Д. Ю. Порівняльний аналіз шляхів визначення зображення на відеоряді засобами машинного навчання. *Записки ТНУ імені В. І. Вернадського*. 2022. С. 20–26. URL: https://tech.vernadskyjournals.in.ua/journals/2022/5_2022/3.pdf
24. Website OnlineMSchool. URL: https://ua.onlinemschool.com/math/library/analytic_geometry/point_point_length/

MANOILENKO OLEKSANDR

Candidate of Technical Sciences, Associate Professor,
Associate Professor of Department of Mechanical
Engineering, Kyiv National University
of Technologies and Design, Ukraine
<https://orcid.org/0000-0002-5670-4977>
Scopus Author ID: 57194469280
E-mail: manojlenko.op@knutd.edu.ua

LEBEDENKO YURI

Candidate of Technical Sciences, Associate Professor,
Associate Professor of Department of Information and
Computer Technologies, Kyiv National University
of Technologies and Design, Ukraine
<https://orcid.org/0000-0002-1352-9240>
Scopus Author ID: 57193066468
E-mail: lebedenko.yo@knutd.edu.ua

KALININA KRISTINA

Student of Department of Information and Computer
Technologies, Kyiv National University of Technologies
and Design, Ukraine
<https://orcid.org/0009-0006-8713-2584>
E-mail: kalinina.kr@knutd.edu.ua

DIDYK MYHAILO

Student of Department of Mechanical Engineering,
Kyiv National University of Technologies
and Design, Ukraine
<https://orcid.org/0009-0001-9692-902X>
E-mail: didykmyhajlo@gmail.com

MANOILENKO O. P., KALININA K. R., LEBEDENKO Yu. O., DIDYK M. A.

Kyiv National University of Technologies and Design, Ukraine

**INTELLIGENT CONTROL SYSTEMS OF COBOT-TYPE
MECHATRONIC SYSTEMS USING ML TECHNOLOGIES**

Goal. The works are research and development of the design of the manipulator hand, its mechatronic system and software control, with a special emphasis on the developed kinematics of the "Cobot" type bionic hand brush [1, 2]. Improvement of mechatronic systems control methods and development of control system algorithms for the creation of an innovative "Cobot" manipulator arm control system based on ML machine vision technologies and the Arduino platform.

Development of control algorithms in the Python programming language and their integration with Arduino using standard frameworks: "CV2", "MediaRipe", "Numpy" and "Serial" [3–6]. Coding of biomechanical joints for use in computer simulation modeling and remote control using artificial vision technologies.

Method. The research methodology included the use of mathematical methods to analyze the kinematics of the bionic hand and calculate the coordinates and kinematics of the mechanical system. The

development of software control algorithms was based on the Python programming language. Standard frameworks: "CV2", "MediaRipe", "Numpy" and "Serial" are used to optimize the code. Software integration with an Arduino microcontroller was also used to control the mechatronic system.

The results. An innovative control system has been developed based on a "Cobot" hand manipulator, which responds to hand movements and gestures in front of the web camera. A block diagram of the interaction of electronic and electromechanical devices of the hand control unit of the "Cobot" hand manipulator was developed. Control algorithms have been implemented that can be applied in various fields of mechanical engineering, including medicine, industry, and the field of augmented and virtual reality. Algorithms for gesture recognition are presented, expanding the possibilities of using machine learning methods in various fields. The dependences of the kinematics of the fingers of the "Cobot" type bionic brush hand are given.

Practical significance. The obtained results indicate the powerful potential of using intelligent control systems of mechatronic devices. The system can be applied in medicine, industry and the field of augmented and virtual reality. The work contributes to the development of scientific platforms and is useful for further research in the field of mechatronics and intelligent control.

Keywords: artificial intelligence; mechatronic systems; computer vision; machine vision; robotic arm; remote control; Arduino; Python programming language.