

ДОДАТОК А
Висвітлення результатів кваліфікаційної роботи

Міністерство освіти і науки України



NURE

Харківський національний університет
радіоелектроніки

ЗБІРНИК

студентських наукових статей

«Автоматизація та приладобудування»

«Automation and Development of Electronic Devices»

ADED-2023

(Випуск 2)

[електронне видання]



<http://nure.ua/department/kafedra-komp-yuterno-integrovanih-tehnologiy-avtomatizatsiyi-ta-mehatroniki-kitam>



<http://itez.zntu.edu.ua/>



<http://kafea.kdu.edu.ua>

Харків 2023

ЗМІСТ

<i>Я.І. Халімонов</i>	
Перспективи: Автоматизації вимірювання умов у житлових та робочих приміщеннях з використанням комп'ютерно-інтегрованих рішень	9
<i>С.Ю. Гавриков, А.Я. Осман</i>	
Дослідження технологій виробництва деталей на 3D принтері	12
<i>А.С. Андреев</i>	
QR-коди в науці та техніці	17
<i>Ф. Куррота</i>	
Development of Automated Environmental Control System for Portable Greenway Section ..	23
<i>К.К. Стеценко</i>	
Моделювання BEAM-робота в середовищі TINKERCAD	27
<i>О.В. Удовиченко</i>	
Вплив розвитку штучного інтелекту на комп'ютеризовані та робототехнічні системи ..	30
<i>Б.О. Чеснаков</i>	
3D моделювання роботизованої платформи для гуманітарного розмінуванні	33
<i>Є.В. Шевченко</i>	
Розробка кіберфізичної системи моніторингу технологічних процесів на виробництві ..	37
<i>Є.О. Єфімік</i>	
Розроблення концепт макету малогабаритного мобільного робота підвищеної прохідності	44
<i>М. Манічкін</i>	
Аналіз кінематики та розробка моделі розрахунків елементів матриці гомогенних перетворень для зооморфного мобільного робота	49
<i>М.М. Моргунов</i>	
Розробка методу передачі інформації всередині статичного зображення для мобільних роботів	55
<i>Є.С. Ключник</i>	
Аналіз систем автоматизованого свердління у Industry 4.0	61
<i>О.Д. Юрченко</i>	
Розроблення системи моніторингу роботи засобів виробництва та персоналу приладобудівного приміщення з використанням ESP32-CAM	66
<i>М.О. Бендеберя</i>	
Розробка алгоритмічно-функціональної моделі робота маніпулятора на базі ABB Robot Studio	74
<i>І.В. Балабанов</i>	
Визначення залежності часу та інтенсивності випромінювання на температуру фотополімерної смоли	79
<i>М.Д. Лисун</i>	
Аналіз кінематик 3D принтерів за технологією FDM/FFF	83
<i>С.В. Шматко</i>	
Аналіз сучасних роботів телеприсутності, як людського помічника	87
<i>І.С. Коваленко</i>	
Перспективи розвитку повітряної робототехніки	92
<i>М.С. Лубінець</i>	
Розроблення методу прокладення траєкторії руху робота-сапера на основі даних від металошукача	97

<i>О.О. Рак</i>	
Розробка автоматизованого модуля моніторингу параметрів об'єктів критичної інфраструктури	104
<i>О.І. Черненко</i>	
Автоматизація процесу сортування деталей на виробництві	109
<i>О.А. Тищенко</i>	
Моделювання пристрою позиціонування вантажного робота	114
<i>В.О. Веснянка</i>	
Розроблення інформаційної системи для оптимізації бізнес-процесів закладу харчування	121
<i>Ю.А. Бердник</i>	
Аналіз сучасних автономних роботизованих платформ	126
<i>М.В. Звєгінцев</i>	
Розробка модуля позиціонування сонячних панелей	133
<i>Д.Д. Леценко</i>	
Моделювання руху маніпулятора робота з використанням динамічної ланки з прямою та зворотною кінематикою	138
<i>П.М. Савченко</i>	
Огляд датчиків положення для обладнання, що працює в умовах аварійних відключень електроживлення	142
<i>П.М. Савченко</i>	
Створення сучасних систем управління з застосуванням мікропроцесорної техніки та засобів автоматизації	148
<i>Є.Р. Васильченко</i>	
Огляд принципів побудови пожежно-охоронної системи	153
<i>А.Д. Єчевський</i>	
Система моніторингу та управління параметрами мікроклімату в офісних приміщеннях	159
<i>А.І. Конєва</i>	
Перспективи розвитку безпілотних систем	164
<i>В.І. Фомін</i>	
Використання робототехнічних систем з елементами штучного інтелекту в приладобудуванні	171
<i>В.І. Фомін</i>	
Застосування 3D-друку у виробництві та промисловості	177
<i>О.В. Чернишенко</i>	
Оптимізація маршрутів в логістичних мережах виробничого процесу	182
<i>Р.Р. Шаталюк</i>	
Використання віртуальної та доповненої реальності для навчання та симуляцій у робототехніці	188
<i>Р.Р. Шаталюк</i>	
Програмування мікроконтролерів для автоматизації систем	193
<i>Т.А. Лихо</i>	
Вибір обладнання для розробки мобільного робота для відеонагляду	197
<i>В.О. Александров</i>	
Безпілотні літальні апарати. види, технічні особливості, автоматизація	203
<i>С.О. Вінниченко</i>	
Еволюція виробництва: Роль MES-системи у оптимізації та контролі промислових	208

процесів на підприємстві	
<i>А.В. Готовська</i>	
Підтримка прийняття рішень в технології проєктування роботизованого виробничого процесу	213
<i>Я.В. Олінкевич</i>	
Впровадження сgr-системи на виробництві	219
<i>М. Коваленко</i>	
Схема керування транспортними роботами на основі візуальних ознак	223
<i>В.К. Маковська</i>	
Контейнеризація та оркестрація: DOCKER та KUBERNETES	228
<i>Д.Р. Придятько</i>	
Огляд методів розпізнавання об'єктів за допомогою систем технічного зору	234
<i>А.А. Большаков</i>	
Розроблення архітектури SCADA-системи гнучкого виробництва та вибір апаратних засобів	239
<i>В.С. Головіна</i>	
Розроблення системи керування мобільним пошуково-рятувальним роботом	244
<i>Д.В. Мілько</i>	
Дослідження програмного методу визначення відстані до об'єкту за допомогою параметрів камери	250
<i>І.А. Манякін</i>	
Аналіз методів автоматичного розпізнавання осіб	254
<i>Ю.С. Візір</i>	
Автоматичне енергоефективне управління освітленістю з використанням кіберфізичних підходів в умовах виробництва	259
<i>В.І. Дульський</i>	
Методи оптимізації керуючих програм для верстатів з ЧПУ	264
<i>М.С. Карпов</i>	
Використання бездротових мереж для організації контролю в промисловості	269
<i>М.А. Пісклов</i>	
Алгоритми створення та оптимізації розкладу для загальноосвітніх навчальних закладів	275
<i>А.Ю. Губарь</i>	
Веб-додаток для моніторингу та управління запасами в 3D-друкарні	281
<i>І.А. Поддубняк</i>	
Аналіз сучасних візуальних SLAM систем в робототехніці	286
<i>Д.П. Редько</i>	
Технології транспортування вибухонебезпечних предметів за допомогою роботизованого пристрою	292
<i>В.О. Заїкін</i>	
Роботизовані системи та їх застосування у інноваційних методах виявлення та знешкодження вибухонебезпечних предметів	296
<i>К.О. Вадурін, А.С. Шандро</i>	
Розробка структури інформаційно-аналітичної система для збору, обробки та аналізу даних щодо використання енергетичних ресурсів багатоповерховою будівлею	302
<i>Є.М. Грищенко</i>	
Аналіз систем контролю виготовлення 3D деталей на потоковому роботизованому виробництві	309

<i>В.А. Савін</i>	
Класифікація роботизованих систем для пошуку вибухонебезпечних предметів	319
<i>М. Збітнев</i>	
Аналіз мобільних робототехнічних платформ для гуманітарного розмінування	329
<i>В.А. Сторожук В.А., М.А. Вісковатов</i>	
Розробка інтелектуального модуля для моніторингу параметрів на базі ІоТ	334
<i>М.В. Толстий</i>	
Аналіз методів намотування дротів на станках з ЧПУ у роботизованому виробництві .	340
<i>В.В. Цешевський</i>	
Огляд сучасних конструктивних схем роботів для переміщення сходами	354
<i>О.О. Зибенко</i>	
Інновації та досягнення в електророзробній обробці: формування комп'ютерно-інтегрованого виробництва	356
<i>К.О. Левченко</i>	
Моделювання автоматизованого комплексу безтарного сховища сировини	361
<i>О.Д. Нікулін</i>	
Конвеєрні технології та автоматизація у аддитивному виробництві	364
<i>Д.В. Пархоменко</i>	
Аналіз систем інжекції з'єднувальної речовини у технології 3D друку 3DP	370
<i>К.С. Скрипник</i>	
Моделювання та розрахунок дозування пластику у шнековому екструдері	374
<i>С.Ю. Мірошніченко</i>	
Автоматизована система управління для знешкодження вибухонебезпечних предметів	381
<i>В.С. Тараненко</i>	
технологія екструзійного 3D друк без підтримок	386
<i>Є.О. Зуєв, М.Ю. Лучанінов</i>	
Дослідження методів автономного позиціонування та навігації робототехнічних мобільних платформ	390
<i>О.С. Пащенко, К.О. Зозуля</i>	
Сучасне виробництво з використанням комп'ютерного управління та інформаційних технологій	394
<i>Є.Г. Федосєєв</i>	
Аналіз методів імітаційного моделювання технологічних процесів складання	401
<i>К.С. Редькін</i>	
Локальна навігація мобільного робота в приміщенні	404

УДК 004.75

ВИБІР ОБЛАДНАННЯ ДЛЯ РОЗРОБКИ МОБІЛЬНОГО РОБОТА ДЛЯ ВІДЕОНАГЛЯДУ

Т.А. Лихо

Харківський національний університет радіоелектроніки

Україна, 62433, Люботин, вул. Шмідта, 9

E-mail: tymur.lykho@nure.ua

Анотація: У даній статті розглядається підбір компонентів для розробки мобільного робота відеонагляду. Будемо розглядати можливі компоненти, з яких буде складатись даний робот, аналізувати їх та обирати оптимальний варіант для проекту. В результаті роботи буде отримано список основних компонентів для розробки мобільного робота відеонагляду.

Ключові слова: мобільний робот, відеонагляд, вибір компонентів, ESP32-CAM.

EQUIPMENT SELECTION FOR THE DEVELOPMENT OF A MOBILE ROBOT FOR SURVEILLANCE

T. Lykho

Kharkiv National University of Radio Electronics

Ukraine, 62433, Liubotyn, Schmidt Str., 9

E-mail: tymur.lykho@nure.ua

Annotation: This article explores the component selection for the development of a mobile surveillance robot. We will consider potential components comprising the robot, analyze them, and choose the optimal options for the project. The outcome of the work will be a list of essential components for the development of a mobile surveillance robot.

Keywords: mobile robot, surveillance, component selection, ESP32-CAM.

АКТУАЛЬНІСТЬ РОБОТИ. У наші часи все більше набирають обертів системи охоронної та/або пожежної сигналізації, «розумний дім» і подібні. Зростання попиту на такі системи спричинене тим, що наразі в країні війна і багато людей були вимушені покинути свої будинки заради безпеки. Для контролю власного будинку, промислового приміщення, тощо пропонується альтернативний варіант відеонагляду, такий як мобільний робот з камерою, вибір обладнання для розробки такого робота буде наведено у даній роботі.

МАТЕРІАЛ І РЕЗУЛЬТАТИ ДОСЛІДЖЕНЬ.

1. Технічні вимоги до камери.

Розгляд обладнання почнемо з технічних вимог до камери. Головна задача камери на мобільному роботі – прийом зображення з оточуючого середовища і передача даних на мікроконтролер для подальшої обробки. Камера повинна мати:

- достатню роздільну здатність (800×600 і більше) для забезпечення нормального відображення зображення користувачу;
- оптимальний кут огляду для керування роботом у приміщенні, таким можна вважати кут огляду 60-70 градусів, вважається що це середнє значення, але цього буде достатньо для забезпечення передачі деталізованого зображення предметів на відстані до 10 метрів;
- компактний розмір для забезпечення компактності всього робота.

Провівши аналіз можливих на ринку камер було обрано камеру OV2640, характеристики якої наведено у таблиці 1 [1].

Таблиця 1 – Основні характеристики камери OV2640

Розмір матриці, пікселів	1600×1200 (UXGA)
Розмір лінзи, дюймів	0,25
Чутливість, В/Люкс-сек	0,6
Кут оглядку, °	66
Напруга живлення, В	1,7 – 3,3
Споживання енергії, мВт	125 – 140

Бачимо, що дана камера задовольняє потребам проекту.

2. Конструкція робота.

Головною ознакою мобільного робота є його мобільність, тож розглянемо можливі варіанти платформ для забезпечення пересування робота.

Платформа повинна мати:

- невеликі габарити;
- мінімум два ведучих колеса, для забезпечення можливостей повороту, розвороту платформи;
- колеса діаметром від 6 см, для забезпечення високої прохідності по приміщенню.

За наведеними вище вимогами було обрано дві платформи, розглянемо кожен з них. На рисунку 1 показано платформу, що має 4 колеса, з яких 2 ведучих, вона компактна, має два «поверхи» діаметром 140 мм. Головним недоліком такої платформи є її погана стійкість, що може призводити до перевертання платформи, тому є доцільним розглянути інший варіант платформи.



Рисунок 1 – 4-колісна платформа з двома ведучими [2]

На рисунку 2 показано платформу з трьома колесами, з яких 2 ведучих. Такий варіант платформи є оптимальним для даного проекту, бо він є більш стійким, ніж попередній варіант платформи.



Рисунок 2 – 3-колісна платформа з двома ведучими [3]

Наведена на рисунку 2 платформа має:

- два двигуни постійного струму з коефіцієнтом уповільнення 48;
- два колеса діаметром 65 мм;
- псевдоколесо;
- відсік для батарейок;
- габарити: ширина – 150 мм, довжина – 200 мм, висота – 48 мм.

Обрана робоплатформа задовольняє всім потребам проекту, тому такий варіант є доцільним.

Для керування двигунами також потрібен драйвер, для цього було обрано найпопулярніший модуль на базі мікросхеми L298 (рис. 3). Він може керувати двома моторами, підтримує роботу з керуючим мікроконтролером з напругою рівнів 3,3В.

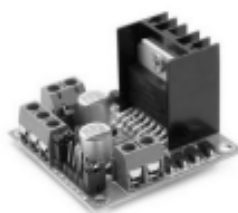


Рисунок 3 – Модуль на базі мікросхеми L298 [4]

Детально ознайомитись з характеристиками мікросхеми L298 можна у документації [5].

Також вдалим доповненням буде реалізувати можливість вертикального та горизонтального повороту камери, що дозволить швидко та зручно регулювати положення камери. Це можливо реалізувати встановивши камеру на штатив, в який інтегровано два серводвигуни.

3. Система живлення.

Живлення мобільного робота повинно бути засноване на використанні акумуляторних батарей з можливістю їх підзарядки.

У мобільних роботах часто використовуються літій-іонні (Li-ion) та літій-полімерні (Li-Po) акумуляторні батареї. Ось чому:

- висока енергетична ємність: Літій-іонні та літій-полімерні акумулятори мають високий відносний обсяг енергії, що дозволяє забезпечити тривалу роботу мобільного пристрою на одному заряді;
- малий розмір та вага: Ці акумулятори мають високий відносний обсяг енергії при порівняно невеликому об'ємі та масі, що важливо для мобільних роботів, де обмежені простір та вага;
- низька саморозрядка: Літій-іонні батареї мають низький рівень саморозряду, тобто вони втрачають мало енергії, коли не використовуються, що важливо для забезпечення довготривалого зберігання та готовності до роботи;
- швидкий заряд: Більшість літій-іонних та літій-полімерних батарей можуть бути швидко заряджені, що є важливим для мобільних роботів, які можуть потребувати швидкого підзарядження для максимальної продуктивності;
- довгий термін служби: Літій-іонні та літій-полімерні акумулятори зазвичай мають довгий термін служби, що робить їх ефективними для застосувань, де важлива надійність та тривалість.

Ці переваги роблять літій-іонні та літій-полімерні акумулятори популярними вибором для живлення мобільних роботів.

Для забезпечення безперервної роботи мобільного робота можна розглянути варіант док-станції для підзарядки елементів живлення, таке рішення часто використовується для підзарядки роботів-порохотягів.

4. Вибір мікроконтролера.

Для реалізації проекту потрібно обрати мікроконтролер, що забезпечить з'єднання через Wi-Fi, передачу зображення у реальному часі з оптимальними показниками якості та компактність конструкції.

На ринку існують мікроконтролери з вбудованим Wi-Fi та Bluetooth, наприклад, ESP32, ESP8266. Реалізувати підключення обраної раніше камери OV2640 можливо до плати розробника, наприклад, ESP32-WROOM-32D, але через великий шлейф дротів конструкція буде сильно громісткою і ненадійною, а якість передачі зображення не буде відповідати мінімальним потребам.

Оптимальним варіантом для проекту «Мобільний робот для відеонагляду» є модуль ESP32-CAM (рис.4). Цей модуль поєднує у собі мікроконтролер ESP-32S з додатковою оперативною пам'яттю 4 Мбайти та відеокамеру OV2640. Основні характеристики ESP32-CAM:

- WiFi 802.11b / g / n;
- Bluetooth 4.2 LE з друкованою антеною, роз'єм u.FL. (IPX13);
- 32 Мбіт SPI флеш-пам'ять;
- 4 Мб PSRAM;
- Слот для мікро SD-карти до 4 Гб;
- Габаритні розміри – 40,5 x 27 x 4,5 мм;
- Вага - 10 гр.



Рисунок 4 – Модуль ESP32-CAM [6]

Детально ознайомитись з характеристиками можна у документації [7].

Модуль ESP32-CAM дозволить керувати роботизованою платформою, яку було описано вище.

5. Можливість дистанційного керування.

Дистанційне керування може бути реалізовано за допомогою програмного застосунку, що буде відображати зображення з камери у реальному часі.

Програмний застосунок може використовуватись як через браузер на ПК, так і через додаток для ОС Android.

6. Вартість обладнання

Розглянемо вартість всіх основних компонентів, що необхідні для реалізації проекту «Мобільний робот для відеонагляду», таблиця 2.

Таблиця 2 – Вартість основних компонентів

Назва компоненту	Кількість	Вартість одиниці, грн	Сума, грн
Модуль ESP32-CAM з камерою OV2640	1	248,00	248,00
Роботизована платформа Car Chassis 2WD	1	134,00	134,00
Контролер шагового двигуна L298N	1	62,50	62,50
Серводвигун SERVO-SG90	2	73,50	147,00
Сумарна вартість обладнання			591,5

Враховуючи, що звичайна IP-камера коштує приблизно 1000 гривень можна вважати, що розробка є економічно вигідною.

ВИСНОВКИ.

1. Основні результати досліджень:
 - обрано камеру OV2640 з високою роздільною здатністю та оптимальним кутом огляду для мобільного робота відеонагляду;
 - вибрана 3-колісна платформа з двома ведучими колесами для забезпечення стійкості та компактності робота;
 - використано драйвер L298 для керування двигунами та серводвигуни SG90 для реалізації можливості вертикального та горизонтального повороту камери;
 - вибрано літій-іонні або літій-полімерні акумулятори для живлення мобільного робота через їх високу енергетичну щільність та інші переваги;
 - використано модуль ESP32-CAM для реалізації бездротового підключення та передачі зображення у реальному часі.
 2. Практична цінність:
 - застосування мобільного робота відеонагляду актуально в умовах війни, коли безпека стає пріоритетом, і багато людей покидають свої будинки;
 - використання дистанційного керування та можливості перегляду зображення у реальному часі робить систему зручною та функціональною.
 3. Подальші перспективи:
 - можливість розширення функціоналу мобільного робота, наприклад, додавання додаткових сенсорів, систем аналізу даних, тощо;
 - застосування штучного інтелекту для автономного навігації та прийняття рішень;
 - інтеграція з системами «розумного дому» для більш широкого застосування в повсякденному житті;
 - дослідження можливостей використання мобільного робота в екстремальних умовах та розробка захисту від впливу навколишнього середовища.
- Загалом, розробка мобільного робота для відеонагляду має перспективи в різних сферах, привертає увагу своєю ефективністю та можливістю використання в різноманітних сценаріях.

ЛІТЕРАТУРА

1. OV2640 2 MPixel product brief. Sunnyvale : OmniVision, 2006. 2 p. URL: <https://pdf1.alldatasheetru.com/datasheet-pdf/view/312407/OMNIVISION/OV2640.html> (date of access: 10.11.2023).
2. 4-колісна платформа з двома ведучими. Arduino Придбати в Києві, Україна. URL: https://arduino.ua/products_pictures/medium_ARC101-1.jpg (дата звернення: 10.11.2023).
3. 3-колісна платформа з двома ведучими. Arduino Придбати в Києві, Україна. URL: https://arduino.ua/products_pictures/large_ARC100-1.jpg (дата звернення: 10.11.2023).

4. Модуль на базі мікросхеми L298. Arduino Придбати в Києві, Україна. URL: https://arduino.ua/products_pictures/large_L298moduleRed-2.jpg (дата звернення: 13.11.2023).
5. L298N datasheet. Italy : STMicroelectronics, 2000. 13 p. URL: <https://html.alldatasheet.com/html-pdf/22440/STMICROELECTRONICS/L298N/21107/13/L298N.html> (date of access: 12.11.2023).
6. Модуль Wi-Fi ESP32-CAM з камерою 2MP. Arduino Придбати в Києві, Україна. URL: https://arduino.ua/products_pictures/large_aoc673_1.jpg (дата звернення: 13.11.2023).
7. ESP32-CAM development board. 5 p. URL: https://media.digikey.com/pdf/Data%20Sheets/DFRobot%20PDFs/DFR0602_Web.pdf (date of access: 13.11.2023).

***Науковий керівник:** Максимова Світлана Святославівна, доцент кафедри, кандидат технічних наук, доцент кафедри КІТАМ, Харківський національний університет радіоелектроніки.*

ДОДАТОК Б
Висвітлення результатів кваліфікаційної роботи

Міністерство освіти і науки України



NURE

Харківський національний університет
радіоелектроніки

ЗБІРНИК

студентських наукових статей

«Автоматизація та приладобудування»

«Automation and Development of Electronic Devices»

ADED-2024

(Випуск 1)

[електронне видання]



<http://nure.ua/department/kafedra-komp-yuterno-integrovanih-tehnologiy-avtomatizatsiyi-ta-mehatroniki-kitam>



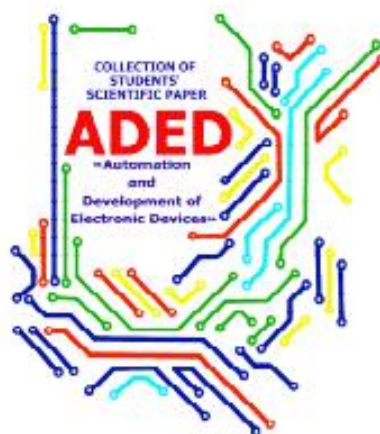
<http://itez.zntu.edu.ua/>



<http://kafea.kdu.edu.ua>

Харків 2024

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки
кафедра комп'ютерно-інтегрованих технологій, автоматизації та робототехніки
(КІТАР)



ЗБІРНИК
студентських наукових статей
«Автоматизація та приладобудування»
«Automation and Development of Electronic Devices»
ADED-2024
(Випуск 1)
[електронне видання]

Харків 2024

Головний редактор	Невлюдов Ігор Шакирович , доктор технічних наук, професор, завідувач кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.
Редакційна колегія:	<p>Филипенко Олександр Іванович, доктор технічних наук, професор, декан факультету Автоматики та комп'ютеризованих технологій, Харківського національного університету радіоелектроніки.</p> <p>Цимбал Олександр Михайлович, доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.</p> <p>Андрусевич Анатолій Олександрович, доктор технічних наук, професор, начальник Криворізького коледжу національного авіаційного університету</p> <p>Косенко Віктор Васильович, доктор технічних наук, професор, зам. директора Державного підприємства «Південний державний проектно-конструкторський та науково-дослідний інститут авіаційної промисловості».</p> <p>Замірець Микола Васильович, доктор технічних наук, професор, директор Державного підприємства Науково-дослідного технологічного інституту приладобудування.</p> <p>Свищ Володимир Митрофанович, доктор технічних наук, професор, радник директора Державне науково-виробниче підприємство «Об'єднання Комунар».</p> <p>Фомовська Олена Владиславівна, кандидат технічних наук, доцент завідувач кафедри «Електронних апаратів» Кременчуцького національного університету імені Михайла Остроградського.</p> <p>Кухаренко Дмитро Володимирович, кандидат технічних наук, доцент кафедри «Електронних апаратів» Кременчуцького національного університету імені Михайла Остроградського</p> <p>Демська Наталія Павлівна, кандидат технічних наук, доцент кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.</p> <p>Фурманова Наталія Іванівна, кандидат технічних наук, доцент, в.о. декана факультета Радіоелектроніки і телекомунікацій, Національного університету «Запорізька політехніка».</p>
Відповідальний редактор:	Євсєєв Владислав В'ячеславович , доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.

Автоматизація та Приладобудування («Automation and Development of Electronic Devices» ADED-2024) [Електронний ресурс]: збірник студентських наукових статей / Харківський національний університет радіоелектроніки ; [редкол.: І.Ш. Невлюдов та ін.]. – Харків : ХНУРЕ, 2024. – Вип. 1. – 207с.

Collection of Students' Scientific Paper «Automation and Development Of Electronic Devices» ADED-2024 Part 1 (Key infrastructure 2024) - Kharkiv/ The Editorial: Nevlyudov I.Sh. (head), that all. Kharkiv: Kind of Kharkiv National University of Radio Elekrtonics [electronic edition], 2024. – 207p with.

Рекомендовано рішенням
Науково-технічної ради
Харківського національного
університету радіоелектроніки
протокол №6 від 29.11.2018

Рекомендовано рішенням Вченої ради
факультету Автоматики і комп'ютеризованих технологій
Харківського національного
університету радіоелектроніки
протокол № 10 від 20.05.2024

Збірник містить наукові статті здобувачів першого (бакалаврського), другого (магістерського) рівнів вищої освіти кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР) Харківського національного університету радіоелектроніки, кафедри Інформаційних технологій електронних засобів (ІТЕД) Запорізького національного технічного університету та кафедри Електронних апаратів (ЕА) Кременчуцького національного університету ім. М. Остроградського які навчаються за спеціальностями: 151 Автоматизація та комп'ютерно-інтегровані технології, 174 Автоматизація, комп'ютерно-інтегровані технології та робототехніка; 172 Телекомунікації та радіотехніка, 171 Електроніка та 163 Біомедична інженерія. Статті надані в авторській редакції.

©ХНУРЕ, 2024 рік

ЗМІСТ

<i>Візір Ю.С.</i>	
Штучний інтелект у системах управління освітленістю	7
<i>Тимошенко М.В.</i>	
Огляд комп'ютерних телекомунікаційних мереж та технологій	12
<i>Бендеберя М.О.</i>	
Розробка алгоритмічно-функціональної моделі робота маніпулятора на базі ABB ROBOT STUDIO	18
<i>Дяченко Е.С.</i>	
Сучасні формати даних та їх вплив на швидкодію ВЕБ-додатків	23
<i>Karpenko A.</i>	
Overview at Autonomous Construction Development Tendencies	29
<i>Мороз М. В.</i>	
Необхідність та актуальність програмного забезпечення для автоматизації розсилки повідомлень	35
<i>Натарова В.С.</i>	
Інтеграція датчиків та контрольних систем для оптимізації параметрів вирощування рослин на основі технологій гідропонних	41
<i>Остапенко І.В.</i>	
Дослідження методів керування ТП з використанням робототехнічних засобів	47
<i>Редькін К.С.</i>	
Вдосконалення модуля автоматизованого управління режимами роботи теплообмінника на центральному тепловому пункті	51
<i>Савченко П.М.</i>	
Аналіз принципів побудови адаптивних систем автоматичного управління	55
<i>Савченко П.М.</i>	
Використання інтелектуальних технологій у створенні та вдосконаленні програмного забезпечення систем управління роботами	59
<i>Соломатін В.О.</i>	
Розробка системи сповіщення про стан пристрою дозування пластичних матеріалів	63
<i>R. Maksim</i>	
The Way to Efficient Production: Cals Approaches for Managing Product Data	70
<i>Тимошенко М.В.</i>	
Аналіз структури сучасної системи контролю та управління доступом	75
<i>Кирпота Ф.В.</i>	
Роль автоматизованої системи контролю навколишнього середовища теплиці	80
<i>Біліченко А.С.</i>	
Аналіз проблем і можливостей, пов'язаних з пошуком інформації в мережі інтернет ...	85
<i>Манякін І.А.</i>	
Пошукові технології у медичній сфері: відкриття та перспективи	91
<i>S.V. Shmatko</i>	
Evolution of Information and Search Systems From Beginnings to Present: Review	96
<i>Васильченко С.Р.</i>	
Аналіз функцій та основних принципів роботи охоронно-пожежної сигналізації	101
<i>Халімонов Я.І</i>	
Використання сенсорів та IoT-технології для моніторингу параметрів робочого середовища	106

<i>R. Maksim</i>	
Strategies for Implementation of Production Automation Using CALS Approaches	111
<i>Андреев А.С.</i>	
Пошук інформації в інтернеті: Проблеми та можливості	116
<i>Yechevskiy A.D.</i>	
System Of Monitoring and Control of Microclimate Parameters in Office Premises	122
<i>Лихо Т.А.</i>	
Роль розпізнавання образів та комп'ютерного зору в удосконаленні робототехнічних систем підтримки рішень	127
<i>Макушев І.А.</i>	
Огляд та актуальність сучасних повітряних дронів	133
<i>Соколов Т.О.</i>	
Роль інтелектуальних систем підтримки рішень в автоматизації та оптимізації робототехнічних процесів	138
<i>Зарубін І.С.</i>	
Огляд сучасних повітряних роботів	144
<i>Остроухов Є.С.</i>	
Дистанційно керовані роботи – нові можливості для медичної допомоги	150
<i>Придятько Д.Р.</i>	
Аналіз методів пошуку вибухонебезпечних предметів	155
<i>Shmatko S.V.</i>	
Impact of Information Search Systems on Users and Society	161
<i>Удовиченко О.В.</i>	
Застосування штучного інтелекту в промисловості та автомобільній галузі	166
<i>Фомін В.І.</i>	
Математичні методи в системах автоматизації	169
<i>Фомін В.І.</i>	
Етика та правові аспекти в робототехніці	173
<i>Черноморченко Б.О.</i>	
Аналіз інтелектуальних систем забезпечення безпеки виробництва	177
<i>Шаталюк Р.Р.</i>	
Виклики та перспективи впровадження адаптивних роботів у виробництво	182
<i>Шаталюк Р.Р.</i>	
Оцінка впливу роботизації на продуктивність та якість виробництв	187
<i>Довбня М.</i>	
Аналіз лабораторних блоків живлення, представлених на ринку електроніки	192
<i>Довбня М.</i>	
Порівняльний аналіз дронів для розмінування українських територій	200

УДК 004.9

РОЛЬ РОЗПІЗНАВАННЯ ОБРАЗІВ ТА КОМП'ЮТЕРНОГО ЗОРУ В УДОСКОНАЛЕННІ РОБОТОТЕХНІЧНИХ СИСТЕМ ПІДТРИМКИ РІШЕНЬ

Т.А. Лихо

Харківський національний університет радіоелектроніки
Україна, 62433, Люботин, вул. Шмідта, 9
E-mail: tymur.lykho@nure.ua

Анотація: У даному дослідженні розглянуто роль процесу розпізнавання образів та комп'ютерного зору в удосконаленні робототехнічних систем підтримки рішень. Проаналізовані основні алгоритми розпізнавання, такі як згорткові нейронні мережі, машини опорних векторів, SLAM та відстеження об'єктів. Також висвітлені сфери застосування комп'ютерного зору, призначення РСППР та сучасні проблеми, з якими вони стикаються.

Ключові слова: комп'ютерний зір, розпізнавання образів, робототехнічні системи підтримки рішень.

THE ROLE OF IMAGE RECOGNITION AND COMPUTER VISION IN IMPROVING DECISION SUPPORT ROBOTIC SYSTEMS

T. Lykho

Kharkiv National University of Radio Electronics
Ukraine, 62433, Liubotyn, Schmidt Str., 9
E-mail: tymur.lykho@nure.ua

Annotation: This study will examine role of image recognition and computer vision processes in improving decision support robotic systems. The main recognition algorithms such as convolutional neural networks, support vector machines, SLAM, and object tracking will be analyzed. The applications of computer vision, purpose of DSRS, and current challenges they face will also be highlighted.

Keywords: computer vision, image recognition, decision support robotic systems.

У наші часи, коли інформатизація, автоматизація, роботизація всіх сфер життя людини набирають все більших обертів [1-6], удосконалення систем підтримки рішень проводиться щоденно і це не дивно, адже такі системи значно полегшують життя людей, роблячи його більш комфортним, безпечним та економічним.

Для початку розглянемо процес розпізнавання образів у комп'ютерному зорі (КЗ).

Розпізнавання образів у комп'ютерному зорі означає здатність робототехнічної системи розпізнавати форми, образи та елементи на зображеннях та/або відео. Даний процес є схожим на людське зорове сприйняття, тобто наш мозок інтерпретує візуальні дані, завдяки чому ми можемо робити висновки про характеристики того чи іншого об'єкта. У комп'ютерному зорі для розпізнавання образів використовують певні алгоритми, які можуть виявлять та класифікувати різні елементи візуальних даних, таких як об'єкти, обличчя, жести, сцени та інше [7].

Існує два основних алгоритми розпізнавання образів у комп'ютерному зорі:

– метод опорних векторів, відомий в англійській літературі як support vector machine (SVM). Даний алгоритм навчається на прикладах та використовується для класифікації об'єктів, даний підхід розкриває значні можливості для побудування адаптивних систем автоматичного керування [8];

– метод з використанням згорткових нейронних мереж (convolutional neural network, CNN). Даний алгоритм виділяє з вихідного зображення малі частини, що містять опорні характерні ознаки, такі, як ребра, контури, дуги та грані, а на наступних кроках обробки з цих частин розпізнаються більш складні фрагменти текстур, такі як кола, квадрати та інші фігури, які у свою чергу складають такі складні текстурні частини обличчя, колесо

Комп'ютерний зір, розпізнавання образів і системи підтримки та прийняття рішень (СППР) активно застосовуються для вирішення складних задач.

У таблиці 1 наведено основні алгоритми розпізнавання образів та комп'ютерного зору, їх короткий опис та роль у роботизованих системах підтримки та прийняття рішень.

Таблиця 1 – Роль розпізнавання образів та комп'ютерного зору у СППР.

Алгоритм	Опис	Роль в СППР
Згорткова нейронна мережа (CNN)	Глибокий алгоритм навчання, призначений для класифікації та розпізнавання зображень. Він витягує ознаки зображень та вивчає шаблони через кілька шарів згортки.	Згорткові нейронні мережі є ключовими для СППР у завданнях, таких як виявлення об'єктів, класифікація та розпізнавання. Вони допомагають роботам ідентифікувати об'єкти та приймати рішення на основі візуальних вхідних даних.
Машина опорних векторів (SVM)	Навчальний алгоритм, що використовується для класифікації та регресії. SVM знаходить оптимальну гіперплощину, яка розділяє різні класи в просторі ознак.	SVM відіграють важливу роль у СППР, забезпечуючи метод класифікації виявлених об'єктів на зображеннях. Вони допомагають роботам категоризувати візуальні дані та приймати відповідні рішення.
Однчасне визначення місцеположення та картографування (SLAM)	SLAM – це техніка, яка використовується для створення карти невідомого середовища та одночасного визначення місцезнаходження робота у цьому середовищі. Вона поєднує дані з датчиків, таких як камери, LiDAR та інерціальні мікроелектромеханічні датчики для виконання завдань локалізації та картографування.	SLAM є невід'ємною частиною СППР, оскільки вона дозволяє роботам автономно навігувати та працювати в невідомих або динамічних середовищах. Вона допомагає роботам будувати карти, локалізувати себе та планувати оптимальні шляхи для прийняття рішень.
Алгоритм відстеження об'єктів	Алгоритми відстеження об'єктів відслідковують рух об'єктів протягом часу в послідовності кадрів. Вони використовують такі техніки, як оцінка руху, відстеження ознак та фільтрування, щоб передбачити траєкторію об'єктів.	У СППР алгоритми відстеження об'єктів важливі для моніторингу та підтримки уваги до об'єктів інтересу. Вони дозволяють роботам визначати, як об'єкти змінюють своє місцеположення та взаємодіють один з одним в часі, що в свою чергу допомагає роботам приймати рішення на основі динамічних змін в оточуючому середовищі.

Процес розпізнавання образів і комп'ютерного зору відіграє важливу роль у робототехнічних системах підтримки рішень, схематично процес наведено на рисунку 1.

Цей процес є ітераційним і може включати зворотні зв'язки між етапами для покращення результатів. Точна реалізація залежить від конкретного застосування та використовуваних алгоритмів.

Детальніше про сфери застосування комп'ютерного зору.

Зараз комп'ютерний зір використовується у багатьох сферах, одним з найбільш важливих застосувань є обробка зображень в медицині. В більшості випадків, відеодані отримують за допомогою мікроскопії, рентгенографії, ангиографії, ультразвукових досліджень та томографії. Прикладом інформації, яка може бути отримана з таких відеоданих є виявлення пухлин, атеросклерозу чи інших злоякісних змін.

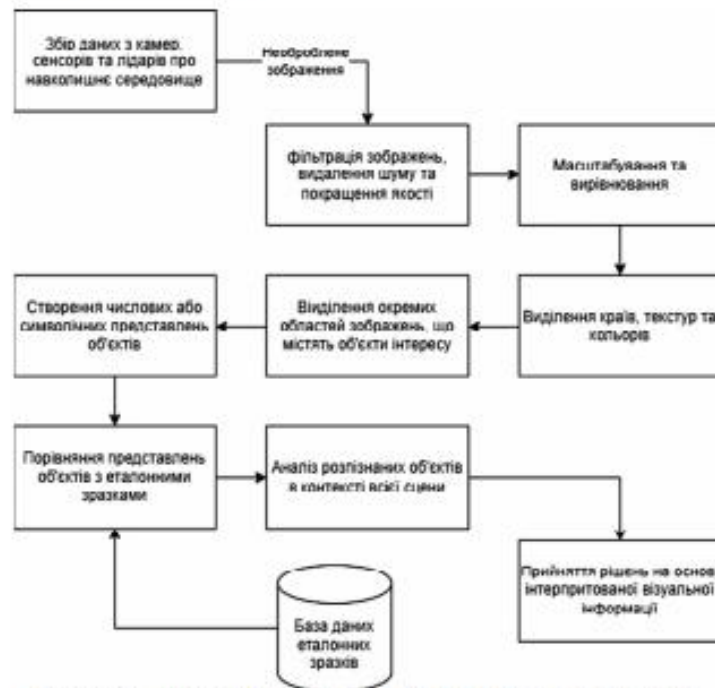


Рисунок 1 – Процес розпізнавання образів і комп'ютерного зору

Рідше комп'ютерний зір використовується у промисловості для підтримки виробничого процесу, наприклад, контроль якості, коли деталі чи кінцевий продукт автоматично перевіряють на наявність дефектів, або вимірювання положення та орієнтація деталей, які піднімає рука робота.

Також комп'ютерний зір використовується у військових цілях, тут КЗ автоматизує процес виявлення ворожих солдатів і транспортних засобів та керування ракетами. Найбільш досконалі системи керування ракетами відправляють ракету в задану область, замість конкретної цілі, а визначення цілей відбувається тоді, коли ракета досягає заданої області, базуючись на відеоданих, що надходять.

Найсучаснішою галуззю застосування комп'ютерного зору є автономні транспортні засоби: підводні, наземні (роботи, автомобілі), повітряні. Рівень автономності може варіюватися від систем, що лише допомагають водію (наприклад, адаптивний круїз-контроль), до повністю автономних транспортних засобів, які не потребують людського керування [9].

Комп'ютерний зір відіграє ключову роль у функціонуванні автономних транспортних засобів. Він використовується для:

- визначення місцезнаходження, створення карти навколишнього середовища, планування маршруту;
- розпізнавання інших транспортних засобів, пішоходів, велосипедистів, стаціонарних об'єктів;
- розуміння динаміки дорожнього руху, прогнозування поведінки інших учасників руху;
- вибір оптимальної траєкторії руху, контроль швидкості, гальмування.

Призначення робототехнічних систем підтримки рішень.

Після висвітлення можливостей комп'ютерного зору та розпізнавання образів, варто звернути увагу на сучасні тенденції в галузі робототехнічних систем підтримки та прийняття рішень (РСППР).

Робототехнічні системи підтримки та прийняття рішень використовуються для автоматизації складних процесів, що вимагають аналітичного мислення та прийняття рішень. Вони можуть бути використані в різних галузях, включаючи промисловість, медицину, науку та інше.

Розглянемо можливості РСППР:

- автоматизація рутинних завдань, що вимагають аналітичного мислення, звільняючи людей від монотонних робіт;
- надання корисної інформації та аналітичних інструментів, що допомагають приймати обгрунтовані рішення;
- завдяки автоматизації рутинних завдань та підтримки прийняття рішень, РСППР можуть значно покращити продуктивність;
- зменшення помилок, що часто виникають при виконанні складних завдань людьми;
- РСППР можуть бути налаштовані для виконання різних завдань в різних областях застосування;
- забезпечує високу продуктивність та ефективність, завдяки тому що такі системи можуть працювати безперервно;
- забезпечення безпеки, завдяки тому, що РСППР можуть виконувати завдання в небезпечних умовах, зменшуючи ризик для людей.

Враховуючи ці аспекти, РСППР можуть бути важливим інструментом для підвищення ефективності та продуктивності в різних галузях. Вони можуть допомогти людям краще використовувати свій час, зосереджуючись на більш важливих та творчих завданнях.

Сучасні робототехнічні системи підтримки рішень стикаються з рядом викликів і проблем, які потребують уваги та розв'язання для покращення їхньої ефективності та надійності. Наприклад, проблеми, з якими зіштовхуються такі системи:

- робототехнічні системи потребують точного та швидкого розпізнавання об'єктів, людей, перешкод та інших аспектів навколишнього середовища для прийняття рішень, але у реальному житті можуть виникати проблеми з недостатньою точністю розпізнавання, особливо в умовах зміни освітлення, шуму чи обмеженого доступу до даних;
- складнощі інтеграції робототехнічних систем з іншими інформаційними та технологічними системами, що може ускладнювати обмін даними та спільну роботу;

- використання робототехнічних систем у важливих сферах може ставити питання безпеки даних, конфіденційності, а також етичних аспектів використання таких систем, особливо в контексті автономних рішень;

- навчання робототехнічних систем може бути викликом, особливо коли мається на увазі адаптація до непередбачуваних сценаріїв або робота в нових середовищах;

- управління групою роботів може вимагати складних алгоритмів та механізмів координації для досягнення спільної мети без конфліктів чи перекриття завдань.

Для подолання складнощів, з якими стикаються сучасні робототехнічні системи підтримки рішень, необхідно застосовувати різноманітні підходи та стратегії. Нижче розглянемо деякі можливі способи вирішення цих проблем:

- продовження досліджень у галузі штучного інтелекту, машинного навчання та глибокого навчання для створення більш точних та ефективних алгоритмів розпізнавання образів та управління робототехнічними системами;

- розробка та впровадження розподілених систем, які дозволяють робототехнічним системам обмінюватися інформацією та координувати свою діяльність для ефективного вирішення завдань у реальному часі;

- використання новітніх технологій, таких як блокчейн та інтернет речей (IoT), для підвищення безпеки, ефективності та надійності робототехнічних систем;

- розробка більш ефективних та адаптивних систем навчання для робототехнічних систем, які дозволяють їм швидше адаптуватися до нових умов та сценаріїв роботи;

- врахування етичних аспектів у використанні робототехнічних систем, зокрема забезпечення конфіденційності даних, захисту приватності та відповідального використання цих систем у різних сферах діяльності;

- розробка інтерактивних інтерфейсів, які спрощують взаємодію людини з робототехнічними системами та полегшують процес прийняття рішень;

- розробка та впровадження методів та стратегій для ефективної співпраці між робототехнічними системами та людьми у різних сценаріях роботи.

В результаті, в даному дослідженні визначено ключову роль процесу розпізнавання образів і комп'ютерного зору в удосконаленні робототехнічних систем підтримки рішень. Розглянуто основні алгоритми та сфери застосування комп'ютерного зору, призначення РСППР, а також висвітлено сучасні проблеми, з якими вони стикаються.

Практична цінність полягає у демонструванні потенціалу впровадження РСППР для підвищення ефективності, продуктивності та безпеки в різних галузях.

Перспективами досліджень є: вдосконалення алгоритмів розпізнавання та управління, інтеграція новітніх технологій, покращення систем навчання, врахування етичних аспектів, розробка інтерактивних інтерфейсів для ефективної людино-машинної взаємодії.

ЛІТЕРАТУРА

1. Сотник С. В., Аналіз систем автоматизації визначення умов у житлових та робочих приміщеннях з використанням комп'ютерно-інтегрованих рішень / С. В. Сотник, Я. І. Халімонов // Automation, electronics and robotics (AERT-2023). – 2023. – P. 32-35.

2. Sotnik S. V. Analysis of design process of automated fire protection system / S. V. Sotnik, Y. R. Vasylychenko // Automation, electronics and robotics (AERT-2023). – 2023. – P. 59-62.

3. Sotnik S. V. Safe cobots in development of industrial robotics : дис. / S. V. Sotnik, Y. S. Usenko, P. V. Shakhov // The 8th International scientific and practical conference "European scientific congress" (September 4-6, 2023). – Barca Academy Publishing, Madrid, Spain. – 2023. – P. 80-84.

4. Сотник С. В. Огляд базових елементів автоматизованої системи контролю навколишнього середовища портативної ділянки зеленого побуту / С. В. Сотник, Ф. В. Кирпота // Автоматизація, електроніка та робототехніка (AERT-2023). – 2023. – Р. 80-84.
5. Sotnik S. V. Development of remote control for thermoplastics dosing automation system / S. V. Sotnik, V. V. Trokhin, D. O. Tereshchuk // The 5th International scientific and practical conference "Topical aspects of modern scientific research" (January 25-27, 2024) CPN Publishing Group, Tokyo, Japan. – 2024. – P. 179-184.
6. Sotnik S. V. Design features of control panels and consoles in automation systems / S. V. Sotnik, K. S. Redkin // 9th International scientific and practical conference "Science and innovation of modern world" (May 18-20, 2023) Cognum Publishing House, London, United Kingdom. – 2023. – P. 201-205.
7. Gomedé E. Deciphering patterns: the role of pattern recognition in computer vision. Medium / E. Gomedé // The Modern Scientist. – 2023.
8. Шеремет О. І. Метод опорних векторів (SVM) / О. І. Шеремет та інші. // Математичне моделювання. – 2013. – №. 1. – Р. 13-17.
9. Петришин В. С. Машинне навчання / В. С. Петришин та інші. // Комп'ютерні технології обробки даних. – 2022. – Р. 167-170.

Посилання

на

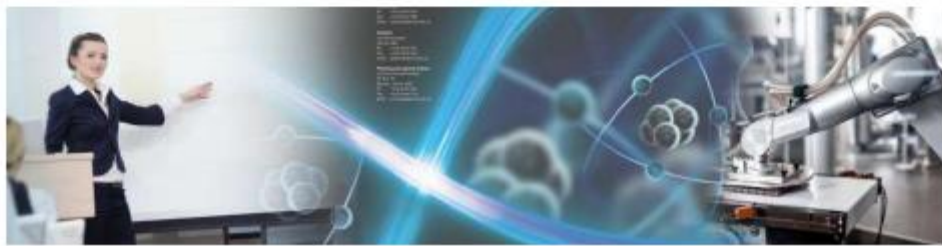
статтю:

<https://openarchive.nure.ua/server/api/core/bitstreams/4f761402-f21e-48b5-8f2b-f0df59491a86/content>

ДОДАТОК В
Апробація результатів роботи

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки
Кафедра комп'ютерно-інтегрованих технологій, автоматизації та робототехніки

**VIII Міжнародна Конференція
ВИРОБНИЦТВО
&
МЕХАТРОННІ СИСТЕМИ 2024**



**VIII International Conference
MANUFACTURING
&
MECHATRONIC SYSTEMS 2024**

M&MS
2024
VII International Conference
25-26 October
Kharkiv

M&MS 2024, 25-26 October, Kharkiv, Ukraine

УДК: 005:004.896:62-65:338.3

Виробництво & Мехатронні Системи 2024: матеріали VIII-ої Міжнародної конференції, Харків, 25-26 жовтня 2024 р.: тези доповідей / [редкол. І.Ш. Невлюдов (відповідальний редактор)].-Харків: [електронний друк], 2024. – 135 с.

У збірник включені тези доповідей, які присвячені сучасним тенденціям розвитку технологій та засобів виробництва та мехатронних систем, передовому досвіду та впровадженню їх в галузях систем промислової автоматизації та керування виробництвом; системній інженерії; CAD/CAM/CAE системах; мехатроніці (електро-механічних системах, електронних інструментах систем керування, механічних CAD системах); робототехніці та засобах інтелектуалізації; MEMS (сучасних матеріалів та технологіях виготовлення MEMS) та компонентах і технологіях автоматизації видобутку, переробки та транспортування нафти та газу.

Редакційна колегія: І.Ш. Невлюдов, В.В. Євсєєв.

Manufacturing & Mechatronic Systems 2024: Proceedings of VIII st International Conference, Kharkiv, October 25-26, 2024: Thesises of Reports / [Ed. I.Sh. Nevlyudov (chief editor).] - Kharkiv .: [electronic version], 2024. - 135 p.

The collection includes the thesises of reports on modern trends in the development of technologies and means of production and mechatronic systems, top experience and implementation of them in fields of: industrial automation and production management systems; systems engineering; CAD/CAM/CAE systems; mechatronics (electrical and mechanical systems, electronic control tools, mechanical CAD systems); robotics and intellectual tools; MEMS (modern materials and manufacturing technologies MEMS) and components and technologies for the automation of oil, gas and oil extraction, processing and transportation.

Editorial board: Igor.Sh. Nevludov, Vladyslav.V. Yevsieiev

© Кафедра комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР), ХНУРЕ, 2024

M&MS 2024, 25-26 October, Kharkiv, Ukraine

Міністерство освіти і науки України (МОНУ)
Харківський національний університет радіоелектроніки (ХНУРЕ)
Варшавський університет сільського господарства (WULS - SGGW)
Азербайджанський державний університет нафти і промисловості
Національний університет «Львівська політехніка»
Festo Didactic Україна
Jabil Circuit Ukraine Limited
ТОВ «Науково-виробниче підприємство «УКРІНТЕХ»»
Факультет автоматичних і комп'ютеризованих технологій (АКТ)
Кафедра комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР),
Державне підприємство «Харківський науково-дослідний інститут технологій
машинобудування»
Державне підприємство «Південний державний проектно-конструкторський та
науково-дослідний інститут авіаційної промисловості»

МАТЕРІАЛИ

VIII-ої Міжнародної Конференції

ВИРОБНИЦТВО & МЕХАТРОННІ СИСТЕМИ 2024

**(25-26 жовтня 2024)
Харків, Україна**

ОРГАНІЗАТОРИ



Міністерство
освіти і науки
України

Міністерство освіти і науки України (МОНУ)
The Ministry of Education and Science of Ukraine



NURE
Kharkiv National University
of Radioelectronics

Харківський національний університет
радіоелектроніки (ХНУРЕ)

Kharkiv National University of Radioelectronics



**WARSAW UNIVERSITY
OF LIFE SCIENCES
- SGGW**

Варшавський університет сільського
господарства (WULS - SGGW)

Warsaw University of Life Sciences WULS - SGGW



Азербайджанський державний університет
нафти і промисловості

Azerbaijan State Oil and Industry University



Festo Didactic Україна

Festo Didactic Ukraine



ТОВ «Науково-виробниче підприємство
«УКРІНТЕХ»»

Research and Production Enterprise
"UKRINTECH" Ltd



Національний університет «Львівська
політехніка»

National University Lviv Polytechnic

Державне підприємство «Харківський науково-
дослідний інститут технології машинобудуван-
ня», м. Харків, Україна

State Enterprise «Kharkiv Scientific-Research
Institute of Mechanical Engineering Technology»,
Kharkiv, Ukraine



Державне підприємство «Південний державний
проектно-конструкторський та науково-
дослідний інститут авіаційної промисловості»,
м. Харків, Україна

State Enterprise «National Design & Research
Institute of Aerospace Industries», Kharkiv,
Ukraine



Jabil Circuit Ukraine Limited

КОМІТЕТ КОНФЕРЕНЦІЇ

МІЖНАРОДНИЙ ПРОГРАМНИЙ КОМІТЕТ КОНФЕРЕНЦІЇ

- Ігор Шакирович Невлюдов** голова комітету конференції, доктор технічних наук, професор, заслужений діяч науки і техніки України, лауреат Державної премії в галузі науки і техніки України; лауреат Державної премії України в галузі освіти, завідувач кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР), Харківського національного університету радіоелектроніки, Україна
- Олександр Іванович Филіпенко** заступник голови комітету конференції, доктор технічних наук, професор, лауреат Державної премії України в галузі освіти, декан факультету Автоматики і комп'ютеризованих технологій (АКТ), Харківського національного університету радіоелектроніки, Україна.
- Мурад Анвер огли Омаров** доктор технічних наук, професор, заслужений діяч науки Азербайджанської Республіки проректор з міжнародного співробітництва, Харківський національний університет радіоелектроніки, Україна
- Владислав В'ячеславович Свєтєв** секретар, доктор технічних наук, професор, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР), Харківського національного університету радіоелектроніки, Україна.
- Andrzej Chochowski** доктор технічних наук, професор Варшавського університету сільського господарства (WULS - SGGW), Польща
- Pawel Obstawski** доктор технічних наук, професор Варшавського університету сільського господарства (WULS - SGGW), Польща.
- Сергій Богомолів** лектор/доцент, доктор філософії (комп'ютерні науки), Дослідницька школа комп'ютерних наук, Коледж інженерії та комп'ютерних наук, Австралійський національний університет, Австралія.
- Микола Васильович Замірець** доктор технічних наук, професор, заслужений діяч науки і техніки України, лауреат Державної премії України в галузі науки і техніки, директор Державного підприємства Науково-дослідного технологічного інституту приладобудування, Україна
- Михайло Васильович Лобур** доктор технічних наук, професор, заслужений діяч науки і техніки України, відмінник народної освіти України, завідувач кафедри систем автоматизованого проектування Національного університету «Львівська політехніка», Україна.
- Євген Сергійович Риженко** керівник відділу дидактики ДП «Фесто», Україна

- Сергій Володимирович Демченко** директор ТОВ «Науково-виробничого підприємства «УКРІНТЕХ», Україна.
- Самед Імамалі огли Юсіфов** кандидат технічних наук, доцент, декан факультету інформаційних технологій та управління, Азербайджанський державний університет нафти і промисловості, Азербайджан.
- Фарід Гаджі огли Агаєв** кандидат технічних наук, доцент, завідувач кафедри управління та системної інженерії, Азербайджанський державний університет нафти і промисловості, Азербайджан.
- Віктор Васильович Косенко** доктор технічних наук, професор, професор кафедри автоматки, електроніки та телекомунікацій, Національний університет «Полтавська політехніка імені Юрія Кондратюка» Україна.
- Володимир Вікторович Козирський** доктор технічних наук, професор, заслужений діяч науки і техніки України, директор Навчально-наукового інституту енергетики, автоматки та енергозбереження, Національний університет біоресурсів і природокористування України, Україна.
- Віталій Пилипович Лисенко** доктор технічних наук, професор, заслужений працівник освіти України, завідувач кафедри автоматки та робототехнічних систем ім. акад. І.І. Мартиненка, Національний університет біоресурсів і природокористування України, Україна.
- Юрій Францевич Зінковський** доктор технічних наук, лауреат Державної премії України в галузі освіти, професор кафедри радіоконструювання і виробництва радіоапаратури, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Україна.
- Володимир Митрофанович Свищ** доктор технічних наук, професор, заслужений діяч науки і техніки України, професор кафедри систем управління літальними апаратами, Національний аерокосмічний університет ім. М. С. Жуковського "Харківський авіаційний інститут", Україна.
- Віталій Євгенович Овчаренко** доктор технічних наук, професор, заступник директора з наукової роботи Державного підприємства «Науково-дослідний технологічний інститут приладобудування», Україна.
- Лариса Сергіївна Глоба** доктор технічних наук, професор, лауреат Державної премії Кабінету Міністрів України, завідувач кафедри інформаційно-комунікаційних мереж, Інститут телекомунікаційних систем Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського», Україна.
- Анатолій Олександрович Андрусевич** доктор технічних наук, професор, начальник Криворізького коледжу Національного авіаційного університету, Україна.

- Роман Володимирович Артюх** кандидат технічних наук, директор Державного підприємства «Південний державний проектно-конструкторський інститут авіаційної промисловості», Україна.
- Kurtwitz** генеральний менеджер Titan Machinery Limited, Шотландія.
- Liu Shan** генеральний менеджер Titan Machinery Limited, Китай.
- Володимир Андрійович Павлиш** кандидат технічних наук, професор, заслужений діяч науки і техніки України», перший проректор Національного університету «Львівська політехніка», Україна
- Сергій Іванович Осадчий** доктор технічних наук, професор, в.о. завідувача кафедри конструкції повітряних суден, авіадвигунів та підтримання льотної придатності, Льотна академія НАУ, м.Кропивницький, Україна.
- Анатолій Афанасійович Єфіменко** доктор технічних наук, професор, завідувач кафедри електронних засобів та інформаційно-комп'ютерних технологій, Одеський національний політехнічний університет, Україна
- Володимир Михайлович Решетюк** кандидат технічних наук, доцент кафедри автоматичної та робототехнічних систем ім. акад. І.І. Мартиненка, Національний університет біоресурсів і природокористування України, Україна.

ОРГАНІЗАЦІЙНИЙ КОМІТЕТ КОНФЕРЕНЦІЇ

- Олександр Михайлович Цимбал** заступник голови конференції з організаційних питань, доктор технічних наук, професор комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР), Харківський національний університет радіоелектроніки, Україна.
- Сергій Павлович Новоселов** кандидат технічних наук, доцент, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР), Харківський національний університет радіоелектроніки, Україна.
- Євген Анатолійович Разумов-Фризюк** кандидат технічних наук, доцент, доцент кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР), Харківський національний університет радіоелектроніки, Україна.
- Наталія Павлівна Демська** кандидат технічних наук, доцент, доцент кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР), Харківський національний університет радіоелектроніки, Україна.

ЗМІСТ

<i>Svitlana Alyokhina</i>	
System Approach to the Positive Energy District Analysis	12
<i>Dmytro Gurin</i>	
Розробка динамічного представлення параметрів моделі опису навколишнього середовища колаборативного робота	15
<i>Artem Hubar</i>	
Automation of Power Grid Element Management to Enhance Energy Efficiency	19
<i>Артем Бронніков, Стеценко Катерина</i>	
Автономний робот на Raspberry Pi з аналізом облич та емоцій в реальному часі	22
<i>Andrii Lvov, Svetlana Sotnik</i>	
Analysis of electronic locks existing systems	24
<i>Artem Tverdokhlib, Svetlana Sotnik</i>	
Intelligent tools for optimizing information and search engines	28
<i>Igor Zarubin, Svetlana Sotnik</i>	
Basic principles of building aerial robots	32
<i>Pavlo Sukhno, Svetlana Sotnik</i>	
Critical review of GSM network structure	37
<i>Oleksii Shevchenko, Nataliia Furmanova, Vadim Yakovenko, Yaroslav Lukash</i>	
Assessment of the quality of brushless DC motors	42
<i>Artem Zhulai, Nataliia Furmanova</i>	
System for monitoring and alerting in a coal mine	45
<i>Сніжана Вичужаніна, Олександр Малий</i>	
Огляд щодо використання радіоаматорами радіочастотного спектру в Україні	48

<i>Воронов Денис, Сезонова Ірина</i>	
Розробка методу визначення швидкості переміщення об'єктів на основі аналізу зображень	51
<i>Oleh Hurtovyi</i>	
Features of Functional Testing for Low-Power Consumption Devices with Built-In Batteries	55
<i>Варвара Карташова, Артем Бронніков</i>	
Роль експертних систем та голосового керування в сучасному виробництві	58
<i>Антон Паньков</i>	
Інноваційний підхід до візуалізації: розробка автоматизованого модуля для збору, обробки та збереження поточних даних	62
<i>Олег Посашков, Олександр Цимбал</i>	
Аналіз існуючих методів підтримки прийняття рішень у віддаленому управлінні виробництвом	65
<i>Дмитро Максимов, Дмитро Нікітін</i>	
Види зварювання для верстату точкового зварювання з ЧПУ	69
<i>Олексій Фарафонов, Наталія Фурманова, Олександр Малий</i>	
Розроблення технології паралельного керування за допомогою вебінтерфейсу мобільним роботом під керуванням ROS	71
<i>Дмитро Янушкевич, Леонід Іванов, Ігор Толкунов</i>	
Застосування методів вербального аналізу в інтелектуальних системах управління у сфері гуманітарного розмінування	75
<i>Данило Ясир</i>	
Вибір математичної моделі для управління якістю продукції в умовах безперервного виробництва	79
<i>Дмитро Дриньов</i>	
Використання елементів штучного інтелекту для вирішення задач моделювання динамічних процесів	83
<i>Ганна Самоїленко</i>	
Дослідження методів опису динаміки гуманоїдного робота	85

<i>Андрій Слюсар, Софія Хрустальова</i>	
Методи та алгоритми локалізації RFID-міток: сучасні підходи та перспективи	87
<i>Василь Туз, Володимир Чумаков, Олександр Філіпенко, Оксана Сичова</i>	
Дослідження дисперсійних характеристик мікроструктурованого оптичного волокна в умовах деформації	92
<i>Тимур Лихо, Світлана Максимова</i>	
Основні етапи розроблення наземного мобільного робота	96
<i>Vladyslav Yevsieiv</i>	
Using the Dempster-Shafer Theory in Data Fusion Solutions for Collaborative Robotic Manipulators within Industry 5.0	99
<i>Vladyslav Yevsieiv, Nataliia Demska</i>	
A Model of Using Computer Vision to Monitor the Environment of a Collaborative Manipulator Robot	102
<i>Віталій Тетеря, Світлана Максимова</i>	
Розробка системи ідентифікації, розпізнавання та трекінгу для колаборативного робота	105
<i>Vladyslav Yevsieiv, Svetlana Starikova</i>	
Using the Triangulation Method to Measure the Distance to Objects in the Working Area of a Collaborative Manipulator Robot	107
<i>I.V. Жарікова, Д.О. Нікітін</i>	
Дослідження механічних параметрів гнучких комутаційних структур для мобільних роботизованих платформ	110
<i>Svetlana Starikova, Illya Karpenko</i>	
Development of a Structural Control Scheme for a Small-sized Mobile Robot for Investigating Damaged Buildings	114
<i>Максим Вжесневський</i>	
Інтелектуальне керування автономними транспортними шатлами для внутрішньо-складських логістичних систем	117

<i>Родіо Клименко, Дмитро Кухаренко</i>	
Програмне забезпечення для розрахунку резонансних частот мембран живих організмів	120
<i>Микола Мешков, Дмитро Кухаренко</i>	
Алгоритм та програмна реалізація роботи комплексу очних м'язів людини	124
<i>Дмитро Кухаренко, Олексій Юрко, Денис Тимченко</i>	
Автоматизований аналіз довільних ділянок фонокардіограм в середовищі Labview	128
<i>Сергій Новоселов, Владислав Іванов</i>	
Вирішення задачі управління багатоланковим маніпулятором	132

Основні етапи розроблення наземного мобільного робота

Тимур Лихо¹, Світлана Максимова²

1. Кафедра КІТАР, Харківський національний університет радіоелектроніки, УКРАЇНА, Лоботин, вул. Швідта, 9, e-mail: tymur.lykko@nure.ua
2. Кафедра КІТАР, Харківський національний університет радіоелектроніки, УКРАЇНА, Харків, просп. Науки, 14, e-mail: svilana.milyutina@nure.ua

Анотація: У статті розглянуто ключові етапи розробки наземного мобільного робота, що включають визначення функцій та завдань робота, вибір апаратної платформи, програмування, тестування та оптимізацію системи. У результаті роботи буде визначено основні підходи до створення роботів, їх тестування і оптимізації для різних середовищ та завдань.

Ключові слова: мобільний робот, робототехніка, роботизована система

I. Актуальність роботи.

У наші часи мобільні роботи відіграють важливу роль в автоматизації повсякденних, однотипних завдань, що дозволяє значно підвищити ефективність у таких сферах, як логістика, виробництво, аграрний сектор, військова справа та ін. Роботи можуть виконувати складні або небезпечні, для людини завдання, що мінімізує ризики та збільшує продуктивність.

Для розроблення наземного мобільного робота, необхідно пройти певні етапи створення робота, основні з них буде висвітлено у даній роботі.

II. Матеріал і результати досліджень.

Розроблення наземного робота це тривалий та кропіткий процес, що потребує знань у різних науках, від математики та фізики до біології та анатомії, але у загальному для цього необхідно пройти 4 етапи, розглянемо кожен з них.

1. Визначення функцій та завдання робота.

На даному етапі розробнику необхідно чітко визначити основні функції та завдання робота, оскільки це впливатиме на подальший хід розроблення [1].

Перш за все, необхідно з'ясувати, яке завдання має виконувати робот. Це може бути моніторинг (рис. 1.а), патрулювання об'єктів (рис. 1.б), виконання доставок (рис. 1.в).



в

Рисунок 1 – Видляд мобільних роботів: а - робот моніторингу забруднення «Rasmus» [2], б - охоронний патрульний робот «S5.2» [3], в - робот-кур'єр «Camello» [4]

Також варто врахувати робоче середовище у якому буде функціонувати робот. Наприклад, мобільний робот, що працює в умовах закритого приміщення, потребує іншого підходу до навігації, ніж робот, який функціонує на відкритих просторах або у складних природних умовах. Для роботів, які використовуються в логістичних центрах або на складах, важливо забезпечити точність переміщення і здатність працювати в обмежених просторах, тоді як роботи для зовнішніх середовищ мають бути стійкими до кліматичних умов і перепадів рельєфу.

Не менш важливо вирішити яким чином робот буде пересуватись. Наприклад, для роботи в приміщеннях найчастіше використовуються колісні роботи (рис. 1) через їх простоту, високу швидкість та маневреність на рівних поверхнях. У той час, гусеничні роботи (рис. 2.а) підходять для роботи в більш складних умовах, де потрібно долати перешкоди або працювати на нерівних поверхнях. Для роботи в особливо складних середовищах можуть бути застосовані крокуючі роботи (рис. 2.б), які можуть переміщуватися навіть по дуже складній місцевості.



а



6

Рисунок 2 – Вигід мобільних роботів: а – гусеничний робот від компанії RoboTech Vision [5], б – крокуючий робот «Spot» від BostonDynamics [6]

2. Вибір апаратної платформи.

Необхідно визначити, яка апаратна платформа буде використовуватися як основний "мозок" робота. Вибір залежить від вимог до обчислювальних ресурсів, типу завдань, які виконуватиме робот, та його інтерактивності із зовнішнім середовищем. Популярними рішеннями є мікроконтролери ESP32 або Raspberry Pi.

ESP32 відомий своєю низькою енергоспоживаністю та інтеграцією з Wi-Fi та Bluetooth, що робить його чудовим вибором для мобільних роботів із завданням зв'язку та віддаленого контролю.

Raspberry Pi, у свою чергу, забезпечує вищі обчислювальні потужності, що дозволяє використовувати більш складні алгоритми, наприклад, комп'ютерного зору чи нейронних мереж.

Також важливо, забезпечити достатній рівень енергоживлення, обрати тип батарей або акумуляторів, які дозволять роботу функціонувати протягом тривалого часу без перерв. Найпоширенішими варіантами є літій-іонні або літій-полімерні акумулятори завдяки їх високій щільності енергії та компактним розмірам.

Сенсори та система навігації відіграють ключову роль у орієнтуванні робота у навколишньому середовищі. Тут розробники мають обрати відповідні сенсори, зокрема ультразвукові та інфрачервоні датчики, які забезпечують можливість обробки інформації про відстань до об'єктів. Гіроскопи та акселерометри допомагають у навігації та стабілізації, що є критичним для мобільних роботів, які працюють у динамічних умовах. Для того щоб робот міг переміщатися автономно, важливо інтегрувати систему навігації. Це може бути використання GPS для зовнішнього середовища або камера для внутрішнього простору. Навігаційні системи дають роботу можливість не лише слідувати за своєю траєкторією, але й ухилятися від перешкод, взаємодіяти з іншими роботами чи об'єктами навколишнього середовища.

3. Програмне забезпечення

На цьому етапі розробнику необхідно, для початку визначитись з мовою програмування. Найпопулярніші мови програмування для робототехніки [7] наведено у таблиці 1.

Різні мови програмування можна комбінувати між собою в одній системі керування роботизованою системою, наприклад:

- C/C++ використовується для програмування роботи мобільного робота;
- Python необхідний для створення користувацького інтерфейсу для управління роботом.

ТАБЛИЦЯ 1. Мови програмування для робототехніки

Мова програмування	Особливості
C/C++	Є розвиненими мовами програмування загального призначення та забезпечують продуктивність у реальному часі, тобто запрограмовані дії виконуються у визначений час, що важливо в робототехніці
Python	Одна з найпростіших для вивчення мов програмування, що скорочує час програмування, усуваючи потребу визначати та перетворювати типи змінних і дозволяючи кодувати рухи в одному сценарії
Java	Популярна мова серед розробників, які працюють у сфері штучного інтелекту, оскільки можна створювати нейронні мережі, що дозволяє створювати людиноподібних роботів.
C#	Часто використовується для розробки мережевих і комунікаційних структур, таких як порти та сокети, що робить його хорошим вибором для інтеграції систем нейронної мережі в роботизовані програми.

4. Тестування, налагодження та оптимізація системи.

На цьому етапі перед розробниками ставиться задача перевірити функціонування датчиків та систем навігації, та їх калібрування, за необхідністю. Важливо перевірити роботу двигунів, що дозволяють роботу пересуватись у просторі. Також важливо відлагодити мережеві функції, якщо керування роботом потребує участі людини.

Часто для забезпечення довготривалої роботи мобільного робота також необхідно оптимізувати енергоспоживання, це можна зробити за допомогою встановлення режимів сну для робота, при активації якого мобільний робот буде призупиняти виконання менш важливих завдань на певний час.

Не менш важливим, на даному етапі, є перевірка робота у реальних умовах його роботи. Під час тестування оцінюється, наскільки ефективно він виконує поставлені завдання, як він реагує на різні зовнішні умови та перешкоди.

За результатами тестувань можуть вноситися зміни в апаратну частину або вдосконалюватися алгоритми.

III. Висновки

1. Основні результати досліджень:
 - визначено ключові етапи розробки мобільних роботів, які включають визначення функцій робота,

вибір апаратної платформи, програмування, тестування та оптимізацію системи;

- показано, як різні типи мобільних роботів (колісні, гусеничні, крокуючі) можуть бути адаптовані під специфічні умови експлуатації;

- описано переваги та недоліки використання популярних платформ для мобільних роботів, таких як ESP32 та Raspberry Pi, в залежності від завдань і ресурсів;

- наведено різні підходи до навігації та використання сенсорів для забезпечення автономності роботи робота.

2. Практична цінність:

- розробка наземних мобільних роботів має велику практичну цінність для багатьох галузей, таких як логістика, охорона, сільське господарство, виробництво тощо;

- роботи можуть виконувати завдання, які важко або небезпечно виконувати людині, підвищуючи ефективність та безпеку;

- дослідження продемонструвало, як сучасні технології можуть бути інтегровані для створення роботів з високою маневреністю, автономністю та здатністю до виконання складних завдань у різних середовищах.

3. Подальші перспективи:

- подальші дослідження можуть бути спрямовані на покращення автономності роботів за рахунок інтеграції штучного інтелекту та машинного навчання для кращого розпізнавання середовища та оптимізації маршрутів;

- розвиток нових енергоефективних рішень і батарей сприятиме збільшенню часу автономної роботи мобільних роботів.

Література

- [1] Kshitij Tiwari. 5 steps to getting started with mobile robots | kshitij tiwari ph.d. Kshitij Tiwari, Ph.D. URL: <https://kshitijtiwari.com/all-resources/mobile-robots/getting-started-with-mobile-robots/> (date of access: 26.09.2024).
- [2] International conference on intelligent robots and systems. 2008 IEEE/RSJ International conference on intelligent robots and systems (IROS), Nice, 22–26 September 2008. 2022. P. 2210–2211.
- [3] Security & Inspections robots for Industry leaders Turnkey robotic solutions Raas. URL: https://smprobotics.com/wp-content/uploads/2015/10/security_patrol_robot_s52.png (дата звернення: 27.09.2024).
- [4] Autonomus delivery robot "Camello". OTSAW | Accelerating Robotics Globally. URL: <https://otsaw.com/wp-content/uploads/2023/04/camello-cutout-min.png> (date of access: 28.09.2024).
- [5] The Crawler mobile robot. RoboTech Vision | Development of autonomous robots with AI. URL: <https://robotechvision.com/wp-content/uploads/2021/09/farming-mobile-robot-vineyard-crawler-robotechvision-blog.jpg> (date of access: 28.09.2024).
- [6] Spot® robot. Home | Boston Dynamics. URL: <https://bostondynamics.com/wp-content/uploads/2023/05/spot-explorer-web-2-2048x1152.jpg> (date of access: 28.09.2024).
- [7] Biba J. Top 8 robotic programming languages. Built In. URL: <https://builtin.com/robotics/robotic-programming-language> (date of access: 30.09.2024).
- [8] Abu-Jassar AT, Attar H, Amer A, et al. Development and Investigation of Vision System for a Small-Sized Mobile Humanoid Robot in a Smart Environment. International Journal of Crowd Science, 2024, <https://doi.org/10.26599/IJCS.2023.9100018>.
- [9] Yevsiciev, V., Maksymova, S., & Alkhalailah, A. (2024). Improvement of SUSAN Image Filtering Method for PCB Quality Inspection. *Journal of Universal Science Research*, 2(7), 106–116.
- [10] Gurin, D., Yevsiciev, V., Maksymova, S., & Alkhalailah, A. (2024). Using Convolutional Neural Networks to Analyze and Detect Key Points of Objects in Image. *Multidisciplinary Journal of Science and Technology*, 4(9), 5-15.
- [11] Gurin, D., Yevsiciev, V., Abu-Jassar, A., & Maksymova, S. (2024). Using the Kalman Filter to Represent Probabilistic Models for Determining the Location of a Person in Collaborative Robot Working Area. *Multidisciplinary Journal of Science and Technology*, 4(8), 66-75.
- [12] Gurin, D., Yevsiciev, V., Maksymova, S., & Abu-Jassar, A. (2024). Effect of Frame Processing Frequency on Object Identification Using MobileNetV2 Neural Network for a Mobile Robot. *Multidisciplinary Journal of Science and Technology*, 4(8), 36-44.
- [13] Gurin, D., Yevsiciev, V., Maksymova, S., & Alkhalailah, A. (2024). MobileNetV2 Neural Network Model for Human Recognition and Identification in the Working Area of a Collaborative Robot. *Multidisciplinary Journal of Science and Technology*, 4(8), 5-12.
- [14] Maksymova, S., Yevsiciev, V., Nevludov, I., & Uluhan, N. (2024). CONSTRUCTING AN OPTIMAL ROUTE FOR A MOBILE ROBOT USING A WAVE ALGORITHM. *Journal of Natural Sciences and Technologies*, 3(1), 282-289.
- [15] Abu-Jassar, A., Yevsiciev, V., & Maksymova, S. (2024). The Optical Flow Method and Graham's Algorithm Implementation Features for Searching for the Object Contour in the Mobile Robot's Workspace.
- [16] Yevsiciev, V., & Gurin, D. (2023). *Comparative Analysis of the Characteristics of Mobile Robots and Collaboration Robots Within INDUSTRY 5.0* (Doctoral dissertation, European Scientific Platform).
- [17] Yevsiciev, V., Maksymova, S., & Starodubcev, N. (2022). A robotic prosthetic a control system and a structural diagram development. *Collection of scientific papers «ΑΙΓΙΟΣ»*, (August 12, 2022; Zurich, Switzerland), 113-114.

M&MS 2024, 25-26 October, Kharkiv, Ukraine

Посилання на матеріали конференції: https://tapr.nure.ua/wp-content/uploads/2024/10/zbirnik_mms2024_work_compressed.pdf

ДОДАТОК Г
Апробація результатів роботи

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки
кафедра комп'ютерно-інтегрованих технологій, автоматизації та робототехніки
(КІТАР)



МАТЕРІАЛИ

I Всеукраїнської конференції
«Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки»
(Computer-integrated technologies, automation and robotics)

CITAR'24
16-17 травня 2024

[електронне видання]

Харків 2024

УДК: 005:004.896:62-65:338.3

Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки 2024: матеріали I-ої Всеукраїнської конференції, Харків, 16-17 травня 2024.: тези доповідей / [редкол. І.Ш. Невлюдов (відповідальний редактор)].-Харків: [електронний друк], 2024. – 163 с.

У збірник включені тези доповідей, які присвячені сучасним автоматизованим технологіям Industry 4.0 та їх впровадження; інформаційні управляючі системи технологічного призначення; математичні методи в системах автоматизації; розробка та програмування в робототехніці; штучний інтелект та машинне навчання в автоматизації; інтеграція технологій у виробництві та промисловості; сенсорні технології та взаємодія людини з роботами в Industry 5.0; ефективність використання роботизованих систем у виробництві; етика та правові аспекти в робототехніці; Інтернет речей та Інтегровані системи в комп'ютерно-інтегрованих технологіях, автоматизації та робототехніки; технологічні виклики та інновації у світі робототехніки.

Редакційна колегія: І.Ш. Невлюдов, В.В. Євсєєв.

Computer-integrated technologies, automation and robotics 2024: Proceedings of I st All-Ukrainian Conference, Kharkiv, May 16-17, 2024: Theses of Reports / [Ed. I.Sh. Nevludov (chief editor).] - Kharkiv .: [electronic version], 2024. - 163 p.

The collection includes abstracts devoted to modern automated technologies of Industry 4.0 and their implementation; information control systems for technological purposes; mathematical methods in automation systems; development and programming in robotics; artificial intelligence and machine learning in automation; integration of technologies in production and industry; sensor technologies and human interaction with robots in Industry 5.0; efficiency of using robotic systems in production; ethics and legal aspects in robotics; Internet of Things.

Editorial board: Igor.Sh. Nevludov, Vladyslav.V. Yevsieiev

© Кафедра комп'ютерно-інтегрованих технологій, автоматизації та робототехніки (КІТАР), ХНУРЕ, 2024

КОМІТЕТ КОНФЕРЕНЦІЇ

Голова комітету конференції	Невлюдов Ігор Шакирович , доктор технічних наук, професор, завідувач кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.
Секретар конференції	Євсєєв Владислав В'ячеславович , доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.
Технічний секретар	Самойленко Ганна Юріївна , асистент кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.
Редакційна колегія:	<p>Филипенко Олександр Іванович, доктор технічних наук, професор, декан факультету Автоматики та комп'ютеризованих технологій, Харківського національного університету радіоелектроніки.</p> <p>Цимбал Олександр Михайлович, доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.</p> <p>Ромашов Юрій Володимирович доктор технічних наук, професор кафедри комп'ютерно-інтегрованих технологій, автоматизації та робототехніки, Харківського національного університету радіоелектроніки.</p> <p>Косенко Віктор Васильович, доктор технічних наук, професор, зам. директора Державного підприємства «Південний державний проектно-конструкторський та науково-дослідний інститут авіаційної промисловості».</p> <p>Замірець Микола Васильович, доктор технічних наук, професор, директор Державного підприємства Науково-дослідного технологічного інституту приладобудування.</p> <p>Свищ Володимир Митрофанович, доктор технічних наук, професор, радник директора Державне науково-виробниче підприємство «Об'єднання Комунар».</p> <p>Кухаренко Дмитро Володимирович, кандидат технічних наук, доцент кафедри «Електронних апаратів» Кременчуцького національного університету імені Михайла Остроградського</p> <p>Фурманова Наталія Іванівна, кандидат технічних наук, доцент, в.о. декана факультета Радіоелектроніки і телекомунікацій, Національного університету «Запорізька політехніка».</p>

ЗМІСТ

<i>М. О. Вжесневський, О.О. Чала, Ю.В Ромашов</i> Розробка кінематичної схеми транспортувального шатлу для внутрішньоскладської виробничої логістики	6
<i>Н. Р. Курбанов</i> Перспективи розитку систем дистанційного керування роботами розмінувальниками ...	10
<i>К. С. Німець</i> Проблеми та перспективи використання систем комп'ютерного зору у робототехніці ...	14
<i>Г. Ю.Самойленко</i> Методи синхронного управління групою мобільних роботів	17
<i>Svitlana Starikova</i> Comparison of the Laws of Robotics By Isaac Asimov and Beam Robotics.....	21
<i>Vladyslav Yevsieiev</i> Comparative Analysis of Modifications of Rrt Algorithms for Route Planning of a Mobile Robot	25
<i>О. М. Клименко</i> Аналіз методів управління автономною робототехнічною транспортною системою фармацевтичного виробництва	29
<i>В.С. Натарова, О.О. Чала</i> Автоматизація гідропонного вирощування	32
<i>N. Furmanova, O. Farafonov, S. Malyi</i> Automated Reverse Engineered of Printed Circuit Boards	37
<i>O. Malyi, I. Pospieieva, V. Miroshnichenko</i> Creating Methodology Of Pre-Project Selection of Components for Multi-Rotor UAVs	41
<i>Андреев А. С.</i> Штучний інтелект та машинне навчання в автоматизації	45
<i>I. Zaitcev, O. Vasylenko</i> Plant Watering and Lighting Control System for Home and Small Businesses	50
<i>Ф. Курнога</i> Технології у виробництві пристроїв для зеленого обіходу	53
<i>Mykhailo Dovbnya, Dmytro Kukhareno</i> Synthesis of the Electric Diagram of the Laboratory Power Supply Unit for Experiments in Educational Institutions	57
<i>Mykhailo Dovbnya, Dmytro Kukhareno</i> Comparative Analysis of Laboratory Power Units for Experiments in Educational Institutions	61
<i>Т.А. Луко</i> Розроблення Веб-сторінки керування мобільним роботом через протокол MQTT	66
<i>Д. Ю. Філіппенков, М. Ю. Тягунова</i> Розробка автоматизованої системи тролейбусного парку	72
<i>Я.І. Халімонов</i> Забезпечення оптимальних умов на виробничих майданчиках за допомогою сенсорних технологій	74
<i>V. Onyshchenko, O. Shevchenko, P. Kostianoi</i> Development of A Video Stream Transmission System In Digital Form for FPV UAVs	78

<i>Дідик П.Ю., Максимова С.С.</i>	
Оцінка якості кластеризації координат географічних об'єктів різними методами	82
<i>М.Г. Стародубцев, Д.П. Власенков, С.В. Шибанов</i>	
Оперативне управління технологічним процесом виробництва в умовах невизначеності	87
<i>Г.С. Макаренко, Ю.Ю. Мірошник, М.Ю. Білоусов</i>	
Концепція комплексного підходу при проектуванні радіоелектронних засобів з урахуванням вимог електромагнітної сумісності	92
<i>М.Г. Стародубцев, Д.П. Власенков, К.О. Харченко</i>	
Типологія моделей управління технологічними процесами за умов невизначеності	97
<i>Б.С. Місан, І. Ш. Невлюдов, О.А. Рубан</i>	
Перспективи 3D-друку плівок оральних	103
<i>В.І. Фомін</i>	
Робототехнічні системи з елементами штучного інтелекту у виробництві	106
<i>Р.Р. Шаталюк</i>	
Взаємодія людини з колаборативними роботами в Індустрії 5.0	109
<i>А. О. Вайнштейн</i>	
Розробка комунікаційного сервісу універсальної персональної мобільної лабораторії ..	113
<i>Гурін Д.В.</i>	
Розробка структурної схеми підключення роботизованої платформи для розмінування	115
<i>Завалюєв А.О., Гурін Д.В.</i>	
Ультрафіолетові камери. Види, особливості використання	119
<i>Корнієнко А.О., Гурін Д.В.</i>	
Система позиціонування сонячних панелей на базі Arduino Uno	122
<i>Mykola Khranovskyi, Andriy Kernytskyi</i>	
Advantages of Using Zero-Knowledge Proof in Biometric Systems	126
<i>П.М. Савченко</i>	
Сенсори позиціонування в робототехніці в умовах аварійного відключення електроживлення	130
<i>A.S. Karpenko, O.O. Chala</i>	
Constructional automation in Industry 5.0	133
<i>Ю. Л. Гасюк, М. Р. Мельник, А. В. Гасюк</i>	
Математична модель розрахунку оптимальних параметрів нагріву води в системах розумного будинку	137
<i>Ігор Голод</i>	
Методи вимірювання показників мікроклімату приміщень для виробництва технічних засобів автоматизації	141
<i>В. В. Запорізький</i>	
Методи детектування зіткнень колаборативними роботами	146
<i>І.С. Зарубін, С.В. Сотник</i>	
Ефективність використання роботизованих систем у виробництві	150
<i>Г. С. Макаренко</i>	
Математичні методи забезпечення автоматизованої обробки даних температурної залежності теплофізичних характеристик матеріалів	154

РОЗРОБЛЕННЯ ВЕБ-СТОРІНКИ КЕРУВАННЯ МОБІЛЬНИМ РОБОТОМ ЧЕРЕЗ ПРОТОКОЛ MQTT

Т.А. Лихо

Харківський національний університет радіоелектроніки

Україна, 62433, Люботин, вул. Шмідта, 9

E-mail: tymur.lykho@nure.ua

Анотація: У даній статті розглянуто основні аспекти створення веб-інтерфейсу для керування роботизованою платформою відеонагляду на базі ESP32-CAM. В результаті роботи будемо проведемо тестування взаємозв'язку робота та інтерфейсу користувача через протокол MQTT.

Ключові слова: Мобільний Робот, Відеонагляд, Веб-Сторінка, Інтерфейс, Протокол Mqtt, Інновація Процесу

DEVELOPMENT OF A WEB PAGE FOR CONTROLLING A MOBILE ROBOT VIA MQTT PROTOCOL

T. Lykho

Kharkiv National University of Radio Electronics

Ukraine, 62433, Liubotyn, Schmidt Str., 9

E-mail: tymur.lykho@nure.ua

Annotation: This article explores the main aspects of creating a web interface for controlling a robotic surveillance platform based on ESP32-CAM. As a result of the work, we will conduct testing of the interaction between the robot and the user interface via the MQTT protocol.

Keywords: Mobile Robot, Surveillance, Web Page, Interface, Mqtt Protocol, Process Innovation

У наші часи все більше набирають обертів системи охоронної та/або пожежної сигналізації, «розумний дім» і подібні. Альтернативним варіантом системи відеонагляду є мобільний робот з камерою на базі ESP32-CAM.

Для керування таким роботом необхідна зручна веб-сторінка, яка буде мати всі необхідні елементи керування роботом та вікно для відображення відеопотоку з камери, розроблення такої сторінки буде розглянуто у даній роботі.

MQTT - це стандартизований протокол або набір правил для обміну повідомленнями, який використовується для взаємодії між комп'ютерами. Інтелектуальні датчики, носимі пристрої та інші пристрої Інтернету речей (IoT) зазвичай передають та отримують дані через мережі з обмеженими ресурсами та пропускну здатністю. До основних переваг протоколу MQTT відносять його простоту та ефективність, масштабованість, надійність та безпечність.

Протокол MQTT працює на принципах моделі «видавець/підписник». У традиційній мережній комунікації клієнти та сервери спілкуються один з одним безпосередньо. Клієнти запитують ресурси або дані від сервера, після чого сервер обробляє та відправляє відповідь. Проте MQTT використовує модель видавець/підписник для роз'єднання відправника повідомлення (видавця) від отримувача повідомлення (підписника). Замість цього третій компонент, який називається мережним брокером, відповідає за зв'язок між видавцями та підписниками. Завдання брокера полягає в фільтрації всіх вхідних повідомлень від видавців та їх правильного розподілу підписникам. Брокер роз'єднує видавців та підписників за наступними принципами:

– просторове роз'єднання, тобто видавець та підписник не мають інформації про мережеве розташування один одного і не обмінюються такими даними, як IP-адреси або номери портів;

– часове роз'єднання, за цим принципом видавець на підписник не використовують, або не мають одночасного мережевого з'єднання;

– роз'єднання синхронізації, тобто видавці та підписники можуть відправляти та отримувати повідомлення без перешкод один одному, що забезпечує асинхронність. [1]

Таким чином протокол MQTT задовольняє всім потребам проекту мобільного робота відеонагляду.

Для керування роботом на отримання відеопотоку з камери робота необхідно розробити веб-сторінку, що буде містити основні елементи керування мобільним роботом, а саме:

- кнопки напрямку руху мобільної платформи;
- регулятор швидкості руху мобільної платформи;
- кнопки керування механізмом нахилу та панорамування;
- регулятор яскравості спалаху камери;
- вікно із відотоком з камери робота.

За наведеними вимогами до веб-інтерфейсу робота було створено веб-сторінку з використанням HTML та CSS, яку наведено на рисунку 1.



Рисунок 1 – Вигляд веб-інтерфейсу роботизованої платформи відеонагляду

Розглянемо елементи що містяться на сторінці керування роботом. Посередині маємо вікно, у якому буде відображатись відеопотік з камери робота, у разі, якщо зображення не надходить буде виводитись картинка «Немає зв'язку», як показано на рисунку 1. З лівої сторони бачимо клавіші керування механізмом нахилу та панорамування, а саме кнопки нахилу «вверх» та «вниз», кнопки повороту камери «в ліво» та «в право», а також центральну кнопку, що встановлює центральне положення камери. У правій частині сторінки бачимо кнопки для керування роботизованою платформою, а саме кнопки «вперед», «назад», «в ліво», «в право» та «стоп». Також було додано регулятор яскравості спалаху камери, регулятор швидкості руху платформи та випадаючий список, за допомогою якого користувач буде мати можливість змінювати роздільну здатність відеопотоку, що у ситуаціях поганого Інтернет-з'єднання дозволить отримати більшу кількість кадрів.

1. Основні функції для керування роботом з веб-інтерфейсу через протокол MQTT.

Після розробки інтерфейсу робота відеонагляду постає задача як передавати та отримувати данні з веб-сторінки до робота і навпаки. Для вирішення цієї задачі у першій частині цієї статті було обрано протокол MQTT через його зручність та простоту у використанні.

Для написання логічної частини веб-сторінки будемо використовувати мову програмування JavaScript, бо ця мова програмування дозволяє легко реалізувати інтерактивні і динамічні елементи інтерфейсу, при чому логічна частина може знаходитись прямо у HTML-документі, що дозволяє використовувати веб-інтерфейс без розгортання на сервері.[2]

Тож тепер можна перейти до розгляду функцій, що реалізують передачу даних між інтерфейсом та роботом.

Для початку необхідним є підключення до MQTT-брокера, це можна реалізувати, наприклад, з використанням популярної бібліотеки Paho, щоб все працювало коректно її необхідно підключити до нашого HTML-документу за допомогою такої конструкції:

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/paho-mqtt/1.0.1/mqttws31.min.js" type="text/javascript" ></script>. [3]
```

Тепер необхідно написати код підключення до MQTT-брокера, підписку на необхідні «topics» та відслідковування помилок підключення, для цього було написано такий код:

```
var client = new Paho.MQTT.Client("broker.mqttdashboard.com", 8000, "clientId" + new Date().getTime());
client.onConnectionLost = function(responseObject) {
    if (responseObject.errorCode !== 0) {
        console.log("Connection lost:", responseObject.errorMessage);
    }
};
client.connect({
    onSuccess: function() {
        console.log("Connected to MQTT server");
        client.subscribe("camera/test");
    }
});
```

Передача відеопотоку у даному випадку відбувається за допомогою кодування кожного кадру у формат BASE64, що дозволяє перетворити зображення у текстовий вигляд і передати через протокол MQTT. Для того щоб отримати і відобразити потік кадрів на сторінці необхідно додати таку функцію:

```
client.onMessageArrived = function(message) {
    var imageData = message.payloadString;
    if (imageData != undefined && imageData.length > 0) {
        $("#streamedImage").attr("src", imageData);
    }
};
```

Ця функція перевіряє чи містяться данні у топіку «camera/test», якщо так, тоді данні передаються у HTML-елемент img з ідентифікатором «streamedImage».

Завдяки тому, що браузерери можуть автоматично розкодувати формат BASE64 немає необхідності у написанні додаткових функцій декодування кадрів.

Також є необхідність відправляти дані роботу, для цього зручним є використання такої функції:

```
function sendCommand(topic, command) {
    var message = new Paho.MQTT.Message(command);
    message.destinationName = topic;
```

```

    client.send(message);
  }

```

Таким чином ми можемо скоротити написання коду, наприклад, виклик команди на переміщення робота або управління механізмом нахилу та панорамування буде виглядати так:

```

$( ".button, .buttonDrive" ).mousedown( function() {
    var command = $(this).data("command");
    sendCommand(keyboardTopic, command);
});

```

Функція працює таким чином: коли користувач натискає на одну з кнопок відправляються відповідні дані у топик «robot/keyboard», який було збережено у окрему змінну keyboardTopic для читабельності коду, а коли користувач відпускає кнопку виконується функція, яка зупиняє рух робота автоматично:

```

$( ".buttonDrive" ).mouseup( function() {
    sendCommand(keyboardTopic, "driveStop");
});

```

За таким самим принципом реалізовано і функції для керування швидкістю переміщення робота та яскравістю спалаху камери:

```

$( "#speedRange" ).change( function() {
    var speed = $(this).val();
    sendCommand(keyboardTopic, "speed:" + speed);
});
$( "#brightnessRange" ).change( function() {
    var brightness = $(this).val();
    sendCommand(flashTopic, "brightness:" + brightness);
});

```

Наведені вище функції відслідковують положення регулятора з відповідним ідентифікатором та передають значення положень у випадку, коли положення регулятора змінюється.

Зміну роздільної здатності відеопотоку реалізовано наступною функцією:

```

$( ".dropdown-item" ).click( function() {
    var quality = $(this).data("quality");
    sendCommand(quantityTopic, "quality:" + quality);
});

```

2. Тестування взаємозв'язку інтерфейсу та робота

Для проведення тестування роботи інтерфейсу користувача для управління основними функціями роботизованої платформи відеонагляду плату ESP32-CAM було підключено до СОМ-порту комп'ютеру. На рисунку 2 наведено вигляд робочого інтерфейсу та монітор порту у який виводяться дані про натиснуті кнопки.

В результаті проведення дослідів можна зробити висновок, що взаємозв'язок між інтерфейсом та роботизованою платформою відеонагляду за допомогою протоколу MQTT реалізовано коректно і задовольняє потребам проекту.



Рисунок 2 – Результат проведення тестування роботи веб-інтерфейсу

ВИСНОВКИ. Основні результати досліджень:

- визначено основні переваги MQTT-протоколу для використання у даному проєкті;
- розроблено інтерфейс користувача для керування роботизованою платформою відеонагляду за допомогою HTML, CSS;
- використовуючи мову програмування JavaScript реалізовано логіку інтерфейсу;
- проведено тестування роботи розробленого інтерфейсу та зв'язку з роботизованою платформою.

Практична цінність:

- застосування мобільного робота відеонагляду актуально в умовах війни, коли безпека стає пріоритетом, і багато людей покидають свої будинки;
- використання дистанційного керування та можливості перегляду зображення у реальному часі робить систему зручною та функціональною.

Подальші перспективи:

- можливість розширення функціоналу веб-застосунку, наприклад, додавання нових роботів, ресстрація та авторизація, можливість зберігати певні кадри з камери робота, тощо;
- інтеграція з системами «розумного дому» для більш широкого застосування в повсякденному житті.

Загалом, розробка мобільного робота для відеонагляду має перспективи в різних сферах, привертає увагу своєю ефективністю та можливістю використання в різноманітних сценаріях.

ЛІТЕРАТУРА

1. What is MQTT? - MQTT Protocol Explained - AWS. Amazon Web Services, Inc. URL: https://aws.amazon.com/what-is/mqtt/?nc1=h_ls (date of access: 25.03.2024).
2. Ivashchuk I. Чому JavaScript – перспективна мова програмування? Поради початківцям. Dou.ua. 2021. URL: <https://dou.ua/forums/topic/35184/> (дата звернення: 26.03.2024).
3. Eclipse paho | the eclipse foundation. Projects Gateway | The Eclipse Foundation. URL: <https://eclipse.dev/paho/index.php?page=clients/js/index.php> (date of access: 26.03.2024).
4. Akopov, M., Maksymova, S., & Yevsieiev, V. (2023). Choosing a Camera for 3D Mapping. *Journal of Universal Science Research*, 1(11), 28-38.

5. Nikitin, V., Maksymova, S., & Yevsieiev, V. (2023). Traffic Signs Recognition System Development. *Multidisciplinary Journal of Science and Technology*, 3(3), 235-242.
6. Nevludov, I., Ponomaryova, G., Bortnikova, V., Maksymova, S., & Kolesnyk, K. (2018). MEMS accelerometer in hexapod intellectual control. In 2018 XIV-th International Conference on Perspective Technologies and Methods in MEMS Design (MEMSTECH) (pp. 146-150). IEEE.
7. Yevsieiev, V., Maksymova, S., & Starodubcev, N. (2022). Development of an Algorithm for ESP32-Cam Operation in HTTP Server Mode for Streaming Video. *Collection of scientific papers «ΑΙΟΓΟΣ»*, (July 8, 2022; Paris, France), 177-179.
8. Attar, H., & et al. (2022). Zoomorphic Mobile Robot Development for Vertical Movement Based on the Geometrical Family Caterpillar. *Computational Intelligence and Neuroscience*, 2022, Article ID 3046116, <https://doi.org/10.1155/2022/3046116>.
9. Al-Sharo, Y., Abu-Jassar, A., Lyashenko, V., Yevsieiev, V., Maksymova, S. A Robo-hand prototype design gripping device within the framework of sustainable development, *Indian Journal of Engineering*, 20 2023 e37ije1673. <https://doi.org/10.54905/diss.v20i54.e37ije1673>
10. Lyashenko, V., Abu-Jassar, A.T., Yevsieiev, V., Maksymova, S. Automated Monitoring and Visualization System in Production, *Int. Res. J. Multidiscip. Technovation*, 5(6) 2023 09-18. <https://doi.org/10.54392/irjmt2362>
11. Nevludov, I., & et al. (2020). Method of Algorithms for Cyber-Physical Production Systems Functioning Synthesis. *International Journal of Emerging Trends in Engineering Research*, 8(10), 7465-7473.
12. Svitlana Maksymova, Vladyslav Yevsieiev, & Amer Abu-Jassar. (2024). Gripping Device Development: Some Aspects. *Journal of Universal Science Research*, 2(1), 150-158
13. Svitlana Maksymova, & Vladyslav Yevsieiev. (2024). Coin Counting Device Kinematic Diagram Development. *Journal of Universal Science Research*, 2(1), 159-168.
14. Svitlana Maksymova, Vladyslav Yevsieiev, & Ahmad Alkhalaileh. (2024). The Monitoring System Architecture Development. *Journal of Universal Science Research*, 2(1), 69-79.
15. Svitlana Maksymova, Vladyslav Yevsieiev, & Amer Abu-Jassar. (2024). Gripping Device Development: Some Aspects. *Journal of Universal Science Research*, 2(1), 150-158.
16. Svitlana Maksymova, & Vladyslav Yevsieiev. (2024). Coin Counting Device Kinematic Diagram Development. *Journal of Universal Science Research*, 2(1), 159-168.
17. Svitlana Maksymova, Vladyslav Yevsieiev, & Amer Abu-Jassar. (2024). The Bipedal Robot a Kinematic Diagram Development. *Journal of Universal Science Research*, 2(1), 6-17.
18. Andrii Bondariiev, Svitlana Maksymova, & Vladyslav Yevsieiev. (2023). Automated Monitoring System Development for Equipment Modernization. *Journal of Universal Science Research*, 1(11), 6-16. Retrieved from <https://universalpublishings.com/index.php/jusr/article/view/2484>
19. Mykhailo Akopov, Svitlana Maksymova, & Vladyslav Yevsieiev. (2023). Choosing a Camera for 3D Mapping. *Journal of Universal Science Research*, 1(11), 28-38. Retrieved from <https://universalpublishings.com/index.php/jusr/article/view/2486>
20. Vladyslav Nikitin, Svitlana Maksymova, & Vladyslav Yevsieiev. (2023). TRAFFIC SIGNS RECOGNITION SYSTEM DEVELOPMENT . *Multidisciplinary Journal of Science and Technology*, 3(3), 235-242. Retrieved from <https://mjstjournal.com/index.php/mjst/article/view/225>
21. Yevsieiev, V. Comparative Analysis of the Characteristics of Mobile Robots and Collaboration Robots Within INDUSTRY 5.0. / V. Yevsieiev, D. Gurin // In the VI International Scientific and Theoretical Conference, September 8, 2023. Chicago, USA. P.92-94

ДОДАТОК Д
Керівництво користувача

ВСТУП

Розроблена система використовується для віддаленого відеоспостереження. Програмну частину мобільного робота розроблено на мові програмування Arduino. Програмний засіб для дистанційного керування мобільною системою відеонагляду спроектовано та розроблено на мові програмування Python з використанням фреймворку Django, а також JavaScript у програмному середовищі PyCharm Community Edition 2023.2.4.

Розроблена система складається з апаратної частини (мобільного робота з камерою), серверної частини та графічного інтерфейсу. Програму можна використовувати на різних пристроях, як на смартфонах під керуванням операційних системами Android або iOS, так і на комп'ютері з операційними системами Windows, Linux або Mac OS. Для запуску програми використовується віртуальна машина Python.

1 ПРИЗНАЧЕННЯ І УМОВИ ЗАСТОСУВАННЯ

Розроблена система використовується для віддаленого відеоспостереження та дозволяє користувачу мобільного робота керувати ним з будь-якого місця, за умови підключення до мережі Інтернет.

Зареєстрований користувач веб-застосунку має можливість додавати, налаштовувати та видаляти своїх мобільних роботів, керувати ними, переглядати трансляцію з відеокамери, що встановлена на роботі, та робити фото. Фото можна зберігати на сервері, переглядати та видаляти з серверу.

Зі сторони адміністратора (зареєстрованого користувача з правами суперкористувача) користувач має можливості з видалення, додавання, зміни даних інших користувачів.

Перед початком роботи необхідно ознайомитись з посібником користувача.

Система дозволяє вирішити проблеми віддаленого відеоспостереження у малодоступних для людини місцях.

Технічні вимоги: від 2 Гб ОЗП, процесор від x64: 2 ГГц.

2 ПІДГОТОВКА ДО РОБОТИ

Для користування системою необхідним є виконання наступних умов:

- користувач має персональний комп'ютер або телефон з підключенням до мережі Інтернет;
- користувач має мобільного робота ESP32-CAMBOT;
- користувач володіє інформацією щодо використання системи;
- користувач ознайомлений з «Посібником користувача».

Для початку роботи необхідно увімкнути мобільного робота, що має налаштування за замовчуванням. Робот буде знаходитись у режимі налаштувань, про що проінформує білим (спалахом) та червоним світлодіодом, спалах та червоний світлодіод повинні моргнути 6 разів, після чого робот створить точку доступу Wi-Fi для зміни налаштувань Wi-Fi та MQTT.

Користувачу необхідно знайти у доступних мережах Wi-Fi точку з назвою «ESP32-CAM-Setup» та підключитись до неї, пароль «12345678». Тепер необхідно відкрити будь-який браузер і перейти за адресою <http://192.168.4.1/>, рис. 2.1.

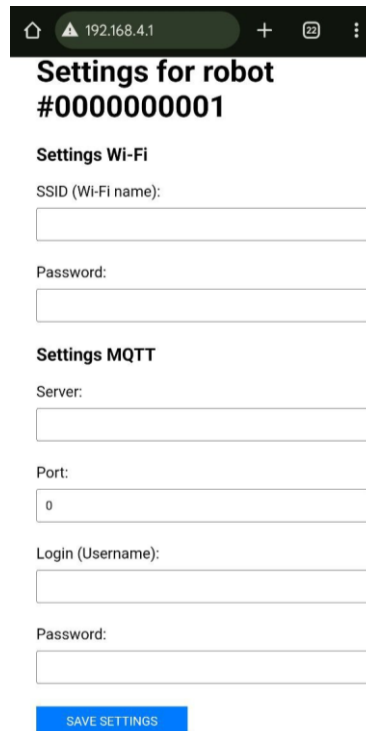
Далі необхідно ввести назву та пароль від мережі Wi-Fi до якої буде підключено MP, а також ввести дані для підключення до MQTT серверу, рис. 2.2. Натиснути кнопку «Save settings».

Робот перезавантажиться і у разі успішного підключення до Wi-Fi червоний індикатор буде постійно увімкнений, а білий швидко моргне 3 рази.

Мобільний робот готовий до роботи.

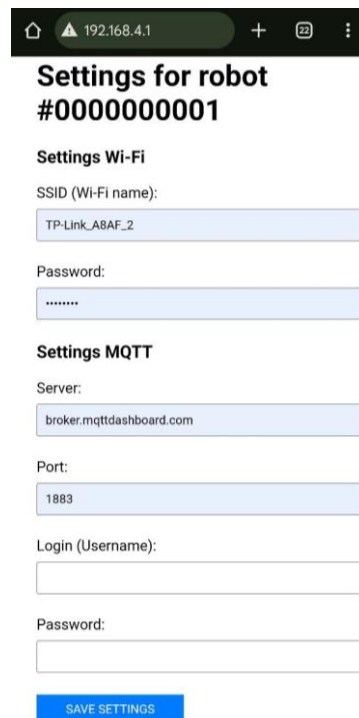
Наступним кроком необхідно пройти реєстрацію у веб-застосунку для керування мобільною системою. Для цього необхідно перейти за посиланням <http://127.0.0.1:8000/remoteapp/> (рис. 2.3) та натиснути на посилання «Registration», користувач побачить форму реєстрації (рис. 2.4), яку необхідно заповнити своїми даними. Нового користувача буде автоматично

авторизовано у системі і перенаправлено на сторінку для додавання нового робота (рис. 2.4).



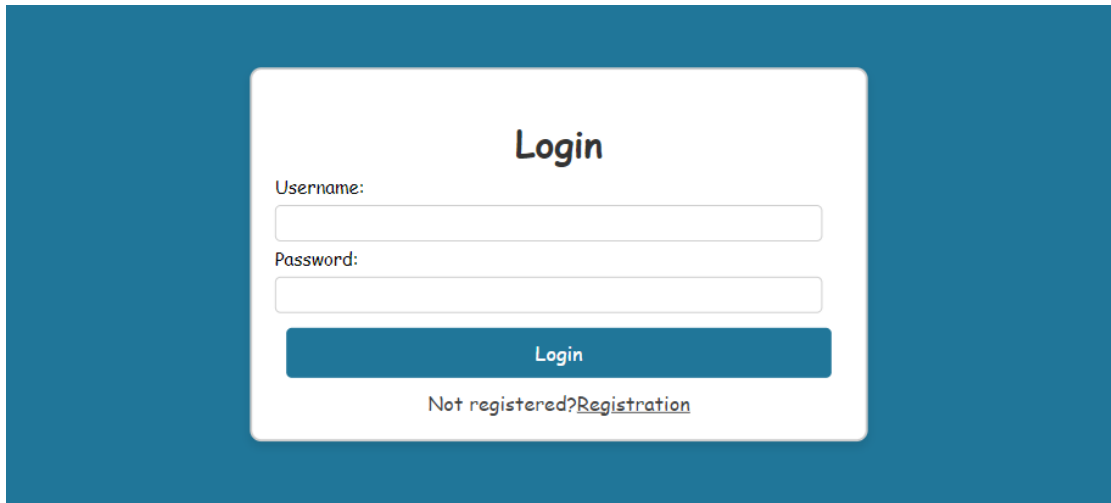
The screenshot shows a mobile browser interface for configuring a robot. The title is "Settings for robot #000000001". Under the "Settings Wi-Fi" section, there are two empty text input fields for "SSID (Wi-Fi name)" and "Password". Under the "Settings MQTT" section, there are four input fields: "Server" (empty), "Port" (containing "0"), "Login (Username)" (empty), and "Password" (empty). A blue "SAVE SETTINGS" button is at the bottom.

Рисунок 2.1 – Порожня форма налаштувань мобільного робота



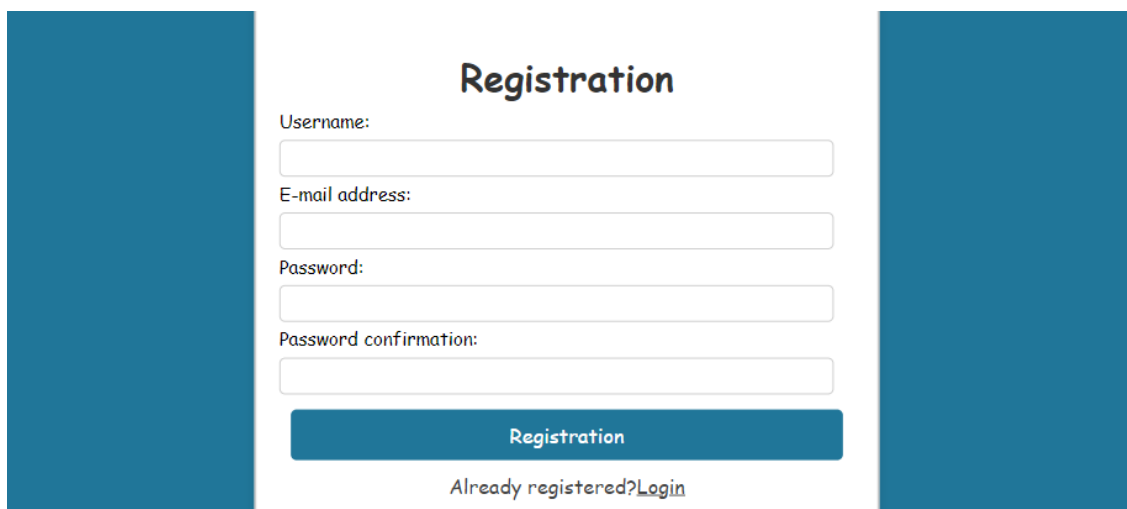
The screenshot shows the same mobile browser interface as Figure 2.1, but with the input fields filled. Under "Settings Wi-Fi", the "SSID (Wi-Fi name)" field contains "TP-Link_A8AF_2" and the "Password" field contains ".....". Under "Settings MQTT", the "Server" field contains "broker.mqttdashboard.com", the "Port" field contains "1883", and the "Login (Username)" and "Password" fields are empty. The blue "SAVE SETTINGS" button remains at the bottom.

Рисунок 2.2 – Заповнена форма налаштувань мобільного робота



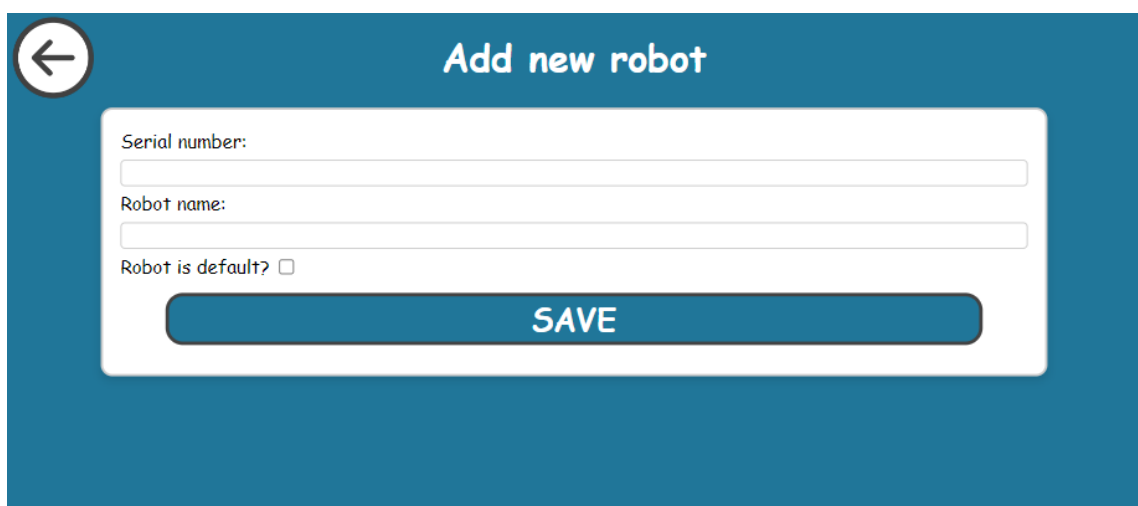
The image shows a login form titled "Login" centered on a white background with a blue border. It features two input fields: "Username:" and "Password:". Below the fields is a blue button labeled "Login". At the bottom, there is a link: "Not registered? [Registration](#)".

Рисунок 2.3 – Стартова сторінка веб-застосунку



The image shows a registration form titled "Registration" centered on a white background with a blue border. It features four input fields: "Username:", "E-mail address:", "Password:", and "Password confirmation:". Below the fields is a blue button labeled "Registration". At the bottom, there is a link: "Already registered? [Login](#)".

Рисунок 2.4 – Форма реєстрації користувача



The image shows a form titled "Add new robot" on a blue background. In the top left corner, there is a circular button with a white left-pointing arrow. The form itself is white and contains three input fields: "Serial number:", "Robot name:", and "Robot is default?" with a checkbox. Below the fields is a blue button labeled "SAVE".

Рисунок 2.5 – Форма додавання нового робота

Користувач повинен заповнити форму, при чому серійний номер робота повинен бути унікальним, він вказується на сторінці налаштувань робота (рис.2.1) та натиснути кнопку «Save».

Тепер користувач повинен налаштувати підключення до MQTT (рис. 2.6).

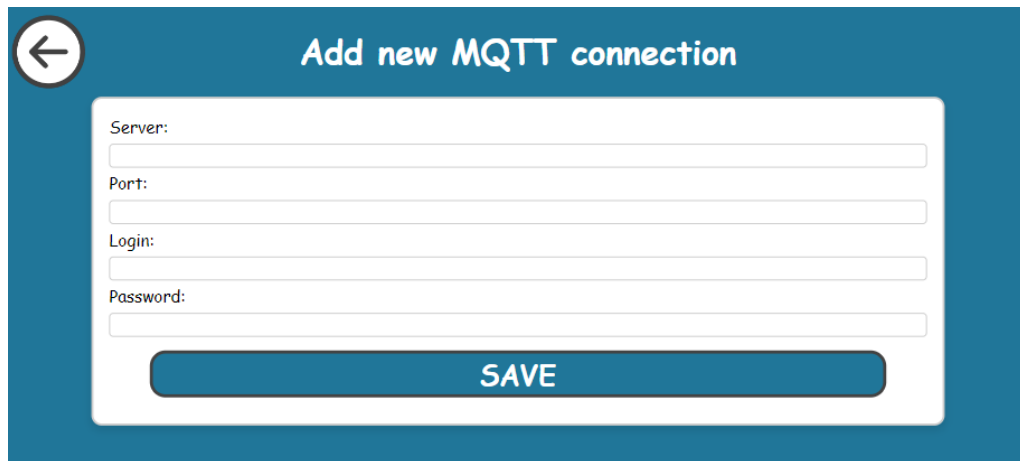


Рисунок 2.6 – Форма для налаштування підключення до MQTT

У разі правильного введення всіх даних користувач готовий до використання системи мобільного відеонагляду, рис. 2.7.



Рисунок 2.7 – Головне вікно веб-застосунку

3 ОПИС ОПЕРАЦІЙ

3.1 Авторизація користувача

Для доступу до веб-застосунку користувач повинен пройти процедуру авторизації. Для цього необхідно перейти за посиланням <http://127.0.0.1:8000/remoteapp/> та заповнити форму авторизації (рис. 3.1) відповідними даними, після чого натиснути кнопку «Login».

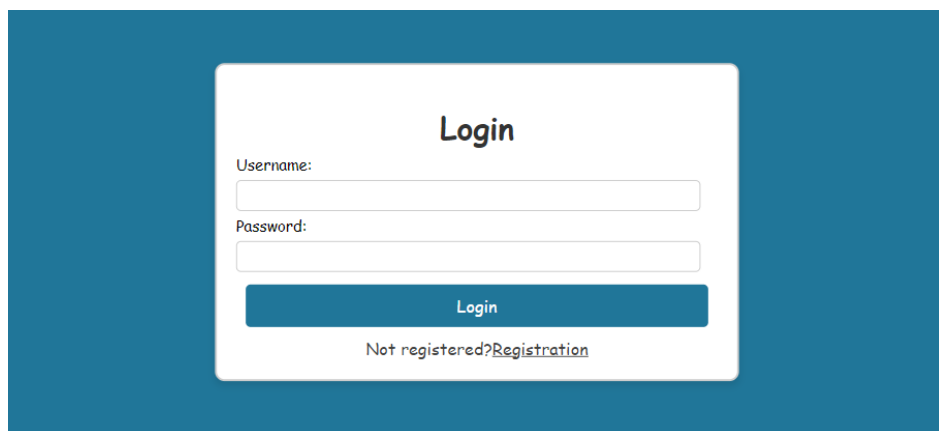
The image shows a login form titled "Login" centered on a white background with a blue border. It features two input fields: "Username:" and "Password:". Below the fields is a blue button labeled "Login". At the bottom, there is a link: "Not registered? [Registration](#)".

Рисунок 3.1 – Сторінка авторизації

Якщо всі дані було введено коректно, користувача буде перенаправлено на головне вікно веб-застосунку. У випадку помилки авторизації, користувач отримає сповіщення (рис. 3.2).

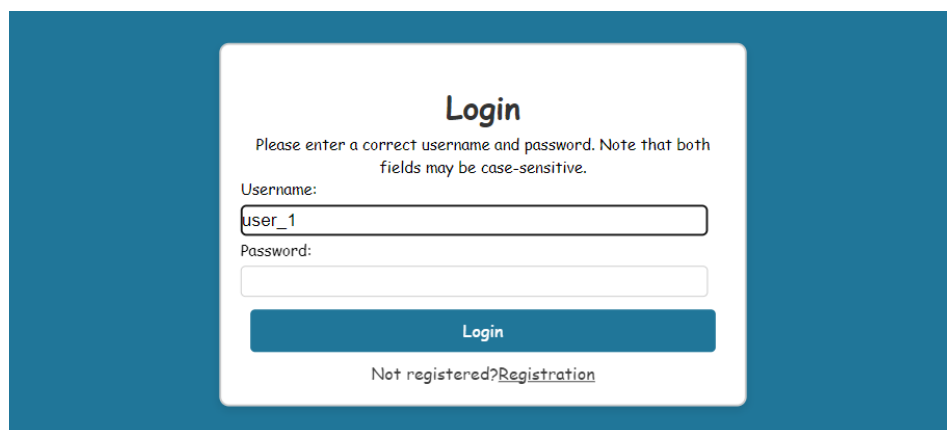
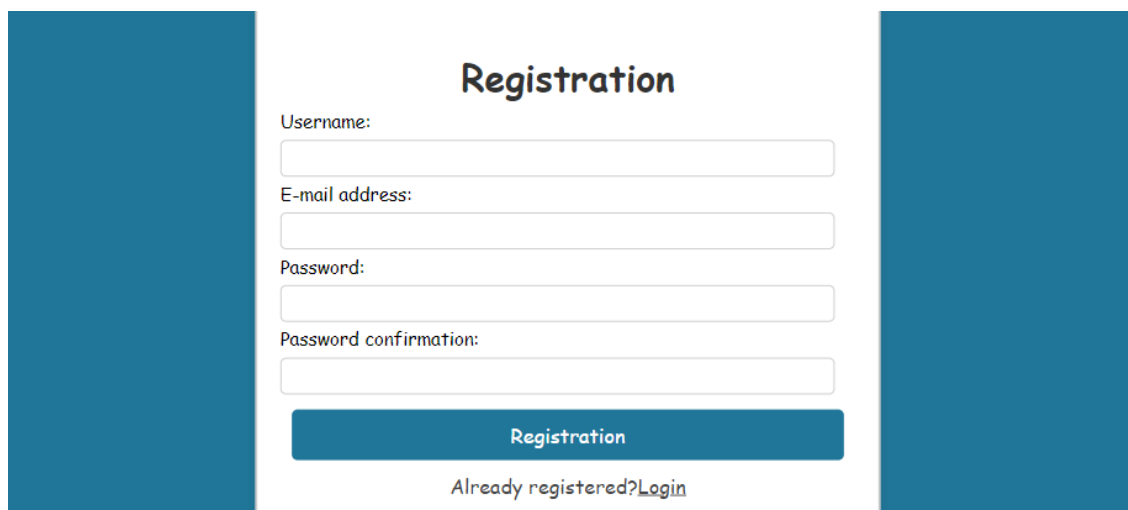
The image shows the same login form as in Figure 3.1, but with an error message displayed above the input fields: "Please enter a correct username and password. Note that both fields may be case-sensitive." The "Username:" field contains the text "user_1". The "Password:" field is empty. The "Login" button and the "Registration" link are still visible at the bottom.

Рисунок 3.2 – Повідомлення про помилку авторизації

Помилка може свідчити про неправильність введених даних, або відсутність облікового запису у БД.

3.2 Реєстрація користувача

Для реєстрації користувач повинен перейти за посиланням <http://127.0.0.1:8000/remotearp/register/> та заповнити форму реєстрації (рис. 3.3). У випадку успішної реєстрації, користувача буде перенаправлено на сторінку додавання робота.



The image shows a web registration form with the following elements:

- Registration** (Title)
- Username:** [Input field]
- E-mail address:** [Input field]
- Password:** [Input field]
- Password confirmation:** [Input field]
- Registration** (Submit button)
- [Already registered? Login](#) (Link)

Рисунок 3.3 – Форма реєстрації користувача

Поле «Username» є індивідуальним для кожного користувача системи і у разі спроби повторної реєстрації на той самий логін користувач отримає помилку (рис. 3.4), в даному випадку, якщо користувач має обліковий запис у системі необхідно пройти процедуру авторизації, або, якщо користувач не реєструвався у системі та отримав дану помилку, слід змінити логін на інший.

Registration

Username:

A user with that username already exists.

E-mail address:

Password:

Password confirmation:

Registration

[Already registered? Login](#)

Рисунок 3.4 – Повідомлення про помилку реєстрації

3.3 Головна сторінка веб-застосунку

У випадку успішної реєстрації або авторизації, користувач потрапляє на сторінку керування мобільним роботом, рисунок 3.5.



Рисунок 3.5 – Головне вікно веб-застосунку

3.4 Меню налаштувань

Щоб перейти у меню налаштувань потрібно у головному вікні натиснути кнопку «Меню», після чого користувача буде перенаправлено на сторінку налаштувань, рис. 3.6.

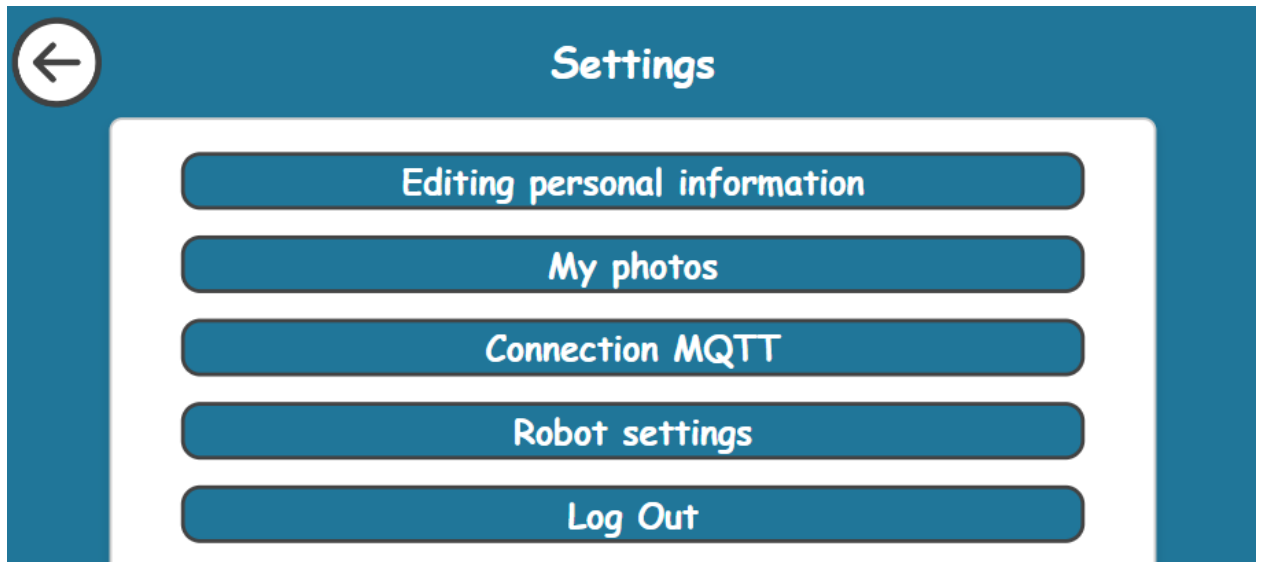


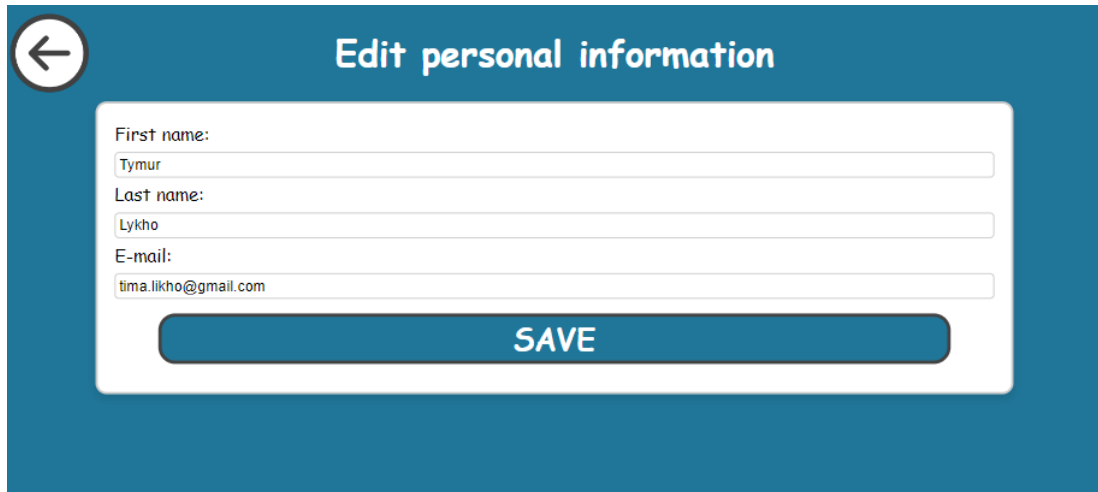
Рисунок 3.6 – Сторінка налаштувань

Тут користувач може перейти до редагування особистих даних, перегляду фотогалереї, редагування налаштувань MQTT, редагування налаштувань роботів та вийти з поточного сеансу.

3.5 Редагування особистих даних користувача

Для цього необхідно перейти у пункт меню «Editing personal information» з меню налаштувань, після чого користувача буде переадресовано на сторінку, що зображена на рис. 3.7.

На цій сторінці користувач може переглянути та за потреби змінити особисті дані. Для зміни особистих даних необхідно обрати потрібний пункт, видалити поточні дані з поля та заповнити новими, натиснути кнопку «Save». Після чого дані буде оновлено у БД.



← Edit personal information

First name:
Tymur

Last name:
Lykho

E-mail:
tima.likho@gmail.com

SAVE

Рисунок 3.7 – Форма перегляду та зміни особистих даних користувача

3.6 Сторінка фотогалереї

Для переходу у фотогалерею необхідно натиснути кнопку «My photos» у меню налаштувань, рис. 3.8.

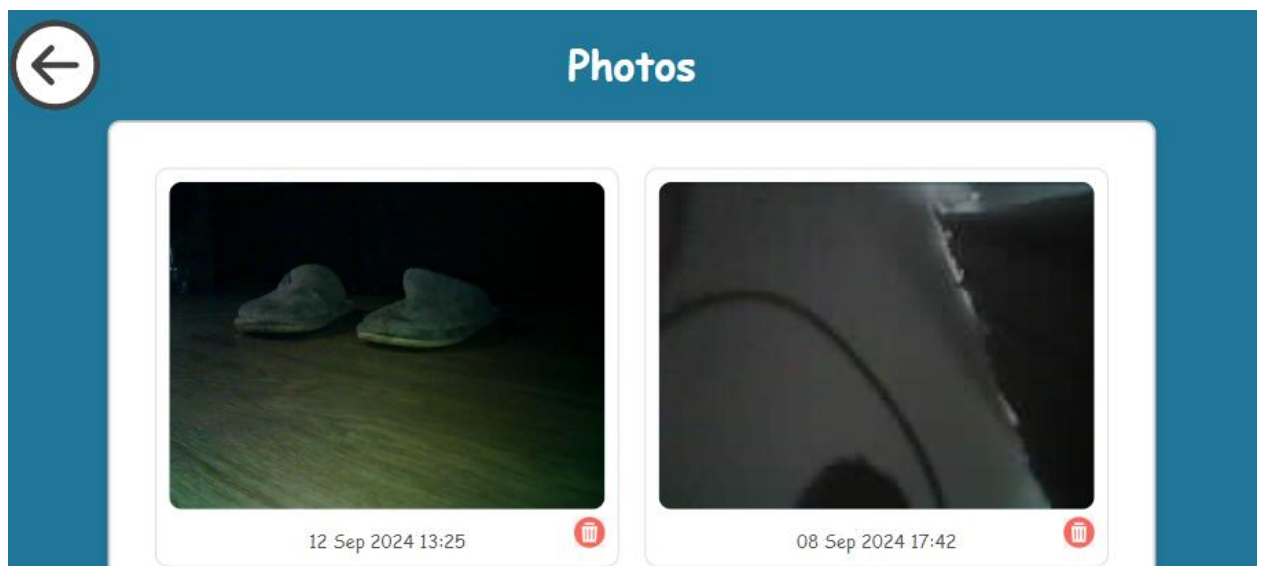


Рисунок 3.8 – Сторінка фотогалереї користувача

Користувач може натиснути на фото для більш детального перегляду у новому вікні, переглянути інформацію, коли було зроблено фото, а також видалити фото з БД, натиснувши на червону кнопку.

3.7 Сторінка налаштувань MQTT

Для переходу на сторінку налаштувань MQTT, користувач повинен натиснути необхідний пункт «Connection MQTT» у меню налаштувань, якщо у користувача додано більше одного робота, буде запропоновано обрати певного робота для зміни його налаштувань MQTT, рис. 3.9.

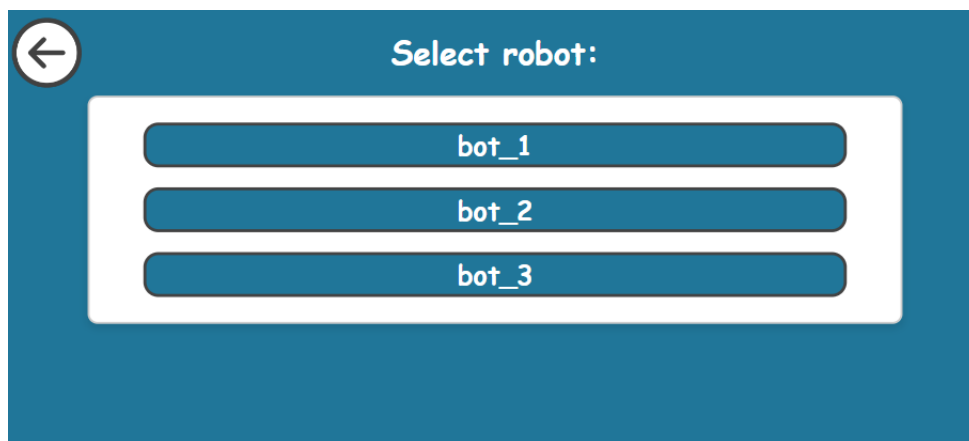


Рисунок 3.9 – Сторінка вибору робота

Користувач повинен обрати робота, налаштування MQTT якого він бажає змінити, після чого його буде переадресовано на сторінку перегляду та редагування даних MQTT для обраного робота, рисунок 3.10.

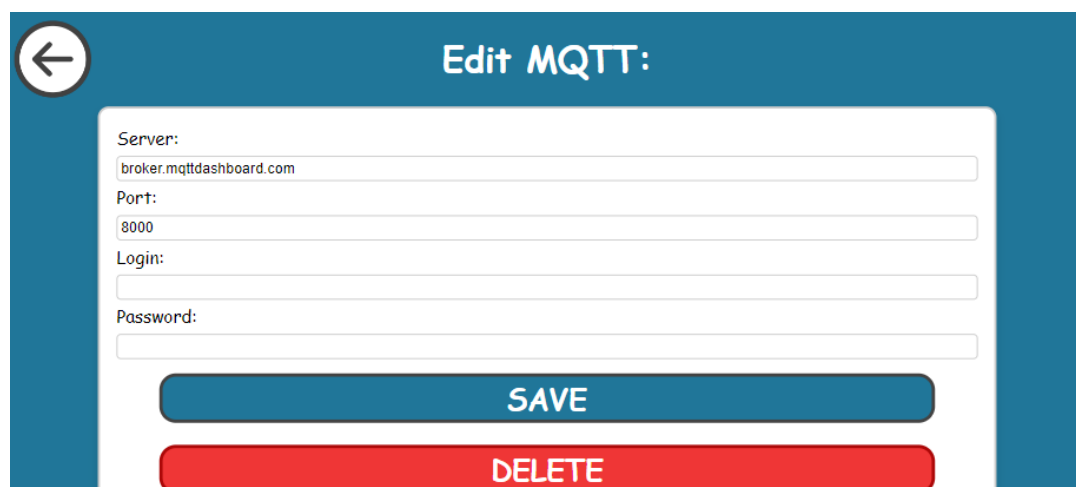


Рисунок 3.10 – Сторінка перегляду та зміни налаштувань MQTT певного робота

Для зміни налаштувань для підключення до MQTT необхідно обрати потрібний пункт, видалити поточні дані з поля та заповнити новими, натиснути кнопку «Save». Після чого дані буде оновлено у БД.

Якщо потрібно повністю видалити дані про підключення, необхідно натиснути кнопку «Delete».

3.8 Сторінка налаштувань робота

Для переходу на сторінку налаштувань робота, користувач повинен натиснути необхідний пункт «Robot settings» у меню налаштувань, якщо у користувача додано більше одного робота, буде запропоновано обрати певного робота для зміни його налаштувань, або додати нового робота, рис. 3.11.

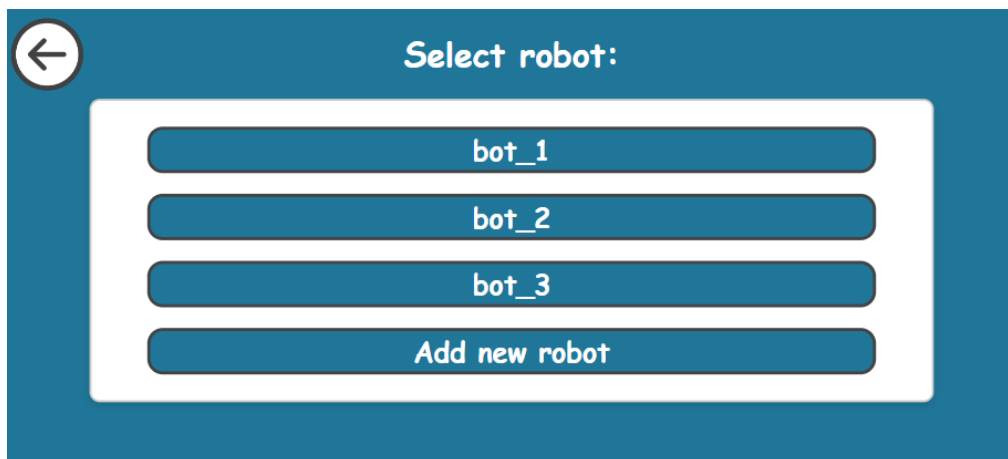


Рисунок 3.11 - Сторінка вибору робота або додавання нового робота

Користувач повинен обрати робота, налаштування якого він бажає змінити, після чого його буде переадресовано на сторінку перегляду та редагування даних обраного робота, рисунок 3.12.

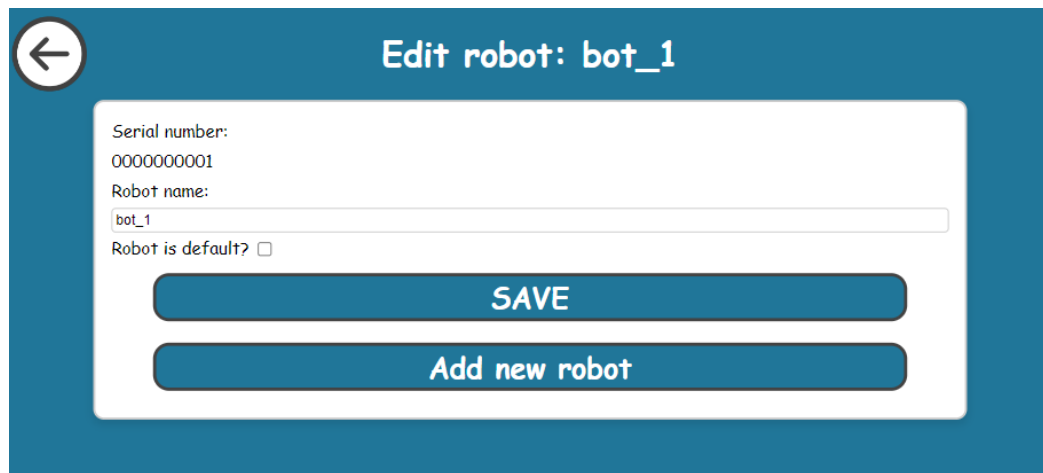


Рисунок 3.12 – Сторінка перегляду та редагування даних робота

На цій сторінці користувач може переглянути серійний номер робота, переглянути або змінити назву робота, обрати цього робота, як робота за замовчуванням, а також може додати нового робота. Якщо користувач змінює певні дані, необхідно натиснути кнопку «Save» для збереження даних.

3.9 Сторінка додавання нового робота

Для додавання нового робота користувача буде переадресовано на сторінку додавання нового робота, рис. 3.13.

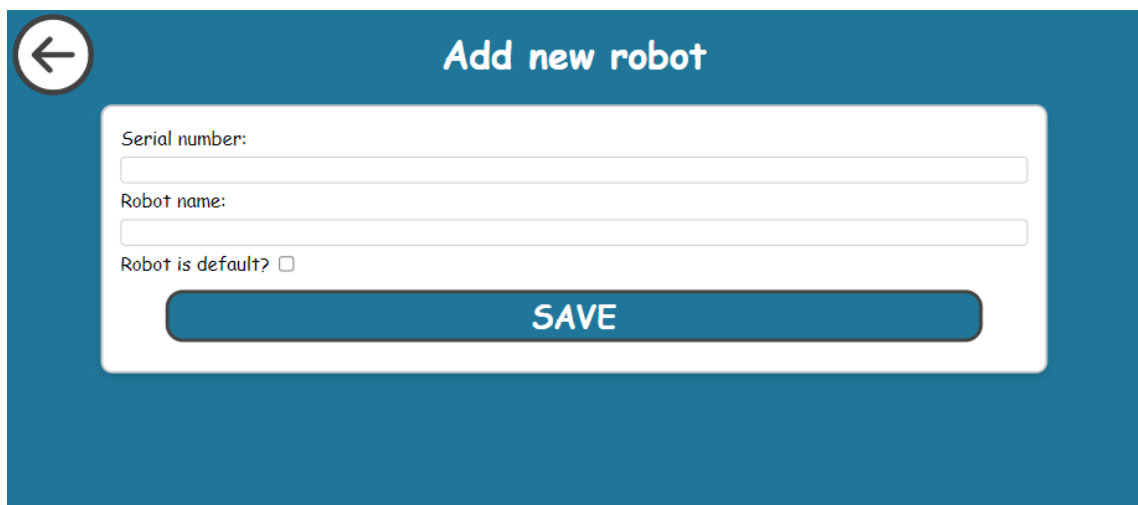


Рисунок 3.13 – Форма додавання нового робота

Користувачу необхідно ввести унікальний серійний номер робота, його ім'я та обрати чи буде використовуватись цей робот за замовчуванням, після чого натиснути кнопку «Save».

3.10 Вихід з облікового запису

Для виходу з облікового запису користувачу необхідно обрати пункт меню налаштувань «Log Out», після чого користувача буде переадресовано на сторінку авторизації.

3.11 Режим налаштувань мобільного робота

Користувачу може бути необхідним змінити налаштування мобільного робота, для цього необхідно на увімкненому роботі набрати комбінацію 3 коротких (менше 1 секунди) натискання після чого довге натискання (не менше 2 секунд, не більше 10 секунд), після чого робот плавно увімкне та вимкне спалах, що означає перехід робота у режим налаштувань. Робот перезавантажиться та моргне спалахом та червоним світлодіодом 6 разів, що інформує про створення точки доступу для зміни налаштувань робота.

Користувачу необхідно знайти у доступних мережах Wi-Fi точку з назвою «ESP32-CAM-Setup» та підключитись до неї, пароль «12345678». Тепер необхідно відкрити будь-який браузер і перейти за адресою <http://192.168.4.1/>, рис. 3.14.

Далі можна переглянути та/або змінити назву та пароль від мережі Wi-Fi до якої буде підключено MP, а також дані для підключення до MQTT серверу, рис. 3.15. Натиснути кнопку «Save settings».

Робот перезавантажиться і у разі успішного підключення до Wi-Fi червоний індикатор буде постійно увімкнений, а білий швидко моргне 3 рази.

Settings for robot #000000001

Settings Wi-Fi

SSID (Wi-Fi name):

Password:

Settings MQTT

Server:

Port:

Login (Username):

Password:

Рисунок 3.14 – Форма з поточними налаштуваннями мобільного робота

The screenshot shows the same settings form as in Figure 3.14, but with a mobile browser interface at the top. The address bar shows the URL 192.168.4.1. The form fields are filled with the same data as in Figure 3.14: SSID 'TP-Link_A8AF_2', Password '.....', MQTT Server 'broker.mqttdashboard.com', and Port '1883'. The 'SAVE SETTINGS' button is highlighted in blue.

Рисунок 3.15 – Заповнена форма налаштувань мобільного робота

Якщо необхідно вийти з режиму налаштувань без зміни даних, користувач може повторно ввести комбінацію натискань кнопки 3 коротких, 1 довгий, після чого робот буде працювати у стандартному режимі.

3.12 Режим відладки мобільного робота

У випадку, коли мобільному роботу необхідне сервісне обслуговування його можна перевести у режим відладки для моніторингу внутрішніх процесів. Для того щоб увімкнути цей режим необхідно відключити від контактів RX та TX обладнання робота та підключити ці контакти до TTL-перетворювача, що дозволить використовувати монітор порта. Далі необхідно, за допомогою кнопки ввести комбінацію «5 коротких, один довгий та 5 коротких», після чого спалах плавно увімкнеться та вимкнеться 2 рази, після чого робот перезавантажиться та буде працювати у режимі відладки.

Для виходу з режиму відладки необхідно повторно ввести комбінацію натискань та дочекатись перезавантаження мобільного робота.

3.13 Скидання налаштувань мобільного робота до стандартних

Можуть виникати випадки, коли необхідно повністю видалити налаштування робота, для цього існує функція «Заводські налаштування». Для того щоб видалити всі налаштування робота необхідно затиснути кнопку мобільного робота на більше ніж 20 секунд. Після відпускання кнопки спалах буде моргати 10 разів, потім робот перезавантажиться, всі налаштування робота буде видалено.

4 АВАРІЙНІ СИТУАЦІЇ

Класифікація основних помилок та способи їх усунення вказані у таблиці 4.1.

Таблиця 4.1 – Основні типи помилок системи

№	Опис помилки	Спосіб усунення
1	Сервер не відповідає	1) Перевірити, чи запущений сервер; 2) Перевірити налаштування запуску сервера; 3) Перевірити з'єднання з Інтернетом.
2	Помилка формату даних	Перевірити правильність введених даних та виправити, відповідно до формату
3	Мобільний робот втратив з'єднання з Wi-Fi	1) Перевірити налаштування МР через режим налаштувань; 2) Перевірити наявність та працездатність мережі Wi-Fi.
4	Мобільний робот втратив з'єднання з MQTT	1) Виконати кроки, що описані у помилці №3; 2) Перевірити працездатність MQTT-серверу.
5	Відеопотік з камери зупинився самостійно	1) Виконати кроки, що описані у помилках №3 та 4; 2) Перезавантажити МР; 3) Перезапустити трансляцію відеопотоку.

Якщо під час роботи з системою буде втрачено з'єднання з сервером, то програма повідомить користувачеві про помилку з'єднання. Система буде намагатися відновити підключення до сервера при кожному наступному запуску функції системи.

5 РЕКОМЕНДАЦІЇ ЩОДО КОРИСТУВАННЯ СИСТЕМОЮ

Програма є простою та легкою у використанні. Згідно з результатами тестування зручності, звичайні користувачі швидко засвоюють, як почати роботу з системою. Отже, можна зробити висновок, що для використання системи не потрібні спеціальні навички або знання. Будь-який користувач з базовими навичками роботи з програмами в системі Windows, iOS та іншими операційними системами зможе використовувати цю програму.

Рекомендується ознайомитися з загальними принципами комп'ютерних інтерфейсів. Це допоможе вам швидше освоїти програму, оскільки багато програм будуються за схожими принципами та використовують подібні елементи, такі як контекстні меню, іконки та панелі інструментів. Особливо це помітно в програмах одного розробника.

Перед початком роботи з даною програмою рекомендується ознайомитися з керівництвом користувача. Це допоможе зрозуміти функції системи та сценарії взаємодії з нею.

Бажано не використовувати мобільного робота протягом довгого часу через особливості роботи MQTT-серверів.

Детальніше про основні функції взаємодії користувача з системою можна знайти в розділі 3 цього додатку.

ДОДАТОК Е
Текст програми

Е.1 – Лістинг коду програми мобільного робота

```

#include <ESP32Servo.h>
#include <WebServer.h>
#include <EEPROM.h>
#include <WiFi.h>
#include <PubSubClient.h>
#include <Arduino.h>
#include "soc/soc.h"
#include "soc/rtc_cntl_reg.h"
#include "esp_camera.h"
#include "Base64.h"

const char* SERIAL_NUMBER = "0000000001";

bool DEBUG;
bool SETTINGS_CHANGE;

bool streaming = false;
//-----AP Settings
const char* apSSID = "ESP32-CAM-Setup";
const char* apPassword = "12345678";
//-----Wi-Fi Settings
String ssid = "";
String password = "";
//-----MQTT Settings
String mqtt_server = "";
unsigned int mqtt_port = 0;
String mqtt_login = "";
String mqtt_password = "";

//const char* mqtt_server = "broker.mqttdashboard.com";
//const unsigned int mqtt_port = 1883;

//#define MQTT_USER ""
//#define MQTT_PASSWORD ""

```

```
//-----TOPICS
#define MQTT_PUBLISH_TOPIC "camera/test"
#define MQTT_SUBSCRIBE_TOPIC "camera/test/get"
#define MQTT_SUBSCRIBE_TOPIC_CHASSIS "robot/keyboard"
#define MQTT_SUBSCRIBE_TOPIC_FLASH "camera/flash"
#define MQTT_SUBSCRIBE_TOPIC_QUANTITY "camera/quality"
#define MQTT_SUBSCRIBE_TOPIC_SETTINGS_WIFI "settings/wifi"
#define MQTT_SUBSCRIBE_TOPIC_SETTINGS_MQTT "settings/mqtt"

WiFiClient espClient;

PubSubClient client(espClient);

WebServer server(80);

//--ESP32-CAM CAMERA MODULE Settings
#define PWDN_GPIO_NUM      32
#define RESET_GPIO_NUM    -1
#define XCLK_GPIO_NUM      0
#define SIOD_GPIO_NUM     26
#define SIOC_GPIO_NUM     27
#define Y9_GPIO_NUM       35
#define Y8_GPIO_NUM       34
#define Y7_GPIO_NUM       39
#define Y6_GPIO_NUM       36
#define Y5_GPIO_NUM       21
#define Y4_GPIO_NUM       19
#define Y3_GPIO_NUM       18
#define Y2_GPIO_NUM        5
#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM     23
#define PCLK_GPIO_NUM     22

//-----Motor pins
```

```

#define motor1_IN 1 // - 14 init ?      / IN2
#define motor1_PWM 15 // - 2 init LOW   / IN1-PWM
#define motor2_IN 13 // - 1 Tx         / IN4
#define motor2_PWM 12 // - 15 init - HIGH / IN3 - PWM

// motor default settings
int motorSpeed = 5;
int mspeed = 0;

//-----Servo pins
#define SERVO_V_PIN 2 // - 12 init LOW
#define SERVO_H_PIN 14 // - 13 init ?

Servo servo_v;
Servo servo_h;

// servo default settings
int servo_v_pos = 90;
int servo_h_pos = 90;
//-----FLASH Settings
#define FLASH_PIN 4
#define LEDC_CHANNEL 0
//-----RED LED Pins
#define LED_PIN 33
#define BTN_PIN 3 //3

//----Custom command parameter values
String command = "";
byte receiveState = 0;
byte cmdState = 1;
byte pState = 1;
byte questionState = 0;
byte equalState = 0;
byte semicolonState = 0;
String feedback = "";

```

```
unsigned long previousMillis = 0; // will store last time LED was
updated

unsigned long previousWiFiMillis = 0;
unsigned long previousMQTTMillis = 0;
const long wifiInterval = 5000;
const long mqttInterval = 5000;

// constants won't change:
const long interval = 150;

//-----BUTTON values
const int buttonPin = 3; // GPIO 3 для кнопки

unsigned long buttonPressStartTime = 0;
int pressSequence[11];
int pressIndex = 0;
bool buttonPressed = false;
bool changeSettingsMode = false;
unsigned long lastDebounceTime = 0;
const unsigned long debounceDelay = 50;
int lastButtonState = HIGH;
int buttonState = HIGH;

unsigned long buttonPressStart = 0;
bool buttonHeld = false;
const int buttonHoldTime = 5000;

void setup() {
    pinMode(buttonPin, INPUT_PULLUP);

    pinMode(LED_PIN, OUTPUT);

    //INIT FLASH(GPIO4)
```

```
ledcSetup(LED_CHANNEL, 5000, 8);
ledcAttachPin(FLASH_PIN, LED_CHANNEL);

loadCredentials();

if (DEBUG) {
  Serial.begin(115200);
}

loadCredentials();
checkButtonOnStartup();

logger("Load login data... %s | %s", ssid.c_str(),
password.c_str());

if ((ssid.length() > 0 && password.length() > 0) &&
!SETTINGS_CHANGE) {
  logger("Using saved credentials: %s | %s", ssid.c_str(),
password.c_str());

  WRITE_PERI_REG(RTC_CNTL_BROWN_OUT_REG, 0); //Turn off the
power and restart it if the power is unstable

  randomSeed(micros());

  initCamera();
  initWiFi();
  client.setServer(mqtt_server.c_str(), mqtt_port);
  client.setCallback(callback);
  client.setBufferSize(2048);
  if (!DEBUG) {
    initMotor();
    initServo();
  }
}
```

```

    } else if (SETTINGS_CHANGE || (ssid.length() == 0 &&
password.length() == 0)){
        createWiFiPoint();
    }
}

void loop() {
// loadCredentials();
    btnControl();

    unsigned long currentMillis = millis();

    if (currentMillis - previousWiFiMillis >= wifiInterval) {
        previousWiFiMillis = currentMillis;
        if (!SETTINGS_CHANGE && (ssid.length() != 0 &&
password.length() != 0)){
            checkWiFi();
        }
    }

    if (currentMillis - previousMQTTMillis >= mqttInterval) {
        previousMQTTMillis = currentMillis;
        if (!SETTINGS_CHANGE && (ssid.length() != 0 &&
password.length() != 0)){
            reconnect();
        }
    }

    if (SETTINGS_CHANGE || (ssid.length() == 0 && password.length()
== 0)) {
        server.handleClient();
    } else {
        if (client.connected() && streaming) {
            unsigned long currentMillis = millis();
            feedback = sendImage();

```

```

        sendText(feedback);
    }
    client.loop();
}

btnControl();
}

void handleRoot() {
    String content = "<!DOCTYPE html><html lang='ru'>";
    content += "<head><meta charset='utf-8'>";
    content += "<meta name='viewport' content='width=device-width,
initial-scale=1, shrink-to-fit=no'>";

    content += "<style>";
    content += "body { font-family: 'Arial', sans-serif; }";
    content += ".container { max-width: 600px; margin: 0 auto;
padding: 20px; }";
    content += ".form-group { margin-bottom: 1.5rem; }";
    content += "input { width: 100%; padding: 0.5rem; margin-top:
0.5rem; }";
    content += "button { padding: 0.5rem 2rem; background-color:
#007bff; color: white; border: none; }";
    content += "</style>";

    content += "<title>Settings Wi-Fi & MQTT</title></head><body>";
    content += "<div class='container'><h1 class='mt-5'>Settings for
robot <span class='serial-number'>#" + String(SERIAL_NUMBER) +
"</span></h1>";
    content += "<h3 class='mt-5'>Settings Wi-Fi</h3>";
    content += "<form class='mt-4' action='/save' method='post'>";

    // Форма Wi-Fi
    content += "<div class='form-group'><label for='wifi-ssid'>SSID
(Wi-Fi name):</label>";

```

```

    content += "<input type='text' class='form-control' id='wifi-
ssid' name='wifi-ssid' value='" + ssid + "'></div>";
    content += "<div class='form-group'><label for='wifi-
password'>Password:</label>";
    content += "<input type='password' class='form-control'
id='wifi-password' name='wifi-password' value='" + password +
"'></div>";

    // Форма MQTT
    content += "<h3 class='mt-5'>Settings MQTT</h3>";
    content += "<div class='form-group'><label for='mqtt-
server'>Server:</label>";
    content += "<input type='text' class='form-control' id='mqtt-
server' name='mqtt-server' value='" + mqtt_server + "'></div>";
    content += "<div class='form-group'><label for='mqtt-
port'>Port:</label>";
    content += "<input type='text' class='form-control' id='mqtt-
port' name='mqtt-port' value='" + String(mqtt_port) + "'></div>";
    content += "<div class='form-group'><label for='mqtt-
login'>Login (Username):</label>";
    content += "<input type='text' class='form-control' id='mqtt-
login' name='mqtt-login' value='" + mqtt_login + "'></div>";
    content += "<div class='form-group'><label for='mqtt-
password'>Password:</label>";
    content += "<input type='password' class='form-control'
id='mqtt-password' name='mqtt-password' value='" + mqtt_password
+ "'></div>";

    content += "<button type='submit' class='btn btn-primary'>SAVE
SETTINGS</button>";
    content += "</form></div>";
    content += "</body></html>";

    server.send(200, "text/html", content);
}

```

```
void handleSave() {
    String new_ssid = server.arg("wifi-ssid");
    String new_password = server.arg("wifi-password");
    String new_mqtt_server = server.arg("mqtt-server");
    String new_mqtt_port = server.arg("mqtt-port");
    String new_mqtt_login = server.arg("mqtt-login");
    String new_mqtt_password = server.arg("mqtt-password");

    if (new_ssid != "") ssid = new_ssid;
    if (new_password != "") password = new_password;
    if (new_mqtt_server != "") mqtt_server = new_mqtt_server;
    if (new_mqtt_port != "") mqtt_port = new_mqtt_port.toInt();
    if (new_mqtt_login != "") mqtt_login = new_mqtt_login;
    if (new_mqtt_password != "") mqtt_password = new_mqtt_password;

    saveCredentials();

    server.send(200, "text/html", "Data saved!");
    delay(1000);
    SETTINGS_CHANGE = true;
    settingsControl();
}

void btnControl(){
    int reading = digitalRead(buttonPin);
    if (reading != lastButtonState) {
        lastDebounceTime = millis();
    }

    if ((millis() - lastDebounceTime) > debounceDelay) {
        if (reading != buttonState) {
            buttonState = reading;
            if (buttonState == LOW) {
                if (!buttonPressed) {
```

```

        buttonPressed = true;
        buttonPressStartTime = millis();
    }
} else {
    if (buttonPressed) {
        unsigned long pressDuration = millis() -
buttonPressStartTime;
        buttonPressed = false;

        if (pressDuration < 1000) {
            handleShortPress();
        } else if (pressDuration >= 2000 && pressDuration <
10000) {
            handleLongPress();
        } else if (pressDuration >= 20000) {
            resetToFactorySettings();
            resetPressState();
        }
    }
}
}
}
lastButtonState = reading;
}
void handleShortPress() {
    if (pressIndex < 11) {
        pressSequence[pressIndex++] = 0;
        loger("Short press detected. %i", pressIndex);
        checkCombination();
    }
}
void handleLongPress() {
    if (pressIndex < 11) {
        pressSequence[pressIndex++] = 1;
        loger("Long press detected. %i", pressIndex);
    }
}

```

```

        checkCombination();
    }
}

void checkCombination() {
    // Комбінація для режиму відладки: ". . . . . _ . . . . ."
    int debugModePattern[11] = {0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0};

    // Комбінація для режиму зміни налаштувань: ". . . _"
    int settingsModePattern[4] = {0, 0, 0, 1};
    if (pressIndex == 11 && comparePatterns(pressSequence,
debugModePattern, 11)) {
        enterDebugMode();
        resetPressState();
        return;
    }
    if (pressIndex == 4 && comparePatterns(pressSequence,
settingsModePattern, 4)) {
        enterSettingsMode();
        resetPressState();
        return;
    }
    if (pressIndex >= 11) {
        resetPressState();
    }
}

bool comparePatterns(int *inputPattern, int *targetPattern, int
length) {
    for (int i = 0; i < length; i++) {
        if (inputPattern[i] != targetPattern[i]) {
            return false;
        }
    }
    return true;
}

```

```
void enterDebugMode() {
    for (int i = 0; i < 2; i++){
        for (int i = 0; i < 100; i += 5) {
            ledcWrite(LEDC_CHANNEL, i);
            delay(100);
        }
        for (int i = 100; i > 0; i -= 5) {
            ledcWrite(LEDC_CHANNEL, i);
            delay(100);
        }
    }
    debugControl();
}
```

```
void enterSettingsMode() {
    for (int i = 0; i < 100; i += 5) {
        ledcWrite(LEDC_CHANNEL, i);
        delay(100);
    }
    for (int i = 100; i < 0; i -= 5) {
        ledcWrite(LEDC_CHANNEL, i);
        delay(100);
    }
    settingsControl();
}
```

```
void resetPressState() {
    pressIndex = 0;
    for (int i = 0; i < 11; i++) {
        pressSequence[i] = -1;
    }
    logger("Resetting press state.");
}
```

```
void resetToFactorySettings() {
```

```

logger("Resetting to factory settings...");
for (int i = 0; i < 10; i++) {
    ledcWrite(LEDC_CHANNEL, 55);
    delay(500);
    ledcWrite(LEDC_CHANNEL, 0);
    delay(500);
}

factoryBoot();
}

void checkButtonOnStartup() {
    int buttonState = digitalRead(buttonPin);
    if (buttonState == LOW) {
        buttonPressStart = millis();
        while (digitalRead(buttonPin) == LOW) {
            if (millis() - buttonPressStart >= buttonHoldTime) {
                SETTINGS_CHANGE = true;
                logger("Button held on startup, entering settings mode.");
                break;
            }
        }
    }
}

String sendImage() {
    camera_fb_t * fb = NULL;
    fb = esp_camera_fb_get();
    if (!fb) {
        logger("Camera capture failed");
        return "Camera capture failed";
    }

    char *input = (char *)fb->buf;
    char output[base64_enc_len(3)];

```

```

String imageFile = "data:image/jpeg;base64,";
for (int i = 0; i < fb->len; i++) {
    base64_encode(output, (input++), 3);
    if (i % 3 == 0) imageFile += String(output);
}
int fbLen = imageFile.length();

String clientId = "ESP32-";
clientId += String(random(0xffff), HEX);
if (client.connect(clientId.c_str(), mqtt_login.c_str(),
mqtt_password.c_str())) {
    client.beginPublish(MQTT_PUBLISH_TOPIC, fbLen, true);

    String str = "";
    for (size_t n = 0; n < fbLen; n = n + 2048) {
        if (n + 2048 < fbLen) {
            str = imageFile.substring(n, n + 2048);
            client.write((uint8_t*)str.c_str(), 2048);
        }
        else if (fbLen % 2048 > 0) {
            size_t remainder = fbLen % 2048;
            str = imageFile.substring(n, n + remainder);
            client.write((uint8_t*)str.c_str(), remainder);
        }
    }
    client.endPublish();
    esp_camera_fb_return(fb);
    return "";
}
esp_camera_fb_return(fb);
return "failed, rc=" + client.state();
}

void initCamera() {
    camera_config_t config;

```

```
config.ledc_channel = LEDC_CHANNEL_0;
config.ledc_timer = LEDC_TIMER_2; //0
config.pin_d0 = Y2_GPIO_NUM;
config.pin_d1 = Y3_GPIO_NUM;
config.pin_d2 = Y4_GPIO_NUM;
config.pin_d3 = Y5_GPIO_NUM;
config.pin_d4 = Y6_GPIO_NUM;
config.pin_d5 = Y7_GPIO_NUM;
config.pin_d6 = Y8_GPIO_NUM;
config.pin_d7 = Y9_GPIO_NUM;
config.pin_xclk = XCLK_GPIO_NUM;
config.pin_pclk = PCLK_GPIO_NUM;
config.pin_vsync = VSYNC_GPIO_NUM;
config.pin_href = HREF_GPIO_NUM;
config.pin_sscb_sda = SIOD_GPIO_NUM;
config.pin_sscb_scl = SIOC_GPIO_NUM;
config.pin_pwdn = PWDN_GPIO_NUM;
config.pin_reset = RESET_GPIO_NUM;
config.xclk_freq_hz = 20000000;
config.pixel_format = PIXFORMAT_JPEG;

if (psramFound()) {
    config.frame_size = FRAMESIZE_UXGA;
    config.jpeg_quality = 10;
    config.fb_count = 2;
} else {
    config.frame_size = FRAMESIZE_SVGA;
    config.jpeg_quality = 12;
    config.fb_count = 1;
}

//-----STREAM INIT
esp_err_t err = esp_camera_init(&config);
if (err != ESP_OK) {
    loger("Camera init failed with error 0x%x", err);
}
```

```

    ESP.restart();
}

//Customizable video frame default size (resolution size)
sensor_t * s = esp_camera_sensor_get();
// initial sensors are flipped vertically and colors are a bit
saturated
if (s->id.PID == OV3660_PID) {
    s->set_vflip(s, 1); // flip it back
    s->set_brightness(s, 10); // up the brightness just a bit,
DEF=1
    s->set_saturation(s, 0); // lower the saturation, DEF=-2
}
// drop down frame size for higher initial frame rate
s->set_framesize(s,    FRAMESIZE_CIF);           //Resolution
SVGA(800x600),    VGA(640x480),    CIF(400x296),    QVGA(320x240),
HQVGA(240x176),    QQVGA(160x120),    QXGA(2048x1564 for OV3660)
}

void qualityChange(String quality) {
    sensor_t *s = esp_camera_sensor_get();
    if (s != NULL) {
        if (quality == "CIF") {
            s->set_framesize(s, FRAMESIZE_CIF);
        } else if (quality == "VGA") {
            s->set_framesize(s, FRAMESIZE_VGA);
        } else if (quality == "SVGA") {
            s->set_framesize(s, FRAMESIZE_SVGA);
        } else if (quality == "QVGA") {
            s->set_framesize(s, FRAMESIZE_QVGA);
        } else if (quality == "HQVGA") {
            s->set_framesize(s, FRAMESIZE_HQVGA);
        } else if (quality == "QQVGA") {
            s->set_framesize(s, FRAMESIZE_QQVGA);
        } else {

```

```

        loger("Failed to get sensor pointer");
    }
}

void flashControl (int brightnessValue) {
    int brightness = map(brightnessValue, 0, 10, 0, 150);
    ledcWrite(LEDC_CHANNEL, brightness);
}

void debugControl(){
    DEBUG = !DEBUG;
    loger(DEBUG ? "Debug mode activated." : "Debug mode
deactivated.");
    EEPROM.begin(512);
    EEPROM.write(196, DEBUG);
    EEPROM.commit();
    EEPROM.end();
    delay(1000);
    ESP.restart();
}

void settingsControl(){
    SETTINGS_CHANGE = !SETTINGS_CHANGE;
    loger(SETTINGS_CHANGE ? "Change settings mode activated." :
"Change settings mode deactivated.");
    EEPROM.begin(512);
    EEPROM.write(197, SETTINGS_CHANGE);
    EEPROM.commit();
    EEPROM.end();
    delay(1000);
    ESP.restart();
}

void factoryBoot(){

```

```
    ssid = "";
    password = "";
    mqtt_server = "";
    mqtt_port = 0;
    mqtt_login = "";
    mqtt_password = "";
    saveCredentials();
    logger("FACTORY BOOT. Data saved");
    delay(1000);
    ESP.restart();
}

void saveCredentials() {
    logger("SSID: %s", ssid.c_str());
    logger("Pass: %s", password.c_str());
    logger("Server: %s", mqtt_server.c_str());
    logger("Port: %i", mqtt_port);
    logger("Username: %s", mqtt_login.c_str());
    logger("PassMQTT: %s", mqtt_password.c_str());
    logger("DEBUG MODE: %i", DEBUG);
    logger("SETTINGS MODE: %i", SETTINGS_CHANGE);

    EEPROM.begin(512);
    EEPROM.writeString(0, ssid);
    EEPROM.writeString(32, password);
    EEPROM.writeString(64, mqtt_server);
    EEPROM.writeString(96, String(mqtt_port));
    EEPROM.writeString(128, mqtt_login);
    EEPROM.writeString(160, mqtt_password);
    EEPROM.commit();
    EEPROM.end();
}

void loadCredentials() {
    EEPROM.begin(512);
```

```

ssid = EEPROM.readString(0);
password = EEPROM.readString(32);
mqtt_server = EEPROM.readString(64);
mqtt_port = EEPROM.readString(96).toInt();
mqtt_login = EEPROM.readString(128);
mqtt_password = EEPROM.readString(160);
DEBUG = EEPROM.read(196);
SETTINGS_CHANGE = EEPROM.read(197);
EEPROM.end();

logger("SSID: %s", ssid.c_str());
logger("Pass: %s", password.c_str());
logger("Server: %s", mqtt_server.c_str());
logger("Port: %i", mqtt_port);
logger("Username: %s", mqtt_login.c_str());
logger("PassMQTT: %s", mqtt_password.c_str());
logger("DEBUG MODE: %i", DEBUG);
logger("SETTINGS MODE: %i", SETTINGS_CHANGE);
}

void logger(const char* format, ...) {
    if (DEBUG == true) {
        char buffer[256];
        va_list args;
        va_start(args, format);
        vsnprintf(buffer, sizeof(buffer), format, args);
        va_end(args);

        Serial.println(buffer);
    }
}

void initMotor() {
    pinMode(motor1_IN, OUTPUT);
    pinMode(motor1_PWM, OUTPUT);
}

```

```
pinMode(motor2_IN, OUTPUT);
pinMode(motor2_PWM, OUTPUT);

motorControl("driveStop");
}

void motorControl(String direct) {
  int mspeed1 = 0;
  int mspeed2 = 0;
  mspeed = map(motorSpeed, 0, 10, 0, 255);
  if (direct == "driveBackward") {
    digitalWrite(motor1_IN, 1);
    digitalWrite(motor2_IN, 1);
  } else {
    digitalWrite(motor1_IN, 0);
    digitalWrite(motor2_IN, 0);
  }

  if (direct == "driveForward") {
    mspeed1 = mspeed;
    mspeed2 = mspeed1;
  } else if (direct == "driveBackward") {
    mspeed1 = 255 - mspeed;
    mspeed2 = mspeed1;
  } else if (direct == "driveStop") {
    mspeed1 = 0;
    mspeed2 = 0;
  } else if (direct == "driveLeft") {
    mspeed1 = mspeed;
    mspeed2 = mspeed * 0.2;
  } else if (direct == "driveRight") {
    mspeed1 = mspeed * 0.2;
    mspeed2 = mspeed;
  }
  analogWrite(motor1_PWM, mspeed1);
}
```

```

    analogWrite(motor2_PWM, mspeed2);
}

void callback(char* topic, byte* payload, unsigned int length) {
    String cmd = "";
    for (int i = 0; i < length; i++) {
        cmd += (char)payload[i];
    }
    executeCommand(cmd);
}

void reconnect() {
    if (!client.connected()) {
        loger("Attempting to reconnect to MQTT...");
        String clientId = "ESP32-";
        clientId += String(random(0xffff), HEX);
        if (client.connect(clientId.c_str(), mqtt_login.c_str(),
mqtt_password.c_str())) {
            loger("MQTT connected to %s %i %s %s", mqtt_server.c_str(),
mqtt_port, mqtt_login.c_str(), mqtt_password.c_str());
            client.publish(MQTT_PUBLISH_TOPIC, "Ready");
            client.subscribe(MQTT_SUBSCRIBE_TOPIC);
            client.subscribe(MQTT_SUBSCRIBE_TOPIC_CHASSIS);
            client.subscribe(MQTT_SUBSCRIBE_TOPIC_FLASH);
            client.subscribe(MQTT_SUBSCRIBE_TOPIC_QUANTITY);
            client.subscribe(MQTT_SUBSCRIBE_TOPIC_SETTINGS_WIFI);
            client.subscribe(MQTT_SUBSCRIBE_TOPIC_SETTINGS_MQTT);

            //-----3 flash led if connected MQTT
            for (int j = 0; j < 3; j++) {
                ledcWrite(LED_CHANNEL, 25);
                delay(100);
                ledcWrite(LED_CHANNEL, 0);
                delay(100);
            }
        }
    }
}

```

```

    }
} else {
    logger("MQTT not connected. RC= %s", String(client.state()));
    //-----flash led if not connected Wi-Fi
    ledcWrite(LED_CHANNEL, 25);
    delay(100);
    ledcWrite(LED_CHANNEL, 0);
    delay(100);
}
}
}

//-----MQTT TEXT SENDER
void sendText(String text) {
    String clientId = "ESP32-";
    clientId += String(random(0xffff), HEX);
    if (client.connect(clientId.c_str(), mqtt_login.c_str(),
mqtt_password.c_str())) {
        client.publish(MQTT_PUBLISH_TOPIC, text.c_str());
    } else {
        logger("failed, rc= %s", String(client.state()));
    }
}

//-----PROCESSING MQTT COMMAND
void executeCommand(String cmd) {
    logger(cmd.c_str());
    if (cmd.startsWith("tilt")) {
        if (DEBUG != true) {
            servoControl(cmd);
        }
    } else if (cmd.startsWith("drive")) {
        if (DEBUG != true) {
            motorControl(cmd);
        }
    }
}

```

```

} else if (cmd.startsWith("speed:")) {
    String speedStr = cmd.substring(6);
    motorSpeed = speedStr.toInt();
} else if (cmd.startsWith("quality:")) {
    String qualityStr = cmd.substring(8);
    qualityChange(qualityStr);
} else if (cmd.startsWith("brightness:")) {
    String brightnessStr = cmd.substring(11);
    int brightness = brightnessStr.toInt();
    flashControl(brightness);
} else if (cmd == "start_streaming"){
    streaming = true;
} else if (cmd == "stop_streaming"){
    streaming = false;
}
}

void initServo() {
    ESP32PWM::allocateTimer(2);
    ESP32PWM::allocateTimer(3);
    servo_v.setPeriodHertz(50);
    servo_h.setPeriodHertz(50);
    servo_v.attach(SERVO_V_PIN, 500, 2400);
    servo_h.attach(SERVO_H_PIN, 500, 2400);

    servoControl("tiltCenter");
}

void servoControl(String direct) {
    if (direct == "tiltUp" && servo_v_pos > 20) {
        servo_v_pos = servo_v_pos - 10;
    } else if (direct == "tiltDown" && servo_v_pos < 150) {
        servo_v_pos = servo_v_pos + 10;
    } else if (direct == "tiltLeft" && servo_h_pos < 150) {
        servo_h_pos = servo_h_pos + 10;
    }
}

```

```

} else if (direct == "tiltRight" && servo_h_pos > 20) {
    servo_h_pos = servo_h_pos - 10;
} else if (direct == "tiltCenter") {
    servo_v_pos = 80;
    servo_h_pos = 90;
}
servo_v.write(servo_v_pos);
servo_h.write(servo_h_pos);
delay(50);
}

void initWiFi() {
    WiFi.mode(WIFI_STA);

    for (int i = 0; i < 5; i++) {
        WiFi.begin(ssid, password);
        delay(1000);
        loger("Connecting to %s", ssid.c_str());

        long int StartTime = millis();

        while (WiFi.status() != WL_CONNECTED) {
            delay(500);
            if ((StartTime + 5000) < millis()) break;
        }

        if (WiFi.status() == WL_CONNECTED) {
            loger("STAIP                address:                %s",
WiFi.localIP().toString().c_str());
            //-----red led if connected Wi-Fi
            digitalWrite(LED_PIN, LOW);
        }
        break;
    }
}

```

```

//-----5 red led if not connected
if (WiFi.status() != WL_CONNECTED) {
  for (int k = 0; k < 5; k++) {
    digitalWrite(LED_PIN, LOW);
    delay(100);
    digitalWrite(LED_PIN, HIGH);
    delay(100);
  }
}

void checkWiFi() {
  if (WiFi.status() != WL_CONNECTED) {
    loger("WiFi disconnected. Attempting to reconnect...");

    digitalWrite(LED_PIN, LOW);
    delay(100);
    digitalWrite(LED_PIN, HIGH);
    WiFi.disconnect();
    WiFi.begin(ssid, password);
  } else {
    static bool wasConnected = false;
    if (!wasConnected) {
      loger("WiFi          connected.          STA          IP:          %s",
WiFi.localIP().toString().c_str());
      wasConnected = true;
      //-----red led if connected Wi-Fi
      digitalWrite(LED_PIN, LOW);
    }
  }
}

void createWiFiPoint(){
  if (WiFi.status() == WL_CONNECTED){

```

```

    WiFi.disconnect();
}

WiFi.softAP(apSSID, apPassword);
logger("Access Point created");
//-----5 red and flash led if not connected
for (int k = 0; k < 6; k++) {
    digitalWrite(LED_PIN, LOW);
    ledcWrite(LEDC_CHANNEL, 10);
    delay(500);
    digitalWrite(LED_PIN, HIGH);
    ledcWrite(LEDC_CHANNEL, 0);
    delay(500);
}
server.on("/", HTTP_GET, handleRoot);
server.on("/save", HTTP_POST, handleSave);
server.begin();
}

```

E.2 – Лістинг коду програми керування мобільним роботом models.py

```

from django.contrib.auth.models import AbstractUser
from django.core.validators import RegexValidator
from django.db import models

from remoteapp.utils import decrypt_data

class CustomUser(AbstractUser):

    def __str__(self):
        return self.username

    class Meta:
        verbose_name = 'User'
        verbose_name_plural = 'Users'

```

```

class Robot(models.Model):
    serial_number = models.CharField(max_length=10, unique=True,
                                     validators=[
                                         RegexValidator(
                                             regex=r'^\d{10}$',
                                             message='The serial
number must contain 10 digits.',
                                         ),
                                     ],
                                     )
    name = models.CharField(max_length=45, null=True, blank=True)
    user = models.ForeignKey(CustomUser,
on_delete=models.CASCADE, related_name='robots')
    register_date = models.DateTimeField(auto_now_add=True)
    default_selected = models.BooleanField(default=False,
blank=True)

    def __str__(self):
        return self.name

    def save(self, *args, **kwargs):
        user_robots = Robot.objects.filter(user=self.user)

        if not self.pk and user_robots.count() == 0:
            self.default_selected = True
        elif user_robots.count() == 1:
            self.default_selected = True
        if self.default_selected:
            user_robots.update(default_selected=False)

        super(Robot, self).save(*args, **kwargs)

class Meta:
    verbose_name = 'Robot'

```

```

verbose_name_plural = 'Robots'

class Photo(models.Model):
    image = models.ImageField(upload_to='photos/%Y/%m/%d/')
    user = models.ForeignKey(CustomUser,
on_delete=models.CASCADE, related_name='photos')
    created_at = models.DateTimeField(auto_now_add=True)

    def __str__(self):
        return f"Photo by {self.user} at {self.created_at}" if
self.image else "No image"

    class Meta:
        verbose_name = 'Photo'
        verbose_name_plural = 'Photos'

class Mqtt(models.Model):
    mqtt_server = models.CharField(max_length=255)
    mqtt_port = models.CharField(max_length=4,
                                validators=[
                                    RegexValidator(
                                        regex=r'^\d{4}$',
                                        message='The port number
must be 4 digits long.',
                                    ),
                                ],
                                )
    mqtt_login = models.CharField(max_length=255, null=True,
blank=True)
    mqtt_password = models.TextField(null=True, blank=True) #
(max_length=255, null=True, blank=True)
    # robot = models.ForeignKey(Robot, on_delete=models.CASCADE,
related_name='mqttts')

```

```

robot = models.OneToOneField(Robot, on_delete=models.CASCADE,
related_name='mqttts')

def __str__(self):
    return self.mqtt_server if self.mqtt_server else "No
Server"

def get_password(self):
    return decrypt_data(self.mqtt_password)

def as_dict(self):
    return {
        "id": self.pk,
        "mqtt_server": self.mqtt_server,
        "mqtt_port": int(self.mqtt_port),
        "mqtt_login": self.mqtt_login,
        "mqtt_password": self.get_password()
    }

class Meta:
    verbose_name = 'MQTT setting'
    verbose_name_plural = 'MQTT settings'

```

Е.3 – Лістинг коду програми керування мобільним роботом views.py

```

import json
import base64
from django.core.exceptions import ObjectDoesNotExist
from django.core.paginator import Paginator
from django.http import HttpResponse, JsonResponse,
HttpResponseForbidden
from django.shortcuts import render, redirect, get_object_or_404

from remoteapp.forms import *
from django.contrib.auth import login, logout
from django.contrib.auth.decorators import login_required

```

```
from django.views.decorators.csrf import csrf_exempt
from django.core.files.base import ContentFile

def register(request):
    if request.method == 'POST':
        form = CustomUserCreationForm(request.POST)
        if form.is_valid():
            user = form.save()
            login(request, user)
            return redirect('add_new_robot')
    else:
        form = CustomUserCreationForm()
    return render(request, 'remoteapp/registration_page.html',
{'form': form})

def user_login(request):
    if request.method == 'POST':
        form = UserLoginForm(data=request.POST)
        if form.is_valid():
            user = form.get_user()
            login(request, user)
            return redirect('select_default_robot')
    else:
        form = UserLoginForm()
    return render(request, 'remoteapp/autorization_page.html',
{'form': form})

@login_required
def user_logout(request):
    logout(request)
    return redirect('login')

@login_required
def select_default_robot(request):
```

```

try:
    select_robot = Robot.objects.get(user=request.user,
default_selected=True)
except ObjectDoesNotExist:
    return redirect('add_new_robot')
else:
    return redirect('home', select_robot.pk)

@login_required
def select_robot_in_root(request, robot_id):
    user_agent = request.user_agent
    if user_agent.is_mobile:
        return redirect('index_mobile', robot_id)
    else:
        return redirect('home', robot_id)

@login_required
def index(request, robot_id):
    user_agent = request.user_agent
    if user_agent.is_mobile:
        return redirect('index_mobile', robot_id)
    else:
        mqtt_settings =
Mqtt.objects.get(robot__user=request.user, robot=robot_id)
        robots = Robot.objects.filter(user=request.user)
        select_robot = Robot.objects.get(pk=robot_id)
        if mqtt_settings:
            mqtt_data = mqtt_settings.as_dict()
        else:
            mqtt_data = {}

        if not request.user.is_authenticated:
            return redirect('login')
        else:

```

```

        return render(request,
            'remoteapp/index_desktop.html',
            {'mqtt_data_json':
                json.dumps(mqtt_data),
                'robots': robots, 'select_robot':
                select_robot}))

@login_required
def index_mobile(request, robot_id):
    user_agent = request.user_agent
    if not user_agent.is_mobile:
        return redirect('home', robot_id)
    else:
        try:
            mqtt_settings = Mqtt.objects.get(robot__user=request.user, robot=robot_id)
        except ObjectDoesNotExist:
            return redirect('add_mqtt_connection', robot_id)
        robots = Robot.objects.filter(user=request.user)
        select_robot = Robot.objects.get(pk=robot_id)
        if not select_robot:
            return redirect('add_new_robot')
        if mqtt_settings:
            mqtt_data = mqtt_settings.as_dict()
        else:
            mqtt_data = {}
        if not request.user.is_authenticated:
            return redirect('login')
        else:
            return render(request, 'remoteapp/index_mobile.html',
                {'mqtt_data_json':
                    json.dumps(mqtt_data),
                    'robots': robots, 'select_robot':
                    select_robot}))

```

```
@login_required
def settings(request):
    return render(request, 'remoteapp/settings_page.html')

@login_required
def edit_personal_info(request):
    user = request.user
    if request.method == 'POST':
        form = PersonalInfoForm(request.POST, instance=user)
        if form.is_valid():
            form.save()
            return redirect('settings')
    else:
        form = PersonalInfoForm(instance=user)
    return render(request, 'remoteapp/edit_personal_info.html',
{'form': form})

@login_required
def mqtt_settings(request):
    user = request.user
    robots = Robot.objects.filter(user=user)
    if robots.count() == 1:
        robot = robots.get(user=user)
        return redirect('robot_mqtt_settings', robot.id)
    if robots.count() == 0:
        return redirect('add_new_robot')
    return render(request, 'remoteapp/mqtt_settings.html',
{'robots': robots})

@login_required
def robot_mqtt_settings(request, robot_id):
    user = request.user
    robot = Robot.objects.get(id=robot_id)
    mqttts = Mqtt.objects.filter(robot=robot, robot__user=user)
```

```

    if robot.user != request.user:
        return HttpResponseRedirect("You don't have permission to
edit this robot.")
    if mqttts.count() == 1:
        mqtt = mqttts.get(robot=robot)
        return redirect('edit_mqtt_settings', robot_id, mqtt.id)
    if mqttts.count() == 0:
        return redirect('add_mqtt_connection', robot_id)
    return render(request, 'remoteapp/robot_mqtt_settings.html',
{'mqttts': mqttts, 'robot': robot})

```

```
@login_required
```

```

def edit_mqtt_settings(request, robot_id, mqtt_id):
    mqtt = Mqtt.objects.get(id=mqtt_id, robot=robot_id)
    robot = Robot.objects.get(id=robot_id)
    if robot.user != request.user:
        return HttpResponseRedirect("You don't have permission to
edit this robot.")
    if request.method == 'POST':
        form = ConnectMqttForm(request.POST, instance=mqtt)
        if form.is_valid():
            form.save()
    else:
        form = ConnectMqttForm(instance=mqtt,
initial={'mqtt_password': mqtt.get_password()})
    return render(request, 'remoteapp/edit_mqtt_settings.html',
{'form': form, 'title': f'Edit MQTT:', 'robot':
robot, 'mqtt': mqtt})

```

```
@login_required
```

```

def delete_mqtt_connection(request, mqtt_id):
    mqtt = Mqtt.objects.get(id=mqtt_id)
    if mqtt.robot.user != request.user:
        return HttpResponseRedirect("You don't have permission to
edit this robot.")

```

```

mqtt.delete()
return redirect('mqtt_settings')

@login_required
def add_mqtt_connection(request, robot_id):
    robot = Robot.objects.get(id=robot_id)
    if robot.user != request.user:
        return HttpResponseForbidden("You don't have permission to
edit this robot.")
    if request.method == 'POST':
        form = ConnectMqttForm(request.POST)
        if form.is_valid():
            wifi_connection = form.save(commit=False)
            wifi_connection.robot = robot
            wifi_connection.save()
            return redirect('robot_mqtt_settings', robot_id)
        else:
            form = ConnectMqttForm()
            return render(request, 'remoteapp/edit_mqtt_settings.html',
                {'form': form, 'title': f'Add new MQTT
connection', 'robot': robot})

@login_required
def robot_settings(request):
    user = request.user
    robots = Robot.objects.filter(user=user)
    if robots.count() == 1:
        robot = robots.get(user=user)
        return redirect('edit_robot_settings', robot.id)
    if robots.count() == 0:
        return redirect('add_new_robot')
    return render(request, 'remoteapp/robot_settings.html',
        {'robots': robots})

@login_required

```

```

def edit_robot_settings(request, robot_id):
    robot = Robot.objects.get(id=robot_id)
    if robot.user != request.user:
        return HttpResponseRedirect("You don't have permission to
edit this robot.")
    if request.method == 'POST':
        form = EditRobotForm(request.POST, instance=robot)
        if form.is_valid():
            form.save()
            return redirect('settings')
    else:
        form = EditRobotForm(instance=robot)
    return render(request, 'remoteapp/edit_robot_settings.html',
                  {'form': form, 'title': f'Edit robot:
{robot.name}', 'robot': robot})

@login_required
def add_new_robot(request):
    user = request.user
    if request.method == 'POST':
        form = ConnectRobotForm(request.POST)
        if form.is_valid():
            robot = form.save(commit=False)
            robot.user = user
            robot.save()
            return redirect('settings')
    else:
        form = ConnectRobotForm()
    return render(request, 'remoteapp/edit_robot_settings.html',
                  {'form': form, 'title': f'Add new robot'})

@csrf_exempt
def save_photo(request):
    if request.method == 'POST':
        image_data = request.POST.get('image_src')

```

```

user = request.user
if image_data and user.is_authenticated:
    format, imgstr = image_data.split(';base64,')
    ext = format.split('/')[1]
    data = ContentFile(base64.b64decode(imgstr),
name=f"{user.username}_{user.id}.{ext}")
    photo = Photo(user=user, image=data)
    photo.save()
    return JsonResponse({'status': 'success', 'message':
'Photo saved successfully!'})
    return JsonResponse({'status': 'error', 'message':
'Invalid data'})
    return JsonResponse({'status': 'error', 'message': 'Only POST
requests are allowed'})

@login_required
def photo_gallery(request):
    photos = Photo.objects.filter(user=request.user).order_by('-
created_at')
    if request.user_agent.is_mobile:
        paginator = Paginator(photos, 10)
    else:
        paginator = Paginator(photos, 6)
    page_number = request.GET.get('page')
    page_obj = paginator.get_page(page_number)
    return render(request, 'remoteapp/gallery_page.html',
{'page_obj': page_obj})

@login_required
def delete_photo(request, photo_id):
    photo = get_object_or_404(Photo, id=photo_id,
user=request.user)
    photo.delete()
    return redirect('photo_gallery')

```

E.4 – Лістинг коду програми керування мобільним роботом urls.py

```

from .views import *

urlpatterns = [
    path('register/', register, name='register'),
    path('', user_login, name='login'),
    path('logout/', user_logout, name='logout'),
    path('rotate_device/',
TemplateView.as_view(template_name="remoteapp/rotate_device.html
"), name='rotate_device'),
    path('index/<int:robot_id>/', index, name='home'),
    path('index_mobile/', select_default_robot,
name='select_default_robot'),
    path('index_mobile/<int:robot_id>/', index_mobile,
name='index_mobile'),
    path('select_default_robot/', select_default_robot,
name='select_default_robot'),
    path('select_robot_in_root/<int:robot_id>/',
select_robot_in_root, name='select_robot_in_root'),
    path('settings/', settings, name='settings'),
    path('settings/personal-information/', edit_personal_info,
name='edit_personal_info'),
    path('settings/mqtt-settings/', mqtt_settings,
name='mqtt_settings'),
    path('settings/mqtt-settings/<int:robot_id>/',
robot_mqtt_settings, name='robot_mqtt_settings'),
    path('settings/mqtt-settings/add-mqtt-for/<int:robot_id>/',
add_mqtt_connection, name='add_mqtt_connection'),
    path('settings/wifi-settings/delete-mqtt/<int:mqtt_id>/',
delete_mqtt_connection, name='delete_mqtt_connection'),
    path('settings/mqtt-settings/<int:robot_id>/<int:mqtt_id>/',
edit_mqtt_settings, name='edit_mqtt_settings'),
    path('settings/robot-settings/', robot_settings,
name='robot_settings'),

```

```

    path('settings/robot-settings/<int:robot_id>/',
edit_robot_settings, name='edit_robot_settings'),
    path('settings/robot-settings/add-robot/',          add_new_robot,
name='add_new_robot'),
    path('save-photo/', save_photo, name='save_photo'),
    path('settings/photo_gallery/',                    photo_gallery,
name='photo_gallery'),
    path('delete_photo/<int:photo_id>/',              delete_photo,
name='delete_photo'),
]

```

E.5 – Лістинг JavaScript коду програми керування мобільним роботом

```

var          keyboardTopic          =          "robot/keyboard";
var          flashTopic             =          "camera/flash";
var          qualityTopic           =          "camera/quality";
var  mqttDataElement  =  document.getElementById('mqtt-data');
var  mqttData        =  JSON.parse(mqttDataElement.textContent);
console.log(mqttData);
var          mqttServer             =          mqttData.mqtt_server;
var          mqttPort              =          mqttData.mqtt_port;
var          mqttLogin             =          mqttData.mqtt_login;
var          mqttPassword          =          mqttData.mqtt_password;
var client = new Paho.MQTT.Client(mqttServer, mqttPort, "clientId"
+          new          Date().getTime());

client.onConnectionLost = function (responseObject) {
    if (responseObject.errorCode !== 0) {
        console.log("Connection          lost:",
responseObject.errorMessage);
    }
};

client.onMessageArrived = function (message) {
    var          imageData          =          message.payloadString;
    if (imageData !== undefined && imageData.length > 0) {

```

```

        $("#streamedImage").attr("src",          imageData);
        var          isChecked          =          $(".stream-ctrl
input[type='checkbox']").prop("checked");
        if          (!isChecked)          {
            $("#streamedImage").attr("src",
"https://img.freepik.com/premium-vector/no-signal-old-tv-
screen_268834-925.jpg");
        }
    }
};

client.connect({
    onSuccess:          function          ()          {
        console.log(mqttData);
        console.log("Connected          to          MQTT          server");
        client.subscribe("camera/test");
    }
});

function          sendCommand(topic,          command)          {
    var          message          =          new          Paho.MQTT.Message(command);
    message.destinationName          =          topic;
    client.send(message);
}

//          Control          buttons
$("#upButton").mousedown(function () { sendCommand(keyboardTopic,
"tiltUp");          });
$("#leftButton").mousedown(function          ()          {
sendCommand(keyboardTopic,          "tiltLeft");          });
$("#centerButton").mousedown(function          ()          {
sendCommand(keyboardTopic,          "center");          });
$("#rightButton").mousedown(function          ()          {
sendCommand(keyboardTopic,          "tiltRight");          });
$("#downButton").mousedown(function          ()          {

```

```

sendCommand(keyboardTopic,          "tiltDown");          });

$("#forwardDriveButton").mousedown(function          ()          {
sendCommand(keyboardTopic,          "driveForward");          });
$("#leftDriveButton").mousedown(function          ()          {
sendCommand(keyboardTopic,          "driveLeft");          });
$("#stopDriveButton").mousedown(function          ()          {
sendCommand(keyboardTopic,          "driveStop");          });
$("#rightDriveButton").mousedown(function          ()          {
sendCommand(keyboardTopic,          "driveRight");          });
$("#backwardDriveButton").mousedown(function          ()          {
sendCommand(keyboardTopic,          "driveBackward");          });

//          Stop          driving          when          mouse          up
$(".remote-ctrls          .btn").mouseup(function          ()          {
sendCommand(keyboardTopic,          "driveStop");          });

//          Quality          dropdown
$("#qualityDropdown          a").click(function          ()          {
    var          quality          =          $(this).text();
    sendCommand(qualityTopic,          "quality:"          +          quality);
});

//          Speed          and          brightness          sliders
$("#motor-speed").change(function          ()          {
    var          speed          =          $(this).val();
    sendCommand(keyboardTopic,          "speed:"          +          speed);
});

$("#flash-brightness").change(function          ()          {
    var          brightness          =          $(this).val();
    sendCommand(flashTopic,          "brightness:"          +          brightness);
});

//          Streaming          control          switch

```

```

$(".stream-ctrl input[type='checkbox']").change(function () {
    var isChecked = $(this).prop("checked");
    var command = isChecked ? "start_streaming" :
"stop_streaming";
    sendCommand("camera/test/get", command);
});

var keyPressed = {}; // Object to track pressed keys

$(document).keydown(function (e) {
    if (!keyPressed[e.which]) { // Check if the key was not already
pressed
        keyPressed[e.which] = true; // Set the key press flag
        switch (e.which) {
            case 87: // W
                sendCommand(keyboardTopic, "driveForward");
                break;
            case 38: // Up arrow
                sendCommand(keyboardTopic, "tiltUp");
                break;
            case 65: // A
                sendCommand(keyboardTopic, "driveLeft");
                break;
            case 37: // Left arrow
                sendCommand(keyboardTopic, "tiltLeft");
                break;
            case 83: // S
                sendCommand(keyboardTopic, "driveBackward");
                break;
            case 40: // Down arrow
                sendCommand(keyboardTopic, "tiltDown");
                break;
            case 68: // D
                sendCommand(keyboardTopic, "driveRight");
                break;
        }
    }
});

```

```
        case 39: // Right arrow
            sendCommand(keyboardTopic, "tiltRight");
            break;
    }
}
}).keyup(function (e) {
    keyPressed[e.which] = false; // Reset the key press flag
    switch (e.which) {
        case 87: // W
            sendCommand(keyboardTopic, "driveStop");
            break;
        case 65: // A
            sendCommand(keyboardTopic, "driveStop");
            break;
        case 83: // S
            sendCommand(keyboardTopic, "driveStop");
            break;
        case 68: // D
            sendCommand(keyboardTopic, "driveStop");
            break;
    }
});
```

ДОДАТОК Є
Презентаційний матеріал

Мета, об'єкт та предмет дослідження роботи

- ▶ Об'єкт дослідження - процес відеонагляду за виробничими процесами на приладобудівних підприємствах.
- ▶ Предмет дослідження - система відеонагляду за перебігом виробничих процесів на приладобудівних підприємствах.
- ▶ Мета дослідження - підвищення ефективності відеонагляду за перебігом виробничих процесів на приладобудівних підприємствах шляхом розроблення мобільної системи відеонагляду на базі ESP32-CAM.

2

Постановка задачі

Для виконання поставленої мети необхідно:

- провести аналіз вже існуючих рішень мобільних систем відеонагляду;
- провести підбір необхідних компонентів для побудови мобільного робота;
- провести аналіз мов програмування та технологій, що можуть бути використані для реалізації поставленої мети;
- налагодити програмне забезпечення для передачі відеопотоку з камери мобільної платформи у користувацький інтерфейс;
- налагодити програмне забезпечення для дистанційного керування мобільною платформою, використовуючи мережу Інтернет.

3

Обґрунтування вибору основних компонентів мобільної системи відеоспостереження



4

Розрахунок параметрів джерела живлення

Забезпечення необхідного джерела живлення є важливою задачею при створенні мобільних платформ, бо кожен елемент має своє енергоспоживання.

Для оптимальної швидкості роботи електродвигунів необхідна напруга 12В, тому джерело живлення повинно бути 12В, для цього пропонується зібрати акумуляторний блок, що складається з трьох акумуляторів типу 18650 підключених послідовно, а також матиме модуль захисту АКБ та модуль заряду.

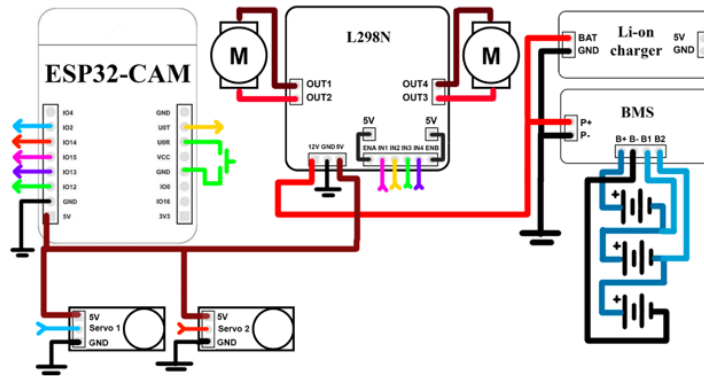
$$U_{\text{вих}} = U_{\text{ак}} \cdot n, \quad U_{\text{вих}} = 4,2 \cdot 3 = 12,6 \text{ В.}$$

$$I_{\text{max}} = \sum_{n=1} I_{\text{max}n} \quad I_{\text{max}} = 0,31 + 2 + 0,8 \cdot 2 = 3,91 \text{ А}$$



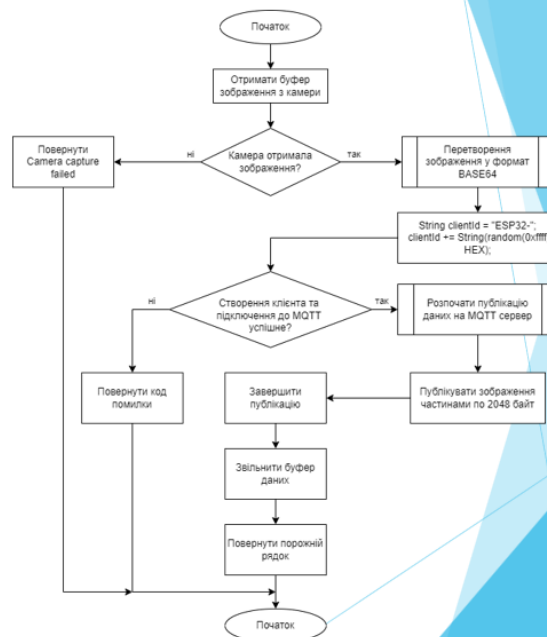
5

Загальна принципова схема підключення компонентів МР



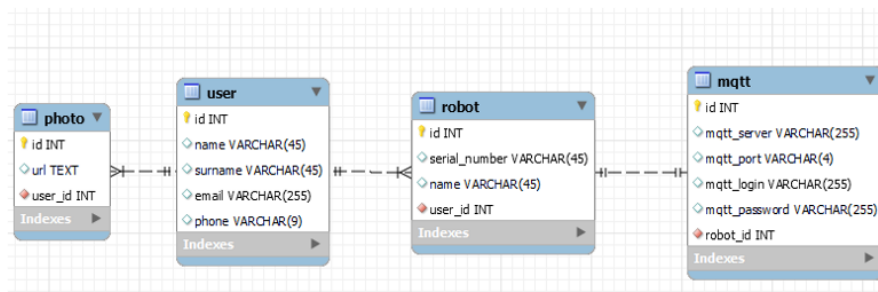
6

Схема алгоритму перетворення та відправки зображень на MQTT сервер



10

ER-діаграма бази даних для веб-застосунку мобільної системи відеонагляду



12

Вигляд головної сторінки користувацького інтерфейсу



13

ЕКСПЕРИМЕНТАЛЬНА ЧАСТИНА

Темою дослідження є порівняння розробленої системи мобільного відеонагляду на базі ESP32-CAM з аналогами.

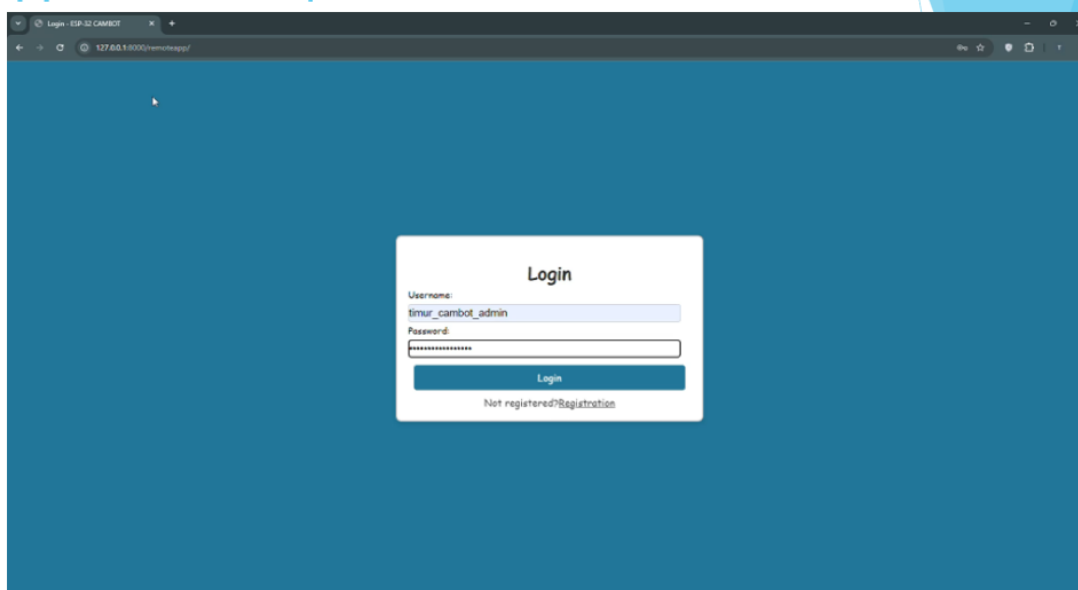
Робоча гіпотеза: Ефективне використання мікроконтролера для передачі потокового відео та керування мобільною платформою можливе за допомогою програмного забезпечення та оптимальних налаштувань.

У результаті проведення експериментів з порівняння аналогічних систем на базі ESP32-CAM та Raspberry Pi 3 Model B було визначено, що:

- час автономної роботи майже не відрізняється;
- дальність стабільної передачі відеопотоку більша у розробленій МСВ у середньому на 3,5 метри;
- зручність та універсальність інтерфейсу керування розробленої МСВ краща, ніж у аналогічній системі;
- вартість розробленої системи менша на 3231 гривню, ніж вартість аналогічної системи.

14

ДЕМОНСТРАЦІЯ РОБОТИ



15

ВИСНОВКИ

- ▶ У роботі розглянуто процес розроблення мобільної роботизованої системи відеонагляду за перебігом виробничих процесів на приладобудівних підприємствах. Проведено аналіз сучасних рішень, обґрунтовано вибір апаратних компонентів: плати керування, драйверів двигунів, джерела живлення. Визначено оптимальну конструкцію триколісного робота, що забезпечує стійкість руху.
- ▶ Розроблено програмне забезпечення для передачі та отримання відеопотоку, дистанційного керування через Wi-Fi та MQTT. Створено алгоритми роботи мобільного робота та клієнтської програми з можливістю керування через веб-інтерфейс.
- ▶ Експериментально доведено переваги системи над аналогами за показниками автономності, стабільності відеопотоку, зручності інтерфейсу та вартості.
- ▶ Результати дослідження можуть бути застосовні в побутових та промислових охоронних системах. Подальший розвиток можливий шляхом додавання датчиків для моніторингу середовища та шляхом вдосконалення компонентів МР.
- ▶ Розробка відповідає Цілі сталого розвитку 9.4.

16

Висвітлення результатів роботи та апробація

- ▶ Вибір обладнання для розробки мобільного робота для відеонагляду.
URL: https://tapr.nure.ua/wp-content/uploads/2023/11/zbimik-aded2023_2.pdf.
- ▶ Роль розпізнавання образів та комп'ютерного зору в удосконаленні робототехнічних систем підтримки рішень.
URL: <https://openarchive.nure.ua/server/api/core/bitstreams/4f761402-f21e-48b5-8f2b-f0df59491a86/content>
- ▶ Основні етапи розроблення наземного мобільного робота.
URL: https://tapr.nure.ua/wp-content/uploads/2024/10/zbimik_mms2024_work_compressed.pdf
- ▶ Розроблення веб-сторінки керування мобільним роботом через протокол MQTT. Матеріали І Всеукраїнської конференції «Комп'ютерно-інтегрованих технологій, автоматизації та робототехніки» (Computer-integrated technologies, automation and robotics) CITAR`24 16-17 травня 2024

17

ДОДАТОК Ж

Відомість кваліфікаційної роботи

