

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Чорноморський національний університет

імені Петра Могили

Факультет комп'ютерних наук

Кафедра комп'ютерної інженерії

ДОПУЩЕНО ДО ЗАХИСТУ

Завідувач кафедри,

д-р техн. наук, проф.

_____ І. М. Журавська

« __ » _____ 2024 р.

КВАЛІФІКАЦІЙНА БАКАЛАВРСЬКА РОБОТА

**Апаратно-програмний модуль візуального
спостереження**

Спеціальність 123 Комп'ютерна інженерія

123 – КБР.01 – 405.22010609

Студент

_____ Р. Д. Кухарчук

підпис

« __ » _____ 202__ р.

Керівник канд. техн. наук, доцент

_____ Я. М. Крайник

підпис

« __ » _____ 202__ р.

Миколаїв – 2024

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Чорноморський національний університет імені Петра Могили
Факультет комп'ютерних наук
Кафедра комп'ютерної інженерії

ЗАТВЕРДЖУЮ

Зав. кафедри _____ І. М. Журавська

« _____ » _____ 2024 р.

ЗАВДАННЯ
на виконання кваліфікаційної бакалаврської роботи

Видано студенту групи 405 факультету комп'ютерних наук

Кухарчук Ростислав Денисович

(прізвище, ім'я, по батькові студента)

1. Тема кваліфікаційної роботи

Апаратно-програмний модуль візуального спостереження

Затверджена наказом по ЧНУ ім. Петра Могили від 30.01.2024 № 17.

2. Строк представлення кваліфікаційної роботи « _____ » _____ 20__ р.

3. Очікуваний результат роботи та початкові дані, якщо такі потрібні

Для створення апаратно-програмного модуля візуального спостереження можна використати різні одноплатні комп'ютери, такі як Raspberry Pi 4, Asus Tinker Board S R2.0, NVIDIA Jetson Nano Developer kit, Odroid-XU4. Крім того, для збирання такого модуля можна використати комплектуючі, такі як камера Raspberry Pi module v2, блок живлення USB-Type C, кабель HDMI, монітор, корпус, розроблений за допомогою 3D-принтера (за бажанням).

Для розробки програмного забезпечення та обробки відео можна використовувати різні інструменти, такі як Microsoft Visual Studio Code для написання алгоритмів, Gstreamer для обробки медіа-потoku, OpenCV для розпізнавання об'єктів, SSH термінал Putty для віддаленого підключення, Fritzing для візуалізації з'єднання схеми та Cascade training GUI для навчання класифікаторів.

4. Перелік питань, що підлягають розробці

Розглянуто основні принципи візуального спостереження. Дати характеристику поширених одноплатних комп'ютерів. Навести огляд модулів камери та провести узагальнення подальших напрямків роботи.

Зробити проектування апаратно-програмного модуля візуального спостереження та реалізацію мікроконтролерної частини модуля. Описати модуль камери та допоміжні елементи, виконати обґрунтований вибір програмного забезпечення. Провести дослідження особливостей впровадження розробленого модуля візуального спостереження. Реалізувати програмну складову модуля та аналіз можливостей щодо розширення функціоналу розроблюваного модуля.

5. Перелік графічних матеріалів

Привести вигляд та схеми основних складових проекту. Виконати та навести функціональну схему та черговість підключення складових. На основі дерева цілей провести оцінку цільового спрямування проекту для широкого використання.

6. Завдання до спеціальної частини

Проаналізувати загальні відомості про системи відеоспостереження, включаючи їх значення в контексті охорони праці. Дослідити правові норми та регуляції, що стосуються використання відеоспостереження на робочих місцях. Оцінити переваги та недоліки відеоспостереження з огляду на охорону праці. Розглянути питання впровадження та управління системами відеоспостереження в Україні, зокрема вибір обладнання, розробку політики відеоспостереження та підготовку персоналу; а також встановити заходи забезпечення приватності та захисту даних, такі як обмеження щодо обробки та зберігання відеоданих.

7. Консультанти:

Консультант	Кафедра (організація)	Частина роботи
Макарова О. В. Старший викладач	кафедра екології Медичного інституту ЧНУ імені Петра Могили	Спеціальна частина з охорони праці

Керівник роботи

канд. техн. наук, доцент, Крайник Ярослав Михайлович
(посада, прізвище, ім'я, по батькові)

(підпис)

Завдання прийнято до виконання

Кухарчук Ростислав Денисович
(прізвище, ім'я, по батькові студента)

(підпис)

Дата видачі завдання « ____ » _____ 20 ____ р.

КАЛЕНДАРНИЙ ПЛАН
виконання кваліфікаційної роботи

Тема: Апаратно-програмний модуль візуального спостереження

№	Найменування роботи	Початок	Закінчення	Примітки
1	Розробка та затвердження завдання на виконання КР	11.12.2023	12.12.2023	Виконав
2	Огляд літератури за темою роботи	15.01.2024	18.02.2024	Виконав
3	Складання календарного плану БКР	19.02.2024	06.03.2024	Виконав
4	Аналіз предметної області	19.02.2024	04.03.2024	Виконав
5	Розробка проєктних рішень	23.02.2024	09.03.2024	Виконав
6	Моделювання та конструювання АПЗ	20.02.2024	27.02.2024	Виконав
7	Перевірка працездатності, тестування та апробація розробленого АПЗ, аналіз результатів тестування	01.03.2024	04.03.2024	Виконав
8	Відгук керівника КР	02.06.2024	07.06.2024	Виконав
9	Оформлення БКР та презентації	15.05.2024	03.06.2024	Виконав
10	Попередній захист	28.05.2024	04.06.2024	Виконав
11	Завершення оформлення КР та презентації	04.06.2024	05.06.2024	Виконав
12	Рецензування	05.06.2024	07.06.2024	Виконав
13	Захист бакалаврської кваліфікаційної роботи	.06.2024	.06.2024	Виконав

Розробив здобувач ВО Кухарчук Ростислав Денисович
(прізвище, ім'я, по батькові) (підпис)

« ____ » _____ 20__ р.

Керівник роботи Крайник Ярослав Михайлович
(посада, прізвище, ім'я, по батькові) (підпис)

« ____ » _____ 20__ р.

АНОТАЦІЯ

до кваліфікаційної бакалаврської роботи

«Апаратно-програмний модуль візуального спостереження»

Студент 405 гр.: Кухарчук Ростислав Денисович

Керівник: канд. техн. наук, доцент, Крайник Ярослав Михайлович

Ця робота присвячена розробці апаратно-програмного модулю візуального спостереження для застосування в різних сферах, включаючи контроль за громадським порядком, безпекою на дорогах та захистом майна. Головною метою проєкту є створення ефективної системи, яка забезпечує високу якість відеофіксації подій із застосуванням сучасних технологій розпізнавання облич, нічної зйомки та інтеграції з мережею для реального часу передачі даних.

Цей модуль може використовуватися для забезпечення безпеки в громадських місцях, моніторингу транспортних потоків, а також для особистих потреб, таких як відеонагляд за приватною власністю.

Результатом роботи є функціональний прототип апаратно-програмного модулю візуального спостереження, який включає в себе оптимізовану мікроконтролерну частину та програмне забезпечення для ефективного збору, обробки та передачі відеоданих.

Пояснювальна записка бакалаврської роботи складається зі вступу, трьох розділів, висновків та трьох додатків. У вступі визначається актуальність теми, сформульовані мета, об'єкт, предмет та завдання дослідження та розроблення бакалаврської роботи. У першому розділі проведено аналіз функціональних можливостей комп'ютерних платформ у сфері візуального спостереження. У другому розділі описано проєктування апаратно-програмного модуля візуального спостереження. В третьому розділі етапи дослідження особливостей впровадження розробленого модуля візуального спостереження. У висновках наведено аналіз виконаної роботи та отриманих результатів дослідження та розроблення. У додатку А наведено Довідку з перевірки на унікальність. У додатку Б – Діаграми послідовностей. У додатку В – Код програми покадрової обробки зображення при візуальному спостереженні.

В цілому, бакалаврська робота без додатків містить 85 с., 40 рис., 13 табл., 35 джерела посилання.

Ключові слова: візуальне спостереження; система аналізу даних, PYTHON, відеосистема, мікроконтролерна частина.

ABSTRACT

of the Bachelor's thesis

"Hardware and software module for visual surveillance "

Student: Kukharchuk Rostislav Denisovich

Supervisor: D. in Engineering, Associate Professor, Kraynik Yaroslav Mykhailovych

This work is dedicated to the development of a hardware and software visual surveillance module for use in various areas, including public order control, road safety, and property protection. The main goal of the project is to create an effective system that provides high quality video recording of events using modern face recognition technologies, night photography, and network integration for real-time data transmission.

This module can be used to ensure security in public places, monitor traffic flows, and for personal needs such as video surveillance of private property.

The result of the work is a functional prototype of a hardware and software visual surveillance module, which includes an optimized microcontroller part and software for efficient collection, processing and transmission of video data.

The explanatory note of the bachelor's thesis consists of an introduction, three chapters, conclusions, and three appendices. The introduction defines the relevance of the topic, formulates the purpose, object, subject, and objectives of the research and development of the bachelor's thesis. The first chapter analyzes the functionality of computer platforms in the field of visual surveillance. The second section describes the design of the hardware and software module for visual surveillance. The third section describes the stages of studying the features of the implementation of the developed visual surveillance module. The conclusions provide an analysis of the work performed and the results of research and development. Appendix A contains the Certificate of Uniqueness Verification. Appendix B - Sequence diagrams. Appendix C contains the code of the program for frame-by-frame image processing during visual observation.

In total, the bachelor's thesis without appendices contains 85 pages, 40 figures, 13 tables, 35 references.

Keywords: *visual surveillance; data analysis system, PYTHON, video system, microcontroller part.*

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ.....	4
ВСТУП.....	5
1 АНАЛІЗ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ КОМП'ЮТЕРНИХ ПЛАТФОРМ У СФЕРІ ВІЗУАЛЬНОГО СПОСТЕРЕЖЕННЯ.....	8
1.1 Основні принципи візуального спостереження	8
1.2 Характеристика поширених одноплатних комп'ютерів	10
1.3 Системи та технології виявлення ознак засинання водіїв для запобігання ДТП	23
1.4 Узагальнення подальших напрямків роботи	29
Висновки до розділу 1.....	29
2 ПРОЄКТУВАННЯ АПАРАТНО-ПРОГРАМНОГО МОДУЛЯ ВІЗУАЛЬНОГО СПОСТЕРЕЖЕННЯ.....	31
2.1 Реалізація мікроконтролерної частини модуля.....	31
2.2 Модуль камери та допоміжні елементи.....	32
2.3 Вибір програмного забезпечення	34
2.4 Операційна система та драйвери	40
Висновки до розділу 2.....	42
3 ДОСЛІДЖЕННЯ ОСОБЛИВОСТЕЙ ВПРОВАДЖЕННЯ РОЗРОБЛЕНОГО МОДУЛЯ ВІЗУАЛЬНОГО СПОСТЕРЕЖЕННЯ.....	44
3.1 Програмна складова модуля	44
3.2 Аналіз можливостей щодо розширення функціоналу розроблюваного модуля	71
3.3 Аналіз енергоефективності пристрою	75
Висновки до розділу 3.....	76
ВИСНОВКИ.....	78
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	80
ДОДАТОК А Довідка про перевірку на унікальність пояснювальної записки.....	85

ДОДАТОК В Код програми покадрової обробки зображення при візуальному спостереженні 87

ПЕРЕЛІК СКОРОЧЕНЬ

ВК	– відеокарта
КБ	– кваліфікаційна бакалаврська робота
ММ	– математичне моделювання
МС	– мікропроцесор
ПЗ	– програмне забезпечення
ПТ	– програмний тренінг
СО	– система охорони
ЦП	– центральний процесор
DVR	– Digital Video Recorder
GPIO	– General Purpose Input/Output
HDMI	– High-Definition Multimedia Interface
LPDDR3	– Low Power Double Data Rate 3
NVR	– Network Video Recorder
RJ	– Registered Jack
SD	– Secure Digital
USB	– Universal Serial Bus
VSC	– Visual Studio Code
Wi-Fi	– Wireless Fidelity

ВСТУП

Наразі системи візуального спостереження стали неодмінною частиною нашого повсякденного життя. Вони застосовуються в різних галузях для нагляду за тим, що відбувається перед камерами.

Електронні пристрої активно використовуються для відеофіксації подій на стадіонах, спортивних майданчиках, вулицях, парках, транспортних магістралях, аеропортах та вокзалах. Інформаційні технології застосовуються для забезпечення громадського порядку, контролю безпеки на дорогах, захисту майна та інших цілей. Якщо раніше системи відеоспостереження були характерні переважно для великих міст, то сьогодні вони стають все більш поширеними й у малих населених пунктах. Камери встановлюються органами влади на різних рівнях, місцевими органами самоврядування та приватними особами [5].

Актуальність дослідження. Сучасні системи відеоспостереження мають ряд інтелектуальних функцій, таких як розпізнавання облич, нічна зйомка, виявлення вторгнень тощо. Вони можуть базуватися як на звичайних відеокамерах, так і на IP-камерах з різними додатковими можливостями, наприклад, Acusense та ColorVu від Hikvision. Крім того, популярність отримують панорамні камери з розширеним кутом огляду, деякі з яких здатні знімати на всі 360°, надаючи можливість отримати повний огляд навколишнього середовища. Також збільшуються можливості відеореєстраторів, зокрема, передача відеоданих в реальному часі через Інтернет на інші пристрої, включаючи смартфони.

Питання використання систем відеоспостереження потребує розгляду в широкому контексті, оскільки реакція суспільства на ці технології може відрізнитися в залежності від культурних, політичних і соціальних умов кожної країни. Іноземний досвід може вплинути на сприйняття подібних систем українським суспільством.

Суспільна реакція на переваги та ризики використання інтелектуальних систем відеофіксації часто виявляється у вигляді невдоволення контролем за особистими діями і виражається у формі опору. Наприклад, під час протестів у Гонконгу у 2019 році демонстранти активно протистояли системам відеоспостереження, знешкоджуючи камери для розпізнавання обличчя. Вони вважали, що ці пристрої використовуються владою для збору даних про активістів та передачі їх до правоохоронних служб для подальшого переслідування [3].

Це лише один із прикладів того, як суспільство реагує на зростаючу роль відеоспостереження і як воно реагує на можливість порушення приватності та обмеження особистих свобод. Реакція може варіюватися в залежності від конкретного контексту та потреб національного законодавства та громадського діалогу. Представлене дослідження присвячене розробці апаратно-програмного модулю візуального спостереження.

Мета дослідження. Основною метою даного дослідження є розробка апаратно-програмного модулю візуального спостереження.

Об'єктом дослідження є системи відеоспостереження.

Предметом дослідження є сукупність необхідних умов, що забезпечують найкращий підхід до розробки апаратно-програмного модулю візуального спостереження.

Для досягнення поставленої мети дослідження, необхідно вирішити **наступні завдання:**

- перелічити основні принципи візуального спостереження та дати характеристику найбільш поширених одноплатних комп'ютерів;
- провести огляд модулів варіантів використання камер та визначити найбільш перспективні напрямки подальших досліджень;
- виконати проєктування апаратно-програмного модуля візуального спостереження та описати черговість реалізація мікроконтролерної частини модуля;

- описати необхідне комплектування для реалізації проєкту та здійснити збір системи;
- провести тестування апаратно-програмного модулю візуального спостереження на основі одноплатних комп'ютерів;
- запропонувати шляхи розширення функціоналу розроблюваного модуля.

Методи дослідження в роботі використані такі: Пошук за наявною методичною та науковою літературою, аналіз знайдених матеріалів, порівняння, класифікація, проєктування, теоретичне моделювання, з'ясування причинно-наслідкових зв'язків, аналіз документації та результатів діяльності дослідників з проблеми, що вивчається.

У роботі також застосовано такі методи: аналітичний – використання математичних методів для обробки фактичних даних; логічний метод, який застосовується протягом всього дослідження, включаючи дедукцію, класифікацію матеріалу, формулювання та вирішення проблеми.

Теоретична та практична цінність роботи полягає в наявності відібраного теоретичного матеріалу з теми дослідження та систематизації матеріалів. Проведене дослідження забезпечує глибше опрацювання напрямку порівняно з попередніми дослідженнями вчених, дисертантів та дослідників у цій галузі.

Структура роботи. Відповідно до мети та завдань дослідження структура роботи містить вступ, три розділи, висновки, списки використаних джерел та додатків. За час роботи опрацьовано 35 літературних джерел.

Джерельна база дослідження. Дослідження базується на аналізі науково-методичної літератури, методичних посібників, наукових статей та періодичних видань.

1 АНАЛІЗ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ КОМП'ЮТЕРНИХ ПЛАТФОРМ У СФЕРІ ВІЗУАЛЬНОГО СПОСТЕРЕЖЕННЯ

1.1 Основні принципи візуального спостереження

Візуальне спостереження в українському законодавстві розглядається як засіб виконання завдань оперативно-розшукової діяльності (пункт 11 статті 8 Закону України «Про оперативно-розшукову діяльність») та кримінального процесу (стаття 269 КПК України). Відтак, доцільно розрізняти візуальне спостереження як оперативно-розшуковий захід (ОРЗ) та як негласну слідчу (розшукову) дію (НСРД).

Сучасні системи візуального спостереження здебільшого вже не є повністю замкнутими телевізійними системами (CCTV – Closed Circuit Television). Вони стали більш відкритими, обмінюючись даними через Інтернет. Такі системи дозволяють не лише спостерігати за об'єктом і записувати отриману інформацію, але й обробляти її з використанням алгоритмів розпізнавання образів та автоматично впливати на системи управління об'єктами [4].

CCTV – це система візуального спостереження, яка складається з обмеженої кількості відеокамер, що передають відео обмеженій кількості осіб, які мають право доступу до цієї інформації. Це мережа пристроїв, які обмінюються відеоінформацією між собою без передачі даних третім особам. Першою CCTV-системою була система відеоспостереження за запуском ракет Фау – 2, розроблена німецьким інженером Уолтером Брухом.

Нові системи відеоспостереження базуються на IP-камерах та цифрових системах обробки даних (NVR) і / або серверах з відповідним програмним забезпеченням. Відеоінформацію можна переглядати за допомогою смартфонів, планшетів, ноутбуків, стаціонарних комп'ютерів, а також через хмарні технології. Для поступового переходу від застарілих аналогових систем відеоспостереження використовуються HD-технології (HD TVI або Turbo HD, HDCVI, AHD), які суттєво покращують якість відео за рахунок

використання існуючих коаксіальних кабелів. Застосування HD відеокамер та HD DVR дозволяє створювати нові системи відеоспостереження, які часто є дешевшими за повністю цифрові системи на базі IP-технологій.

Обладнання систем відеоспостереження, незалежно від їх типу (аналогова, гібридна або цифрова), включає відеокамери для збору відеоінформації, пристрої обладнання для обробки та запису (DVR, NVR, плати відеозахоплення, сервери з прикладним програмним забезпеченням), обладнання для архівування та пристрої для відображення відеоінформації (монітори). Зокрема, DVR - це аналоговий відеореєстратор, а NVR - мережевий відеореєстратор. DVR - це електронний пристрій, до якого підключаються аналогові відеокамери, а відео кодується і обробляється безпосередньо в DVR. На відміну від DVR, NVR використовує програмне забезпечення: відео спочатку кодується і обробляється мережею відеокамер, а потім передається на NVR для зберігання або віддаленого перегляду. Основна відмінність між цими двома типами відеореєстраторів полягає в типі використовуваних камер: DVR підключається до аналогових камер через коаксіальний кабель, в той час як NVR працює з IP-камерами, які передають відео через мережевий кабель. [2].

Існують також гібридні системи відеоспостереження (HVR), які поєднують функції як аналогових, так і IP-відеосистем. Гібридний відеореєстратор використовується для запису відеосигналу, що надходить від камер спостереження. Цей пристрій є невід'ємною частиною будь-якої телевізійної системи безпеки. Принцип роботи відеореєстратора полягає в стисненні отриманого сигналу та архівуванні його на носії, яким може бути жорсткий диск або знімний носій. Залежно від типу сигналу, який вони обробляють, відеореєстратори поділяються на аналогові та IP. Однією з основних характеристик будь-якого відеореєстратора є кількість відеовходів, тобто кількість камер, з якими він може працювати. У випадку з гібридним реєстратором ця характеристика не є постійною і змінюється в залежності від режиму роботи пристрою.

Дуже важливою складовою будь-якої системи відеоспостереження є середовище передачі сигналу - кабельна мережа та підсистема електроживлення. При створенні систем відеоспостереження з використанням IP-технологій кабельна система будується за принципом локальної обчислювальної мережі (LAN) і може використовувати різні фізичні носії для передачі інформації - мідну виту пару, оптоволоконний кабель або бездротову мережу Wi-Fi. Традиційно для живлення відеокамер використовувалися силові кабелі, але останнім часом популярними стали технології PoE (Power over Ethernet), коли живлення і дані передаються по одній витій парі. Відеосигнал від аналогової або HD-камери до пристрою обробки зазвичай передається по хвильоводному коаксіальному кабелю з хвильовим опором 75 Ом, наприклад, RG-75, RG-6 або RG-59. Однак найсучаснішим методом є використання витой пари з пристроями передачі відеосигналу по витій парі.

1.2 Характеристика поширених одноплатних комп'ютерів

Одноплатний комп'ютер - це платформа, яка містить мікропроцесор, пам'ять та основні периферійні пристрої, розташовані на одній платі. За обчислювальною потужністю такі пристрої прирівнюються до сучасних смартфонів, але мають низку переваг перед ними. Оскільки одноплатний комп'ютер не поміщається в компактний корпус, проблему охолодження процесора і графічних ядер можна вирішити традиційними методами, наприклад, встановивши активну або пасивну систему охолодження. Також до такого пристрою можна підключати периферійні пристрої через вбудовані порти вводу/виводу. Основною перевагою є можливість встановлення операційної системи загального призначення (найчастіше Linux), що дозволяє використовувати стандартні засоби програмування, а також криптографічні та інші системи безпеки. У цій статті ми розглянемо два популярні одноплатні комп'ютери, які активно використовуються для розробки систем аналізу даних з елементами штучного інтелекту: Raspberry Pi та NVIDIA Jetson Nano. [1].

Asus Tinker Board 2GB - це одноплатний комп'ютер з ультракомпактним форм-фактором, який забезпечує кращу в своєму класі продуктивність. Плата оснащена потужним процесором ARM Cortex-A17 з тактовою частотою 1,8 ГГц, Gigabit Ethernet, портом HDMI 2.0 для підключення моніторів і телевізорів з роздільною здатністю до 4K і підтримкою H.264, 2 ГБ оперативної пам'яті, звуковою системою 192K/24-біт, SDIO 3.0 та іншими функціями. Крім того, до плати можна підключати різноманітні комерційні модулі розширення. Загальний вигляд Asus Tinker Board 2GB показано на рисунку 1.2, а основні характеристики – у табл. 1.1.

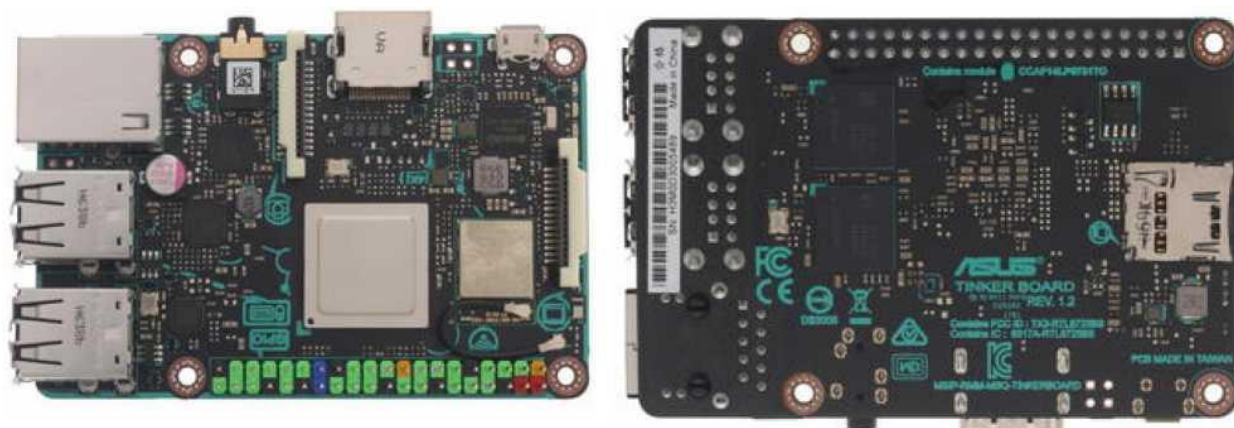


Рисунок. 1.1 – Зовнішній вигляд Asus Tinker Board 2 Гбайт

Основні характеристики ASUS Tinker Board представлені в таблиці

Таблиця 1.1 – Основні характеристики ASUS Tinker Board

Параметри	Значення
Процесор	
Виробник процесора	Rockchip
Модель процесора	Rockchip RK3288
Кількість ядер процесора	4
Частота процесора	1800 МГц
Оперативна пам'ять	
Тип оперативної пам'яті	LPDDR3
Розмір оперативної пам'яті	2 Гбайт
Накопичувачі даних	

Параметри	Значення
Об'єм вбудованої флеш-пам'яті	немає
Відеокарта	
Виробник відеочипу	Mali
Модель відеочипу	Mali T764
Обсяг відеопам'яті	виділяється з оперативної
Інтерфейси, роз'єми	
Відео інтерфейси	HDMI
Аудіо інтерфейси	3.5 мм jack (аудіо)
Інтерфейси периферії	display serial interface, camera serial interface, 40 pin
Додаткове вбудоване	Bluetooth 4.0
Особливості	HDMI 2.0 з підтримкою 4К, слот Micro SD з підтримкою
Комунікації	
Вид доступу в Інтернет	Wi-Fi, Ethernet (RJ – 45)
Швидкість мережевого	1000 Мбіт/с
Габарити (Д*Ш)	85.6 мм * 54 мм

Головними перевагами ASUS Tinker Board є таке:

- у порівнянні з іншими одноплатними комп'ютерами, такими як Orange Pi, Banana Pi тощо, Asus Tinker Board має драйвери, що підтримують апаратне прискорення відео. Підтримка OpenGL ES є значною перевагою. Простий доступ до GPIO забезпечується за допомогою WiringPi [8];
- обладнання добре збалансоване, тобто швидкодія всіх компонентів, а не лише окремих елементів, на високому рівні. Картридер забезпечує швидкість читання до 65 МБіт/с, оперативна пам'ять працює у двоканальному режимі, мережа функціонує значно швидше, ніж у Raspberry Pi. Графічний процесор забезпечує продуктивність у 80 Gflops, що майже дорівнює рівню ноутбуків 2010 року, тоді як у Raspberry Pi цей показник складає лише 10 Gflops. Процесор витримує заявлену частоту, забезпечуючи при цьому якісний звук завдяки розв'язанню з колонкою;

– завантаження стандартної операційної системи TinkerOS займає кілька секунд (з флешки зі швидкістю 65 МБіт/с). Компіляція невеликих скетчів на QT триває 2–3 секунди. Відео на YouTube в HD можна дивитися через браузер без фризів, а з додатковим охолодженням можливо і в Full HD, хоча іноді може виникати затримка. Плеєр VLC відтворює 4K відео з мінімальним FPS, як заявлено в характеристиках Asus Tinker Board.

15-контактний роз'єм DSI і CSI.

Вагомими недоліками ASUS Tinker Board є таке:

– під час роботи пристрій нагрівається до 60-70 градусів, що призводить до активації механізму захисту процесора від перегріву (дроселювання) і зниження частоти під навантаженням. Краще замінити стандартний радіатор на модуль активного охолодження;

– необхідне якісне живлення потужністю 2–3 А. Якщо кабель живлення занадто короткий, можливі самовільні перезавантаження;

– не рекомендується живити систему охолодження через GPIO коли та є активною;

– драйвери в TinkerOS Debian доволі таки "чутливі", через це оновлювати образ ОС некваліфікованими спеціалістами за допомогою apt-get не рекомендується;

– tinkeros Android потребує попередньо встановленого Google Play Store;

– підтримка API GPIO все ще знаходиться на початковій стадії розвитку.

Raspberry Pi – це мініатюрний одноплатний комп'ютер, який можна помістити на долоні. Незважаючи на свої компактні розміри, плата забезпечує достатню продуктивність, що дозволяє їй конкурувати з персональними комп'ютерами. Спочатку Raspberry Pi була розроблена як навчальний посібник з інформатики [10]. Проте ідея виявилася настільки успішною, що за кілька років цей міні-комп'ютер став популярним у широких колах. З часом Raspberry Pi пройшла кілька модифікацій, кожна з яких мала покращені

характеристики порівняно з попередніми версіями. Такий підхід дозволив регулювати вартість пристрою залежно від потреб користувачів, що позитивно вплинуло на його популярність. Вся лінійка Raspberry Pi використовує процесори з архітектурою ARM, яка зарекомендувала себе як найкраща. Плата мікрокомп'ютера Raspberry Pi (рис. 1.2) містить такі компоненти: процесор ARM, порт Ethernet, порти USB, модулі WiFi і Bluetooth, аналоговий аудіовихід, вихід HDMI, роз'єм DSI для дисплея, роз'єм CSI для камери і 40-контактні GPIO (порти вводу/виводу, що підтримують протоколи UART, I2C, SPI, PWM та інші).

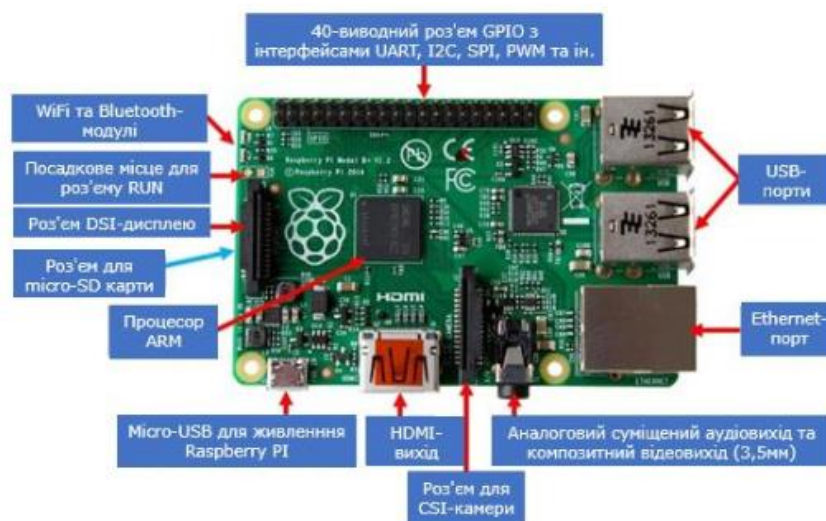


Рисунок 1.2 – Одноплатний мікрокомп'ютер Raspberry Pi

На сьогоднішній день існує 11 версій комп'ютера Raspberry Pi. Останні версії оснащені модулями бездротового зв'язку Wi-Fi і Bluetooth, що розширює застосування міні-комп'ютера в області технології Ethernet. У табл. 1.2 наведено порівняння різних версій Raspberry Pi, а також характеристики кожної моделі з деякими технічними особливостями. Як видно з таблиці, навіть наймолодша модель в лінійці має досить серйозні характеристики, враховуючи, що це одноплатний комп'ютер, більший за кредитну картку [6].

Хоча Raspberry Pi схожа на Arduino, вона функціонує зовсім по-іншому. На відміну від Arduino, ця плата працює під управлінням однієї зі спеціалізованих операційних систем, подібно до стандартного комп'ютера.

Користувачі можуть вибрати операційну систему відповідно до своїх потреб або особистих уподобань. Нижче наведено список найпопулярніших операційних систем для Raspberry Pi з коротким описом кожної.

Таблиця 1.2 – Порівняння технічних характеристик Raspberry Pi різних модифікацій

Модель	Тактова частота	К-ть ядер	Об'єм ОЗП	К-ть GPIO	К-ть USB	Підтримка Ethernet	Підтримка WiFi	Підтримка Bluetooth
B	700 МГц	1	512 Мбіт	26	2	Так	Ні	Ні
A	700 МГц	1	256 Мбіт	26	1	Ні	Ні	Ні
B+	700 МГц	1	512 Мбіт	40	4	Так	Ні	Ні
A+	700 МГц	1	256 Мбіт	40	1	Ні	Ні	Ні
2B	900 МГц	4	1 Гбайт	40	4	Так	Ні	Ні
Zero	1 ГГц	1	512 Мбіт	40	1	Ні	Ні	Ні
3B	1,2 ГГц	4	1 Гб	40	4	Так	Так	4.1
Zero W	1 ГГц	1	512 Мбіт	40	1	Ні	Так	4.0
3B+	1,4 ГГц	4	1 Гбайт	40	4	Так	Так	4.2
3A+	1,4 ГГц	4	512 Мбіт	40	1	Ні	Так	4.2
4B	1,5 ГГц	4	1, 2, 4 Гбайт	40	4	Так	Так	5.0

Raspbian - ця операційна система була представлена у 2015 році як основна операційна система для Raspberry Pi. Вона оптимізована для процесорів на базі ARM і активно розвивається. Дистрибутив містить програмне забезпечення для комп'ютерної алгебри Mathematica, модифіковану версію Minecraft PI та урізану версію браузера Chrome.

Debian - операційна система з відкритим вихідним кодом. Стандартний дистрибутив включає середовище робочого столу GNOME та низку найпопулярніших програм, таких як Firefox, LibreOffice, Evolution та інші мультимедійні інструменти. [7].

Ubuntu – ця система базується на Debian GNU / Linux. Ubuntu є найпопулярнішим дистрибутивом Linux для веб-серверів.

Fedora - це операційна система, заснована на дистрибутиві Linux від Red Hat.

Gentoo Linux - один з найпопулярніших дистрибутивів GNU/Linux, відомий своєю здатністю оптимізуватися під конкретне обладнання. Алгоритм керування пакунками дозволяє легко встановлювати його як на робочі станції, так і на сервери.

RISC OS - це операційна система, спеціально розроблена для процесорів ARM. Особливості ядра RISC OS забезпечують швидший запуск системи завдяки зберіганню даних в ПЗП, що також допомагає захистити дані в разі різних збоїв і шкідливого програмного забезпечення.

Крім перерахованих операційних систем, в Інтернеті можна знайти безліч модифікацій для найрізноманітніших потреб.

Однією з головних відмінностей Raspberry Pi від звичайних комп'ютерів є наявність програмованих портів вводу/виводу GPIO. Їх можна використовувати для керування різними пристроями та отримання телеметрії від різних датчиків. Користувач може керувати цими портами, тобто дисплеї, кнопки, датчики, реле та інші електронні модулі можна підключати до Raspberry Pi і керувати ними за власним бажанням.

Виводи GPIO виконані у вигляді дворядної смужки з контактами, розташованими на кінці плати. Ранні моделі, такі як Model A, містили 26 контактів.

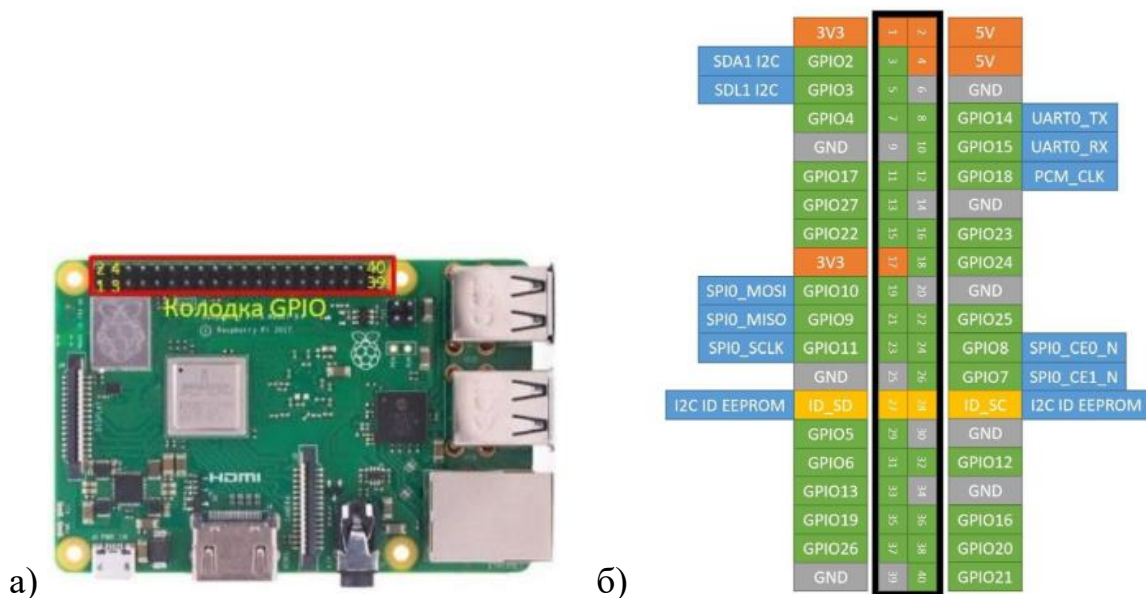


Рисунок 1.3 – Виводи GPIO плати Raspberry Pi 3B+ – а) та їх розпіновка – б)

На рисунку 1.3 а зображено зовнішній вигляд портів загального призначення для плати Raspberry Pi 3B+, а також нумерація їх виводів.

Проте для повного використання GPIO необхідно не лише знати їх нумерацію, а й розуміти, де знаходиться кожен вивід, як його називають і для чого він призначений. На рис. 1.3б) представлено повне розпінування роз'єму GPIO для модифікацій Raspberry Pi A+ та B+ і вище.

Зазвичай програми для Raspberry Pi пишуться мовою програмування Python. Особливістю цієї мови є те, що для запуску програми не потрібен компілятор. Скрипт запускається і працює відразу, але його потрібно записати у файл, а потім завантажити на плату. Ще однією важливою особливістю програмування на Python є суворе дотримання таблиць при написанні програмного коду.

Платформа Jetson Nano (рис. 1.4), як і Raspberry Pi, представлена у вигляді одноплатного комп'ютера зі схожими габаритами і великою сумісністю з периферійними пристроями. Цей пристрій спеціально розроблений для роботи з додатками, що використовують елементи штучного інтелекту.

Jetson Nano доступний у двох версіях: для розробників, виробників та аматорів, а також у вигляді готового модуля для компаній, які хочуть створювати системи для масового ринку. Цей пристрій підтримує датчики з високою роздільною здатністю, може обробляти інформацію з великої кількості сенсорів і керувати кількома нейронними мережами одночасно. Він також підтримує багато популярних фреймворків штучного інтелекту, що дозволяє розробникам інтегрувати свої улюблені моделі та фреймворки.



Рисунок 1.4 – Зовнішній вигляд платформи Jetson Nano

За словами NVIDIA, доступність недорогого одноплатного комп'ютера Jetson Nano відкриває широкі можливості для підприємств, стартапів і дослідників, які раніше не могли дозволити собі більш дорогі пристрої. Таким чином, платформа Jetson розширює свою аудиторію, а прискорювач штучного інтелекту фактично стає доступнішим. За словами Nvidia, ця плата "переносить потужність сучасного штучного інтелекту на доступну платформу, розпалюючи новий порив до інновацій серед творців, винахідників, розробників і студентів. Вони можуть реалізовувати проекти зі штучним інтелектом, які раніше були неможливими, і виводити існуючі проекти - від мобільних роботів і дронів до цифрових асистентів, автоматизованих пристроїв тощо - на новий рівень". [9].

Нижче наведені технічні характеристики платформи Jetson Nano:

- графічний процесор: Nvidia на базі архітектури Maxwell з 128 ядрами CUDA;
- процесор: чотириядерний ARM Cortex-A57 MPCore;

- відео: підтримує 4К при 30 кадрах на секунду (H.264/H.265) і 4К при 60 кадрах на секунду (H.264/H.265) для кодування і декодування відповідно;
- відеовихід: HDMI 2.0 або DP1.2, eDP 1.4, DSI (1x2), можна використовувати обидва одночасно;
- оперативна пам'ять: 4 Гбайт LPDDR4 64-біт; пропускна спроможність 25,6 ГБ/с;
- флеш-пам'ять: 16 Гбайт eMMC;
- камера: підтримка 12 ліній (3x4 або 4x2) MIPI CSI – 2 DPHY 1.1 (1,5 Гбіт/с), 12x (модуль) та 1x (набір розробника);
- роз'єми: 1x1 / 2 / 4PCIЕ, 1xUSB 3.0, 3xUSB 2.0;
- введення / виведення: 1xSDIO / 2xSPI / 6xI2C / 2xI2S / GPIO піни;
- мережа: Gigabit Ethernet;
- підтримує операційні системи: Linux для Tegra;
- розмір модуля: 69,5x45 мм;
- розмір комплекту розробника: 100x80 мм;
- підключення: 260-піновий конектор.

Однією з особливостей Jetson Nano є наявність 128-ядерного графічного процесора, що дозволяє запускати на платі високопродуктивні завдання, такі як CUDA (підтримується і встановлюється з коробки) або Tensorflow. Основний процесор має 4 ядра і заснований на технології ARM. Пам'ять розділена між CPU і GPU і становить 4 ГБ.

Ще однією важливою особливістю є сумісність з Raspberry Pi. Плата має 40-контактний роз'єм з різними інтерфейсами (I2C, SPI тощо), а також сумісний з Raspberry Pi роз'єм для камери. Це означає, що багато аксесуарів, таких як екрани та плати керування двигунами, будуть сумісні з Jetson Nano.

Третя особливість - наявність на платі 2 відеовиходів, Gigabit Ethernet і USB 3.0, що робить Jetson Nano ще більш функціональним, ніж Raspberry Pi. Живлення 5 В може подаватися як через Micro USB, так і через окремий роз'єм, що рекомендується для важких завдань. Програмне забезпечення

завантажується з SD-карти, образ якої необхідно попередньо зберегти. Загалом плата Jetson Nano складається з двох модулів - модуля Jetson Nano і нижньої плати з роз'ємами, які підключаються через спеціальний роз'єм.

Підводячи підсумок, Jetson Nano не призначений для заміни Raspberry Pi, а для завдань, які потребують великої обчислювальної потужності, вона є більш відповідною, завдяки кращим характеристикам графічного процесора, який ефективніше виконує математичні розрахунки та обробку великої кількості паралельних потоків [11].

Orange Pi Zero 256 Мбіт – це мініатюрний одноплатний комп'ютер, який використовує процесор Allwinner H2+ з чотирма обчислювальними ядрами Cortex A7 з тактовою частотою до 1,2 ГГц. Він підтримує різноманітні операційні системи, такі як Android 4.4, Ubuntu, Debian, і має 256 Мбіт оперативної пам'яті DDR3 SDRAM. Також він оснащений графічним прискорювачем Mali – 400MP2, який підтримує декодування 4К-відео формату H. 265 / HEVC з частотою кадрів до 30 кадрів на секунду і підтримує microSD карти пам'яті об'ємом до 64 гігабайт.



Рисунок 1.5 – Загальний вид Orange Pi Zero 256 Мбіт

Основні технічні характеристики Orange Pi Zero 256 Мбіт включають наступне:

- процесор: Allwinner H2+ з чотирма ядрами Cortex A7, які працюють на частоті 1,2 ГГц;

- графічний процесор: Mali400MP2 з тактовою частотою 600 МГц і підтримкою OpenGL ES2.0;
- оперативна пам'ять: 256 МБіт DDR3 RAM;
- пам'ять для програм: microSD;
- мережевий інтерфейс: порт Ethernet 10 / 100М;
- порти USB: 1x USB 2.0 та 2 порти на додатковому роз'ємі розширення. Micro USB порт служить для OTG і живлення;
- відеовихід: композитний;
- інтерфейс камери: відсутній;
- налагоджувальний послідовний порт: 3-контактний з кроком 2.54 мм;
- входи / виходи: 24-контактний і 12-контактний роз'єм, які підтримують USB, UART, SPI, I2C, PWM тощо;
- напруга живлення: 5 В;
- максимальний споживаний струм: 2 А.

LattePanda 2GB / 32GB – це одноплатний комп'ютер, що працює під управлінням операційної системи Windows 10 і має 4-ядерний процесор Intel Cherry Trail Z8350, 2 Гбайт оперативної пам'яті і вбудовану eMMC пам'ять об'ємом 32 Гбайт. Він має всю необхідну периферію для повноцінної роботи, а також може працювати з різноманітним обладнанням, таким як принтери, джойстики, камери та інше. Комп'ютер поставляється з попередньо встановленою операційною системою Windows 10 Home Edition, що дозволяє використовувати потужні інструменти, такі як Visual Studio, Nodejs, Java та інші. Однією з унікальних особливостей LattePanda є наявність інтегрованого контролера Arduino Leonardo для зв'язку з зовнішнім обладнанням. Цей одноплатний комп'ютер може знайти широке застосування в портативних системах, системах збору інформації та керування, 3D принтерах і т.д.



Рисунок 1.6 – Загальний вигляд одноплатного ПК LattePanda 2GB / 32GB

Основні характеристики LattePanda 2GB / 32GB включають:

- процесор: Intel Cherry Trail Z8350 з чотирма ядрами;
- базова частота процесора: 1,44 ГГц (максимальна – 1,92 ГГц);
- операційна система: Windows 10 Home Edition (неактивована);
- оперативна пам'ять: 2 Гбайт DDR3L;
- вбудована пам'ять eMMC: 32 Гбайт;
- графічний прискорювач: Intel HD Graphics з 12 Eus @200–500МГц;
- інтерфейси USB: USB 3.0 x 1, USB 2.0 x 2;
- бездротовий інтерфейс: Wi-Fi 802.11n 2.4G, Bluetooth 4.0;
- інтерфейс Arduino: Atmega32u4 (Arduino Leonardo);
- відео вихід: HDMI і MIPI-DSI;
- рознімання для підключення сенсорної панелі;
- мережевий інтерфейс: Ethernet 100Mbps;
- входи / виходи: GPIO процесора Intel x 6, GPIO контролера Atmega32U4 x 20;
- інтерфейсні роз'єми x 6;
- напруга живлення: 5В 2А;
- розміри плати: 88 x 70 мм;
- вага: 55 г.

1.3 Системи та технології виявлення ознак засинання водіїв для запобігання ДТП

Вибір обладнання для системи візуального спостереження залежить від конкретних потреб, таких як місце встановлення (квартира, будинок, офіс або прилегла територія) і вимоги до якості зображення. Якість зображення визначається якістю камери, включаючи об'єктив і матрицю.

Об'єктив камери визначає ширину або дальність охоплення зображення, а матриця перетворює його в відеосигнал. Розмір матриці (1 / 2.3, 1 / 2.1, 1 / 3.1, 1 / 4) впливає на якість зображення, де більший розмір матриці забезпечує вищу якість.

Камера AXIS 11225-N. 264 має високу роздільну здатність до 1 мегапікселя, можливість налаштування кадрової частоти та широкий температурний діапазон. Вона також підтримує технологію PoE та специфікації ONVIF для сумісності з різними програмними комплексами. Однак, основним недоліком є обмежена дальність вбудованого інфрачервоного підсвічування – лише 50 метрів [8].

Камера Smartec 1926–2MP має велику роздільну здатність до 2 мегапікселів і може передавати кадри з налаштуванням частоти до 5 кадрів за секунду. Вона працює в широкому температурному діапазоні від – 20 °C до +60 °C і може бути встановлена у термокожух. Однак, недоліком є відсутність можливості віддаленого управління об'єктивом і відсутність підтримки специфікації ONVIF. Крім того, дальність вбудованого інфрачервоного підсвічування обмежена 50 метрами.

Камера ArecontMisionH. 264-HDI – 356b також має високу роздільну здатність до 2 мегапікселів і можливість настройки частоти передачі кадрів. Вона працює в широкому температурному діапазоні від – 20 °C до +70 °C і може бути встановлена у термокожух. Проте ця камера також не підтримує специфікацію ONVIF і не має можливості віддаленого управління об'єктивом.

Камера A-TW105 має високу роздільну здатність 1.5 мегапікселів, що дозволяє виконувати необхідні завдання. Вона підтримує налаштування частоти передачі кадрів і працює в широкому температурному діапазоні від – 20 °С до +60 °С, що робить її придатною для різних умов експлуатації. Камера може бути встановлена у термокожух і підтримує можливість віддаленого керування об'єктивом, що спрощує процес налаштування. Проте вона не підтримує специфікацію ONVIF. Крім того, камера підтримує технологію PoE, але дальність вбудованого інфрачервоного підсвічування обмежена 100 метрами.

Камера PoivisionH. 264–3MP – 2В має високу роздільну здатність до 3 мегапікселів і можливість налаштування частоти передачі кадрів. Вона також працює в широкому температурному діапазоні від – 10 °С до +60 °С і може бути встановлена у термокожух. Крім того, камера підтримує віддалене керування об'єктивом і специфікацію ONVIF. Вона також підтримує технологію PoE, але так само, як і попередня модель, дальність вбудованого інфрачервоного підсвічування обмежена 100 метрами [13].

Камера Sony 3MP-O-H. 264-HD пропонує високу роздільну здатність 3 мегапікселів, яка відповідає потребам, що ставляться перед нею. Також можлива настройка кадрової частоти та працездатна в широкому температурному діапазоні від – 20 °С до +60 °С, з можливістю встановлення в термокожух. Крім того, забезпечена можливість віддаленого керування об'єктивом, що спрощує процес налаштування. Камера Sony 3MP-O-H. 264-HD підтримує специфікацію ONVIF і технологію PoE, але дальність вбудованого інфрачервоного підсвічування обмежена 100 метрами.

Камера Avigilon 2.0 MP-HD-H264-B також має високу роздільну здатність 2 мегапікселів і можливість настройки кадрової частоти. Вона працює в широкому температурному діапазоні і може бути встановлена в термокожух. Камера також підтримує віддалене керування об'єктивом і специфікацію ONVIF, а також технологію PoE. Дальність вбудованого інфрачервоного підсвічування обмежена 100 метрами. Особливість систем

контролю доступу Avigilon полягає в тому, що вони працюють на основі браузера і не потребують встановлення додаткового програмного забезпечення, що робить їх готовими до використання та зекономлює час.

Побудуємо зведену таблицю, в яку внесемо основні параметри розглянутих камер зовнішнього візуального спостереження (табл. 1.3)

Таблиця 1.3 – Зведена таблиця основних параметрів камер зовнішнього спостереження перефразувати текст

Марка камери	Роздільна здатність в Мпкс.	Настройка кількості кадрів в секунду	Температурний діапазон, °С.	Термок.	Управління об'єктивом	ONV 1F	Ціна, тис. грн.
AX IS1225-H. 264	1	так	від – 10 до+60	так	так	так	18000
Smartcc 1926–2MP	2	так	від – 20 до+60	так	ні	ні	19000
Arecont Mision H. 264 – HD-356B	2	так	від – 20 до+70	так	ні	ні	13000
A-TW105	1.5	так	від – 20 до+60	так	так	ні	16000
Poivisio nH. 264–3MP – 2B	3	так	від – 10 до+60	так	так	так	10000
Sony 3MP-O- H. 264 – HD	3	так	від – 20 до+60	так	так	так	12000
Avigilon 2.0MP-HD-H264-B	2	так	від – 20 до+60	так	так	так	11000

Після аналізу ми прийшли до висновку, що найкращим варіантом для ТОВ «Кур'єр сервіс» буде камера Avigilon 2.0 MP-HD-H264-B. Ця камера відповідає всім вимогам, викладеним у завданні, і не має очевидних недоліків. Розміри відеокамери Avigilon 2.0 MP-HD-H264-B можна побачити на рис. 1.7.

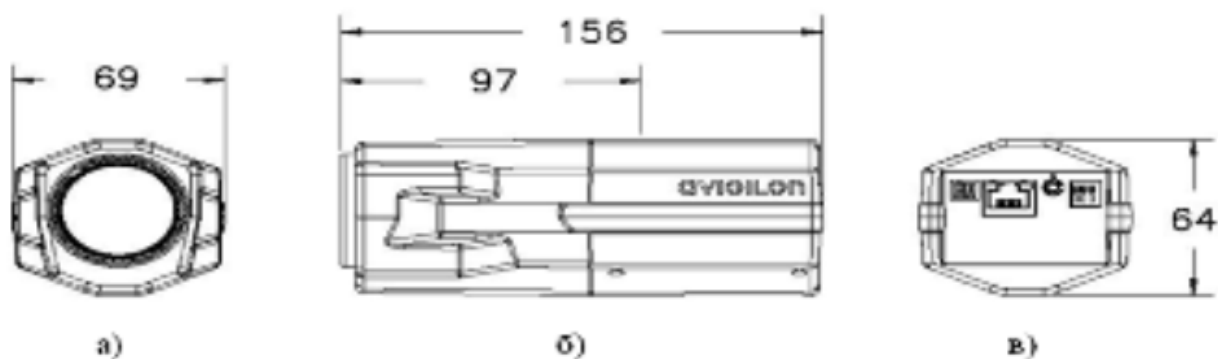


Рисунок 1.7 – Габарити відеокамери Avigillon 2.0 MP-HD-H264-B

Програмне забезпечення Avigilon для керування системами візуального спостереження та контролю доступу має інтуїтивно зрозумілий і простий у використанні інтерфейс. Avigilon Control Center (ACC) пропонує зручний інтерфейс для управління відеоматеріалами і надає повний контроль над відтворенням відео.

Оглянемо типи камер, які найбільш підходять для внутрішнього візуального спостереження. Купольні камери мають форму обтічної півсфери, що дозволяє їм зручно монтуватися на стіни або стелі без кронштейнів і легко налаштовувати об'єктиви. Мініатюрні камери зазвичай мають циліндричну або кубічну форму, їх головне перевага – компактність і вологозахисний корпус, що робить їх ідеальними для вологих середовищ, таких як автомобілки [12].

Модульні камери (безкорпусні) – це одноплатні камери з мініатюрним об'єктивом, які можуть бути вбудовані в різні предмети інтер'єру або пристрої. Корпусні камери, як правило, поставляються окремо без об'єктива і кронштейна, що дозволяє використовувати їх для різних цілей, забезпечуючи високу якість зображення і надійність.

Дані камери показали себе надійними для візуального спостереження в банках, різних установах та на підприємствах. При розгляді моделей IP-камер для внутрішнього візуального спостереження, звернемо увагу на такі моделі, як IP-камери INFINITY. Давайте розглянемо асортимент камер INFINITY серії мініатюрних захищених купольних камер SRD, які доступні в двох

модифікаціях: моделі SRD – 2000EX і SRD – 3000at, з роздільною здатністю 2 і 3 мегапікселя відповідно.

Камера SRD має компактний купольний корпус діаметром 100 мм і висотою 53 мм (рис. 1.8). Корпус камери виготовлений з металу і має клас захисту IP10. Камера є повністю герметичною: всі з'єднання, включаючи кабельний ввід, мають еластичний гумовий ущільнювач, що дозволяє підводити її під воду. Камера живиться від джерела 12VDC або за допомогою PoE. Крім мережевого роз'єму RJ – 45 і клем для підключення живлення, є аудіовхід для підключення зовнішнього активного мікрофона, аудіовихід і роз'єм для підключення тестового аналогового монітора.



Рисунок 1.8 – Камера SRD для внутрішнього відеоспостереження

Серія SRD має ширококутний об'єктив 3.6 мм (80°) з ІЧ-корекцією і вбудований ІЧ-світлодіод для додаткового підсвічування ближньої зони. Кожна камера поставляється з CD, що містить докладні інструкції для всього модельного ряду, а також утиліту для пошуку камер. Web-інтерфейс камер використовує технологію FASH для відображення відео. Модель «SRD – 2000EX має матрицю 1 / 2.8» Sony Exmor IMX 122, тоді як SRD – 3000at використовує сенсор 1 / 3 APТINA AR0331. Обидві матриці є сучасними і хорошими рішеннями. Порівнюючи ці камери, можна виділити такі особливості: SRD – 2000EX більш світлочутлива, ніж SRD – 3000at. Проте для отримання якісного зображення при недостатньому освітленні потрібна додаткова настройка параметрів через веб-інтерфейс, таких як посилення,

різкість і шумозаглушення. SRD – 3000AT менш «чутлива», але не потребує додаткових налаштувань, майже не видає шуму і формує чітке зображення.

Модель SRD – 3000at демонструє високу ефективність у вирішенні проблеми зустрічного засвічення. Навіть у випадку наявності яскравого джерела світла на кадрі та різниці у яскравості між темними та світлими ділянками, зображення відображається однаково якісно. Моделі SRD використовують процесор Texas Instruments, що успішно реалізовує ряд алгоритмів для поліпшення зображення, таких як підвищення різкості, динамічне шумозаглушення та кольорокорекція. У веб-інтерфейсі камери можна вручну змінювати параметри цих функцій, але в більшості випадків стандартні налаштування забезпечують чітке та насичене зображення.

Для внутрішнього візуального спостереження найоптимальнішим вибором буде використання мініатюрної відеокамери моделі SRD – 3000at, оскільки вона не потребує додаткових налаштувань, майже ніколи не «шумить» та надає чітке зображення. У меню цієї камери є окремий розділ WDR з рядом налаштувань відеообробки. Вибір кута огляду є важливим критерієм при виборі камери візуального спостереження, оскільки він визначає зону спостереження. Цей кут залежить від фокусної відстані об'єктива камери та формату її матриці. При виборі кута огляду слід враховувати, що об'єктив забезпечує різкість зображення в певних межах, що також обмежує розмір зони візуального спостереження.

При розробці комплексних систем візуального спостереження, які включають в себе системи відеоспостереження та контролю доступу, важливо враховувати особливості, що впливають на відмовостійкість і надійність системи. Деякі продукти можуть додавати відеоспостереження до існуючої системи безпеки, тоді як інші виступають як основа, до якої інтегруються додаткові системи, розширюючи її функціонал. Інтеграція обох систем в одне ціле можлива при створенні комплексних систем безпеки, де вони працюють як єдиний механізм. Ця інтеграція може відбуватися на різних рівнях, таких як рівень контролерів (які підключаються до сервера відеоспостереження) або

рівень серверів (коли сервер системи контролю доступу підключається до системи відеоспостереження).

1.4 Узагальнення подальших напрямків роботи

Для забезпечення ефективності системи візуального спостереження перспективним напрямком є її функціонування в режимі автоматичного розпізнавання та виділення небезпечних або підозрілих ситуацій, зменшуючи навантаження на оператора. Така система має складатися з відеокамер та програмного забезпечення, яке аналізує транслявання та виявляє потенційні загрози за допомогою відеоаналітики.

Важливою особливістю системи візуального спостереження є можливість запису відео навіть при відключенні електропостачання, що дозволяє виявити зловмисників, які намагаються вимкнути електрику. Відеокамера повинна мати акумулятор та карту пам'яті для продовження запису у разі відключення живлення. Використання акумулятора потребує ефективних алгоритмів обробки відео. Також важливо, щоб камера мала хорошу якість запису в темряві, а встановлення датчика руху допоможе виявити активність у нічний час. Система також здійснює інтелектуальну архівацію відеозаписів, зберігаючи повне відео за останню добу та окремі ділянки, де виявлено підозрілу активність. Користувачам надсилаються повідомлення про підозрілі ситуації в реальному часі.

Висновки до розділу 1

На початку роботи розглянуто основні принципи візуального спостереження та наголошено, що сучасні системи візуального спостереження здебільшого вже не є повністю замкнутими телевізійними системами. Вони стали відкритішими, обмінюючись даними через Інтернет. Такі системи дозволяють не лише спостерігати за об'єктом і записувати отриману інформацію, але й обробляти її з використанням алгоритмів

розпізнавання образів та автоматично впливати на системи управління об'єктами.

Проведено характеристику таких найбільш поширених одноплатних комп'ютерів, як Asus Tinker Board 2 Гбайт, Raspberry Pi, Jetson Nano, Orange Pi Zero 256 Мбіт, LattePanda 2GB / 32GB. Для перелічених одноплатних комп'ютерів вказано переваги та вагомні недоліки.

Окремо розглянуто внутрішні та зовнішні камери для візуального спостереження. Оцінено перспективність дослідження в даній області.

Важливим етапом подальшої реалізації є проєктування апаратно-програмного модуля візуального спостереження, що й буде виконано в наступному розділі.

2 ПРОЄКТУВАННЯ АПАРАТНО-ПРОГРАМНОГО МОДУЛЯ ВІЗУАЛЬНОГО СПОСТЕРЕЖЕННЯ

2.1 Реалізація мікроконтролерної частини модуля

Всі системи відеоспостереження, незалежно від їх типу, складаються з різних технічних компонентів, таких як блок живлення, кабелі, жорсткий диск для зберігання відеозаписів, монітор або комп'ютер для перегляду відео, а також підключення до Інтернету для використання хмарного сервісу [6].

Для створення апаратно-програмного модуля візуального спостереження можна використати різні одноплатні комп'ютери, такі як Raspberry Pi 4, Asus Tinker Board S R2.0, NVIDIA Jetson Nano Developer kit, Odroid-XU4. Крім того, для збирання такого модуля можна використати комплектуючі, такі як камера Raspberry Pi module v2, блок живлення USB-Type C, кабель HDMI, монітор, корпус, розроблений за допомогою 3D-принтера (за бажанням).



Рисунок 2.1 – Одноплатний комп'ютер Raspberry Pi 4 Model B

Для розробки програмного забезпечення та обробки відео можна використовувати різні інструменти, такі як Microsoft Visual Studio Code для написання алгоритмів, Gstreamer для обробки медіа-потоків, OpenCV для розпізнавання об'єктів, SSH термінал Putty для віддаленого підключення, Fritzing для візуалізації з'єднання схеми та Cascade training GUI для навчання класифікаторів.

Наразі одним з найпопулярніших одноплатних комп'ютерів з вбудованою підтримкою 4K та потужним процесором для обробки інформації є Raspberry Pi 4. Побудований апаратно-програмний модуль на базі Raspberry Pi 4 отримує потокове відео і обробляє кадри для виявлення об'єктів та перешкод у системах автомобільного спостереження.

Плата Raspberry Pi 4 Model B представляє собою покращену версію порівняно з попереднім поколінням Raspberry Pi 3 Model B+. У новій версії значно зросла швидкість процесора, покращена мультимедійна функціональність та обсяг пам'яті, а також оновлені порти для підключення периферійних пристроїв.

2.2 Модуль камери та допоміжні елементи

Для цього проєкту було вибрано камеру RasPi Cam v2 з сенсором Sony IMX219. Версія 2 є простішою, швидшою та кращою у багатьох аспектах. Основною перевагою тепер є можливість зйомки фотографій у роздільній здатності 8 Мрх. Це дуже важливий показник, оскільки фотографії будуть чітко відображати об'єкти і перешкоди, що потрапляють у поле зору об'єктиву, без розмиття та розпикселювання зображення.



Рисунок 2.2 – Загальний вигляд камери RasPi cam V2

Характеристики камери наступні:

- повна назва: Камера Raspberry Pi V2;
- роздільна здатність: 8 мегапікселів (до 3280x2464);

- сенсор: Sony IMX 219 PQ CMOS, 1 / 4 дюйма;
- підтримка відеоформатів: 1080p (30 кадрів за секунду), 720p (60 кадрів за секунду), 640 x 480p (90 кадрів за секунду);
- лінза: 3.3 мм;
- діафрагма: f / 2.

Raspberry Pi 4 – це малий одноплатний комп'ютер, всі його компоненти, від пам'яті до USB-портів, розміщені на одній друкованій платі без додаткових плат або аксесуарів. Розміри плати становлять 2,2 на 3,4 дюйма і приблизно 0,6 дюйма у висоту. На борту також є 4 Гбайт оперативної пам'яті і чотириядерний процесор Broadcom, який працює на частоті 1,5 ГГц. Плату також оснащено чотирма USB Type-A портами, двома відеовиходами мікро HDMI, роз'ємом гігабітного Ethernet і приймачем сигналу для 802.11ac Wi-Fi і Bluetooth 5.0.

Більшість модулів використовують 40-контактний роз'єм загального призначення (GPIO), який може жити всі підключені датчики, а також передавати та збирати дані з них. Характеристики плати Raspberry Pi 4 наведено у табл. 2.1.

Таблиця 2.1 – Характеристики плати Raspberry Pi 4

Бездротова мережа	802.11 ac; Bluetooth 5.0
Центральний процесор	Broadcom BCM2711 (Cortex-A72)
Тактова частота	1.5GHz
Пам'ять	4GB LPDDR4
Інтерфейси	40-pin header (GPIO), USB 3.0, 2.0, type-c
Камера	Sony IMX 219
Графіка	ARM VideoCore VI
Вивід відео	Micro HDMI x2
Підтримка кодеків	OpenGL ES3, h. 265, h. 264
Дротова мережа	Gigabit Ethernet

За рахунок зміни архітектури вдалося внести наступні поліпшення. Тепер контролер Ethernet має власний виділений інтерфейс, а порти USB підключені через лінію PCI Express Gen 2. Це забезпечує загальну пропускну здатність на рівні 4 Гбіт/с, що є достатнім для підтримки двох портів USB 3.0 і двох портів USB 2.0.

2.3 Вибір програмного забезпечення

Visual Studio Code об'єднує простоту текстового редактора з потужним інструментом для розробки, таким як автодоповнення коду та засоби для налагодження IntelliSense. Цей цикл редагування-збирання-налагодження, що вимагає мінімум зусиль, дозволяє швидше приступити до втілення ідей та розробки. VS Code має вбудовану підтримку для розробки на Node.js з JavaScript та TypeScript, використовуючи ті ж базові технології, що й Visual Studio. Крім того, VS Code має інструменти для роботи з веб-технологіями, такими як JSX / React, HTML, CSS, SCSS, Less та JSON.

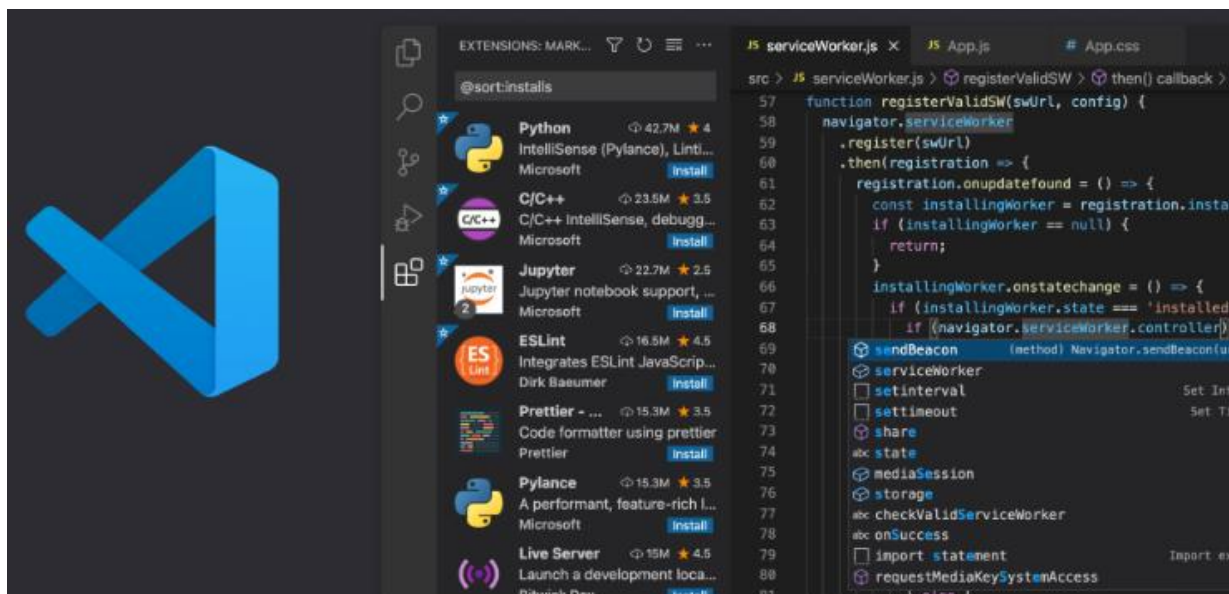


Рисунок 2.3 – Програмне середовище VS Code

Python – це потужна і легка для вивчення мова програмування. Вона надає можливість застосовувати ефективні високорівневі структури даних і пропонує простий, проте дієвий підхід до об'єктно-орієнтованого

програмування. Комбінація зрозумілого синтаксису та динамічної типізації у інтерпретованій мові робить Python ідеальним вибором для написання сценаріїв та швидкої розробки додатків у різних галузях та на більшості платформ.

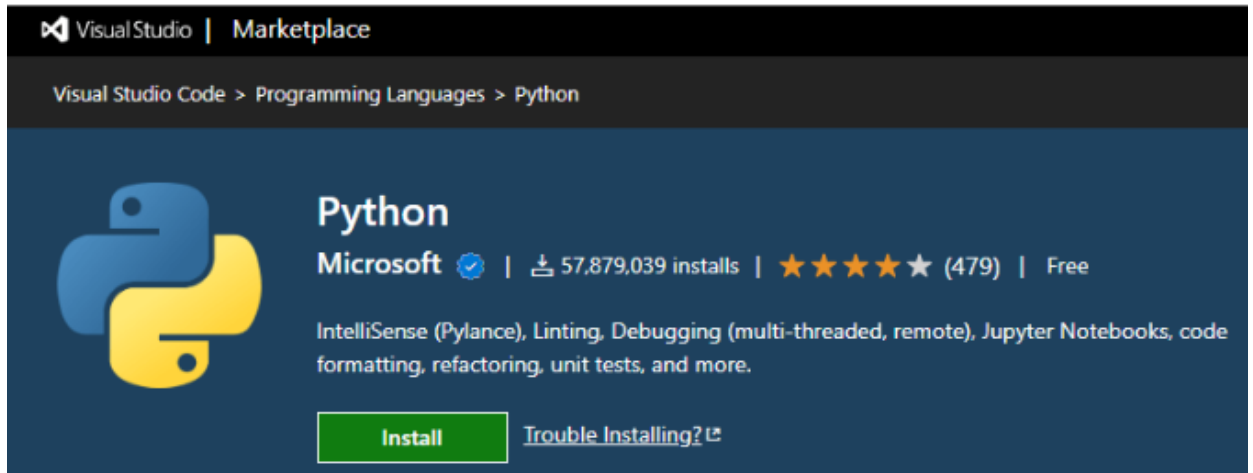


Рисунок 2.4 – Розширення для мови програмування Python

Програмування стало значно динамічнішим завдяки таким особливостям Python:

- динамічна типізація. Типи можуть бути статичними або динамічними. У випадку динамічної типізації розробнику не потрібно вказувати тип змінних – мова програмування визначає його автоматично;
- зручне повернення функцією кількох значень. Розробник може перерахувати значення через кому, і вони автоматично перетворяться на список [13];
- автоматичне управління пам'яттю. Програмісту не потрібно вручну виділяти пам'ять – це робиться автоматично Python. Це знижує трохи рівень контролю над програмою, але значно прискорює процес розробки;
- автоматичне управління сміттям. Ця можливість поліпшує ефективність використання пам'яті і значно спрощує процес розробки;
- прив'язка типу даних. У Python тип даних зв'язується зі значенням, а не змінними.

Мова програмування Python використовується для таких основних завдань:

– для аналізу даних. Аналіз даних став невід'ємним елементом в будь-якій сучасній галузі, де дані є цінним активом. Більшість компаній зацікавлені в збиранні, обробці та аналізі релевантної інформації для витягнення цінних висновків для бізнесу. Python виходить за межі інших мов програмування у цьому відношенні. Особливо цінним є той факт, що Python не лише має обширну стандартну бібліотеку, але й надає доступ до великого набору додаткових модулів, спеціально розроблених для аналітичних завдань. Найбільш відомі бібліотеки Python для аналізу даних, такі як pandas і NumPy, надають можливість очищати, аналізувати, вивчати статистику та візуалізувати тенденції у ваших даних [14];

– для візуалізації даних. Візуалізація даних є важливою частиною аналізу, яка допомагає представляти інформацію у більш зрозумілому форматі, незалежно від того, чи була вона оброблена чи очищена. Python пропонує широкий асортимент інструментів для візуалізації даних, що робить його незамінним у цій сфері. Найпопулярніші з них – matplotlib і базуючийся на ньому seaborn. Використовуючи ці інструменти, ми можемо створювати різноманітні типи візуалізацій, від простих до складних;

– для машинного навчання. Машинне навчання (ML) є ключовим компонентом багатьох завдань у науці про дані. Це галузь штучного інтелекту, що використовує алгоритми для виявлення закономірностей і тенденцій у історичних даних з метою прогнозування на основі нових даних. Завдяки методам ML ми можемо створювати моделі для точного передбачення швидкості відтоку клієнтів, оцінки ризику захворювань, визначення оптимального розташування таксі тощо. З використанням Python, модель ML можна побудувати всього за три рядки коду [16];

– для розробки програмного забезпечення. Окрім свого широкого застосування в науці про дані, Python використовується на кожному етапі розробки програмного забезпечення: від контролю версій і автоматичної

безперервної інтеграції до прототипування, відстеження помилок, тестування та обслуговування програм. Завдяки цій мові програмування можна створювати аудіо- та відеопрограми, що базуються на методах штучного інтелекту, машинного навчання, API (інтерфейсах прикладного програмування), GUI (графічних інтерфейсах) або будь-якому іншому типу програмного забезпечення;

- для веб-розробки. Хоча для створення візуальної частини вебсайту зазвичай використовуються мови HTML, CSS і JavaScript, для його серверної частини часто обирають Python. Серед великих вебсайтів і додатків, створених за допомогою цієї мови, можна відзначити Google, Facebook, Instagram, YouTube, Dropbox та Reddit;

- автоматизація задач/скриптинг. Python є чудовим інструментом для написання програм, які автоматизують різні рутинні завдання.

Інтерпретатор Python та стандартна бібліотека доступні для завантаження у вигляді вихідних кодів та бінарних файлів для різних платформ на офіційному сайті Python і можуть використовуватися без обмежень. Також на цьому сайті можна знайти різноманітні модулі від сторонніх розробників для Python, а також інструменти та додаткову документацію. Python можна легко розширити новими функціями та типами даних, написаними на C/C++ або інших мовах, що підтримують доступ з C. Також Python може використовуватися як мова розширення для інших додатків.

Основні переваги мови програмування Python включають наступне:

- зменшення обсягу коду: завдяки наявності попередньо визначених пакетів, які містять готові алгоритми, програмісти можуть уникнути написання алгоритмів з нуля. Python має функцію «перевірка коду в час написання». Це допомагає зменшити потребу в тестуванні коду;

- готові бібліотеки: у Python існує велика кількість готових бібліотек для різних областей програмування, включаючи машинне та глибоке

навчання. Встановлення цих бібліотек у проєкт здійснюється за допомогою команди `pip install`;

– кросплатформенність: Python працює на різних операційних системах, включаючи Windows, MacOS, Linux, Unix тощо. Для перенесення коду між різними платформами рекомендується використовувати інструменти, які допомагають у вирішенні проблем залежностей, наприклад, PyInstaller.

Мова програмування Go є статично типізованою та компілюється в двійкові файли машини. Вона має вбудовані масиви різної довжини та структури даних у вигляді карт ключ-значення. Крім того, вона має стандартну бібліотеку, яка підтримується [18].

Go дозволяє розробникам працювати з об'єктами, забезпечуючи статичне або суворе введення з паралельним доступом.

Важливо відзначити, що використання бібліотеки комп'ютерного зору OpenCV в мові Go відоме як GoCV. Пакет GoCV надає доступ до функцій бібліотеки OpenCV 4 і підтримує останні версії Go та OpenCV v4.7.0 для операційних систем Linux, macOS та Windows.

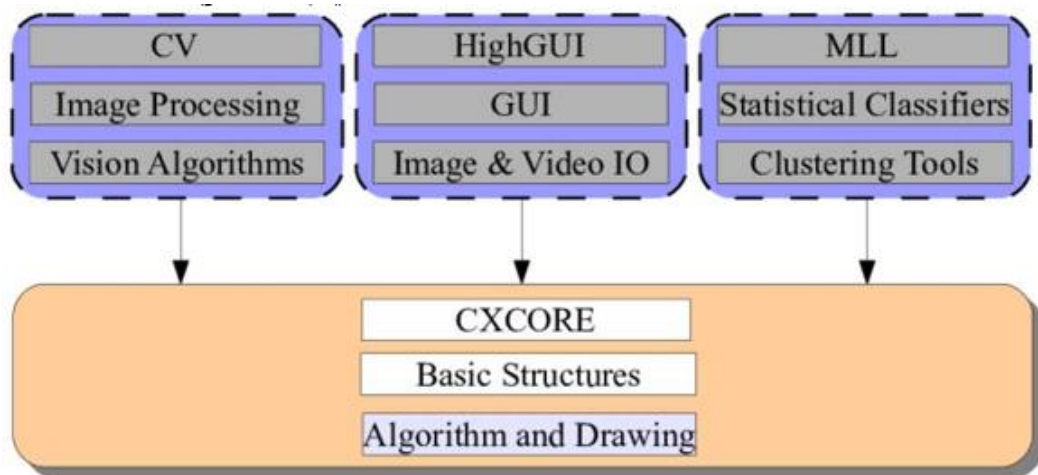


Рисунок 2.5 – Схематичне зображення архітектури OpenCV

Головна мета полягає в створенні сумісності з найновішими розробками у світі OpenCV для мови програмування Go. GoCV підтримує CUDA для використання графічних процесорів NVIDIA.

Основні бібліотечні модулі, які обробляють дані OpenCV, включають такі:

- основні функції. Цей модуль працює з основними структурами даних, такими як Scalar, Point, Range і т.д. Крім того, Mat використовується для зберігання зображення у багатовимірному масиві;
- обробка зображень. Цей модуль використовується для обробки зображень, включаючи роботу з гістограмами, геометричні перетворення, фільтрацію, зміну колірному простору і т.д.;
- відео. Модуль аналізує потік даних, відстежує рух, виявляє об'єкти та віднімає фон;
- відео I / O. Цей модуль відповідає за роботу з відеокодеками та захоплення відеопотоку;
- calib3d. Модуль містить алгоритми геометрії кількох ракурсів, оцінку позиції об'єкта, стереовідповідність та елементи 3D-реконструкції;
- виявлення об'єктів. Цей модуль відповідає за виявлення об'єктів із задалегідь визначених категорій, таких як автомобілі, знаки, світлофори, люди, обличчя тощо.

GStreamer – це мультимедійний фреймворк, що здатний обробляти різноманітні типи медіа-даних, такі як аудіо, відео, запис та потокова відеотрансляція. Цей фреймворк має модульну структуру, що дозволяє користувачам створювати власні конвеєри та інтегрувати плагіни в процес розробки додатків.

Поєднання бібліотек OpenCV і GStreamer дозволяє отримати більшу гнучкість та різноманітність у роботі з мультимедійним потоком та іншими даними. Вони можуть використовуватися як окремо, так і разом, що дозволяє використовувати різні можливості обробки мультимедійних даних.

GStreamer надає можливість маніпулювати, передавати та отримувати дані на нашій машині через різні протоколи.

2.4 Операційна система та драйвери

На Raspberry Pi є конектор із 40 пінів, який дозволяє з'єднувати різноманітні датчики, світильники, двигуни та інші пристрої. Цей конектор може мати 40 або 26 контактів, залежно від моделі, тому важливо знати, як ці виводи розташовані та позначені.

Окрім цього, на платі є додаткові сучасні конектори, які дозволяють підключати різноманітні периферійні пристрої, такі як монітори для відтворення високоякісного зображення. Також наведено можливість підключення додаткових модулів для поліпшення системи розпізнавання об'єктів та перешкод, а також для забезпечення більш зручного користування звичайним користувачем [17].

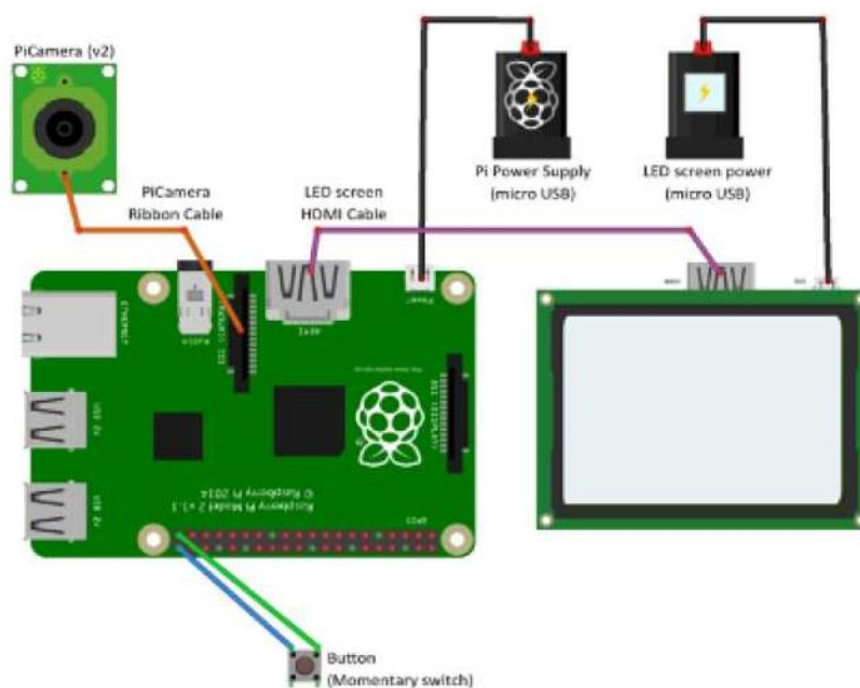


Рисунок 2.6 – Підключення периферії до Raspberry Pi 4

Ця плата має широкий спектр функцій:

- процесор Broadcom BCM2711, який включає чотири ядра Cortex-A72 (ARM v8) з тактовою частотою 1,5 ГГц;
- швидка оперативна пам'ять обсягом 2 Гбайт, 4 Гбайт або 8 Гбайт типу LPDDR4-3200;

- підтримка бездротової мережі на частотах 2,4 ГГц і 5,0 ГГц за стандартом IEEE 802.11ac, а також Bluetooth 5.0 і BLE;
- можливість підключення до мережі Gigabit Ethernet;
- чотири порти USB – 2 порти USB 3.0 та 2 порти USB 2.0;
- 40-контактний заголовник GPIO зі зворотною сумісністю;
- два мікро HDMI-порти з підтримкою дошки 4K при 60 кадрах в секунду;
- порти MIPI DSI для підключення дисплеїв та MIPI CSI для підключення камери;
- підтримка популярних кодеків, таких як H. 265 (декодування 4K при 60 кадрах в секунду), H. 264 (декодування 1080р при 60 кадрах в секунду, кодування 1080р при 30 кадрах в секунду);
- підтримка графічних інтерфейсів OpenGL ES 3.1 і Vulkan 1.0;
- слот для карти Micro-SD для зберігання та читання інформації;
- можливість живлення через USB-C або GPIO піни напругою 5В постійного струму;
- можливість живлення через порт Ethernet;
- також на платі Raspberry Pi 4 є ряд пінів для різноманітних завдань і підключення модулів, включаючи піни для 3В, 5В, землі, вхідно-вихідних пінів загального призначення, ШІМ, інтерфейсів I2C, I2S, SPI та серійного зв'язку.

Порядок пінів та їх значення показано на рис. 2.7.

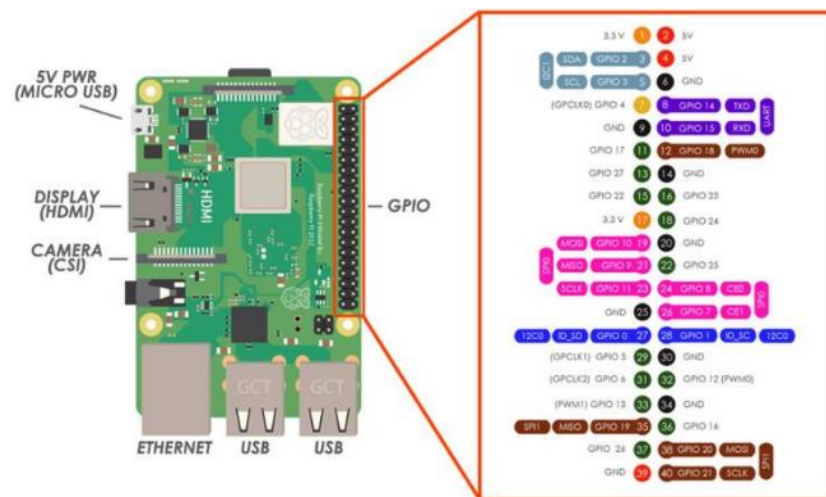


Рисунок 2.7 – Розпіновка плати та їх значення

Raspberry Pi 4 продовжує мати 40 контактів GPIO, проте має додаткові з'єднання для I2C, SPI та UART. GPIO – це один з найпростіших, але популярних аспектів Raspberry Pi. Навіть у багатьох новітніх платах від інших виробників від нього відмовляються, проте він залишається дуже популярним для підключення модулів. Виводи GPIO є цифровими і можуть бути в стані нуль або одиниця (вимкнено або увімкнено). Вони можуть приймати або відправляти сигнали (відповідно вхід або вихід), а також контролювати стани та напрямки сигналів за допомогою мов програмування, таких як Python, JavaScript, node-RED і т.д. Більшість контактів на заголовку прямо підключені до чіпу Broadcom. Важливо ретельно планувати підключення компонентів, щоб уникнути пошкодження плати.

Висновки до розділу 2

Проведено обґрунтування для створюваного апаратно-програмного модуля візуального спостереження. Охарактеризовано використовуваний мікропроцесор Raspberry Pi 4 та камера RasPi cam V2.

Наведено функціональні можливості Visual Studio Code та мови програмування Python. Відзначено такі функціональні переваги цієї мови програмування, як візуалізація даних, яка є важливою частиною аналізу даних. Вона допомагає представляти необроблену чи очищену інформацію у

більш зрозумілій формі. Python пропонує широкий спектр інструментів для візуалізації даних, таких як Matplotlib, Seaborn та Plotly, які дозволяють створювати графіки, діаграми та інші візуальні елементи для глибокого розуміння і представлення даних.

Крім того, Python забезпечує зручність і легкість у написанні коду, що спрощує процес розробки та впровадження програмного забезпечення для відеоспостереження. Використання цих технологій дозволяє створювати ефективні та надійні системи візуального спостереження, що відповідають сучасним вимогам безпеки та моніторингу.

3 ДОСЛІДЖЕННЯ ОСОБЛИВОСТЕЙ ВПРОВАДЖЕННЯ РОЗРОБЛЕНОГО МОДУЛЯ ВІЗУАЛЬНОГО СПОСТЕРЕЖЕННЯ

3.1 Програмна складова модуля

На початку практичної реалізації апаратно-програмного модулю візуального спостереження проаналізуємо вимоги користувача до системи. Вивчення цих вимог та їх врахування дозволяють забезпечити комфорт для користувача і підвищити ефективність апаратно-програмного модулю візуального спостереження.

Надійність є одним із основних критеріїв якості апаратно-програмного модулю візуального спостереження. Це важливо для забезпечення безперебійної роботи системи та збереження її функціональних характеристик у будь-яких умовах експлуатації [19].

Серед ключових вимог до надійності апаратно-програмного модулю візуального спостереження можна виділити такі:

- відновлення після збоїв: Система повинна автоматично відновлюватися після виникнення апаратних збоїв, таких як відключення електроживлення. Однак існують винятки, коли збої пов'язані з пошкодженням носіїв інформації або доступом до інформації на зовнішніх ресурсах;

- обробка виняткових ситуацій: Апаратно-програмний модуль візуального спостереження повинен ідентифікувати та надавати користувачеві інформацію про помилкові дії. Після успішної ідентифікації помилки повинно виводитися відповідне повідомлення;

- вимоги до продуктивності: Система візуального спостереження має забезпечувати швидке та ефективне виконання функцій. Критичними аспектами є час відгуку системи, який не повинен перевищувати 3 секунди, і час очікування відповіді від стороннього ресурсу, що не повинен перевищувати 5 секунд;

– вимоги до ергономіки і зручності експлуатації: Користувацький інтерфейс апаратно-програмного модулю візуального спостереження повинен забезпечувати зручну взаємодію з програмним забезпеченням. Зручність використання апаратно-програмного модулю візуального спостереження визначається характеристиками користувацького інтерфейсу, які впливають на ефективність використання та задоволення користувача від роботи з системою.

Вимоги до ергономіки апаратно-програмного модулю візуального спостереження включають:

- інтуїтивно зрозумілий інтерфейс: Система має мати інтерфейс, який легко розуміється користувачем без додаткових пояснень або інструкцій;
- інформативні повідомлення про помилки: Повідомлення про помилки в системі мають бути короткими, але інформативними, щоб користувач міг швидко зрозуміти причину помилки і вжити відповідних заходів;
- однакова зовнішня поведінка: Елементи інтерфейсу, що виконують однакові функції, повинні мати однакову зовнішню поведінку, щоб користувач легко міг передбачити результати своїх дій [26].

Функціональні вимоги апаратно-програмного модулю візуального спостереження визначаються тим, які завдання він повинен вирішувати і що саме має робити. Ці вимоги можуть включати розрахунки, технічні аспекти, обробку та маніпулювання даними, а також інші конкретні функції, які система повинна виконувати. Діаграма випадків використання є складовою частиною моделі прецедентів і представляє всі можливі дії користувачів, пов'язані з різними сценаріями використання. Ця діаграма повинна повністю описувати функціональність системи на певному рівні абстракції.

Нефункціональні вимоги визначають характеристики якості апаратно-програмного модулю візуального спостереження, такі як доступність, зручність установки, продуктивність, надійність та умови експлуатації. Серед

найважливіших атрибутів якості є надійність, продуктивність і зручність використання.

Функціональні вимоги щодо ергономіки апаратно-програмного модулю візуального спостереження описують, як саме користувач взаємодіє з програмним продуктом. Наприклад, інтерфейс повинен бути інтуїтивно зрозумілим, а користувач повинен мати можливість самостійно обирати пристрій для керування.

Математична модель апаратно-програмного модулю візуального спостереження може бути описана за допомогою методів статистичного моделювання, таких як Вінерівські та узагальнені Вінерівські процеси. Ці методи знаходять застосування в різних галузях, включаючи фінансову математику, обчислювальну математику та моделювання роботи апаратно-програмного модулю візуального спостереження [21].

У даному дослідженні використовується метод статистичного моделювання дробового броунівського руху для опису функціонування апаратно-програмного модулю візуального спостереження. Цей метод ґрунтується на властивості дробового броунівського руху мати стаціонарні прирости та дозволяє побудувати реалізацію руху з заданою проекцією.

Визначення 1. Узагальнений Вінерівський процес (дробовий броунівський рух) з індексом Херста $H \in (0, 1)$ називається гаусівський процес $W_H(t)$, $t \in [0, 1]$ такий, що $W_H(0) = 0$, $E W_H(t) = 0$ і кореляційною функцією:

$$R_H(t, s) = \frac{1}{2} (|t|^{2H} + |s|^{2H} - |t-s|^{2H}) \quad (3.1)$$

Якщо $H = 1/2$, тоді маємо стандартний Вінерівський процес.

Вінерівський процес $W(t)$ – це процес з незалежними змінами. Дробовий Броунівський рух $W_H(t)$ – це процес зі стаціонарними змінами.

Тоді випадковий процес $w(t) = W_H(t + \Delta) - W_H(t)$ з фіксованим $\Delta \in$ стаціонарним гаусівським процесом з кореляційною функцією.

$$E w(t + \tau) w(t) = \frac{1}{2} (|\tau + \Delta|^{2H} + |\tau - \Delta|^{2H} - 2|\tau|^{2H}) \quad (3.2)$$

і спектральна щільність:

$$g(\lambda) = \frac{A^2}{\pi} \left(\frac{1 - \cos(\lambda\Delta)}{|\lambda|^{2H+1}} \right), \lambda \in (-\infty, +\infty) \quad (3.3)$$

де:

$$A^2 = \left(\frac{2}{\pi} \int_0^\infty \frac{1 - \cos(\lambda)}{\lambda^{2H+1}} d\lambda \right)^{-1} = \left(-\frac{2}{\pi} \Gamma(-2H) \cos(H\pi) \right)^{-1} \quad (3.4)$$

Оскільки $WH(0) = 0$, то будь-яку модель дробового броунівського руху можна зобразити:

$$W_H(t + \Delta) = W_t(t) + w(t) \quad (3.5)$$

Моделювання дробового броунівського руху вимагає моделювання гаусівського стаціонарного процесу для апаратно-програмного модуля візуального спостереження.

Нехай $\xi(t)$ – дійсний гаусівський стаціонарний випадковий процес з кореляційною функцією $R(\tau)$ і спектральною функцією [23]:

$$F(\lambda), R(\tau) = \int_0^\infty \cos(\lambda\Delta) dF(\lambda) \quad (3.6)$$

Гаусівський стаціонарний випадковий процес можна представити у вигляді:

$$\xi(t) = \int_0^\infty \cos(\lambda t) d\xi_1(\lambda) + \int_0^\infty \sin(\lambda t) d\xi_2(\lambda) \quad (3.7)$$

де $\xi_1(t)$ та $\xi_2(t)$ – центровані і некорельовані випадкові процеси, такі що при $0 < \lambda_1 < \lambda_2$ має місце:

$$E(\xi_1(\lambda_2) - \xi_1(\lambda_1))^2 = F(\lambda_2) - F(\lambda_1), \quad (3.8)$$

$$E(\xi_2(\lambda_2) - \xi_2(\lambda_1))^2 = F(\lambda_2) - F(\lambda_1), \quad (3.9)$$

Нехай $\Delta\Lambda$ – часткове розбиття інтервалу $[0, \Lambda]$. Модель випадкового процесу $\xi(t)$ можна продемонструвати у вигляді центральної некорельованої субгаусівської послідовності:

$$E(\eta_{1i})^2 = E(\eta_{2i})^2 = F(\lambda_{i+1}) - F(\lambda_i) \quad (3.10)$$

Тобто випадковий процес $w(t)$ можна представити у вигляді:

$$w(t) = \int_0^\infty \cos(\lambda t) d\xi_1(\lambda) + \int_0^\infty \sin(\lambda t) d\xi_2(\lambda) \quad (3.11)$$

Для розбиття/модель випадкового процесу $w(t)$ має вигляд:

$$w_n(t, \Lambda) = \sum_{k=0}^{n-1} (\sin(\lambda_k t) X_k + \cos(\lambda_k t) Y_k) \quad (3.12)$$

де $\{X_k, Y_k\}$ – некорельовані субгаусівські випадкові величини з $EX_k = EY_k = 0$ і

$$E(X_k)^2 = E(Y_k)^2 = \int_{\lambda_k}^{\lambda_{k+1}} g(\lambda) d\lambda \quad (3.13)$$

Нехай випадковий процес $X(t)$ і всі $X_n(t, \Lambda)$ належать деякому функціональному простору $A(T)$ з нормою $\|\cdot\|$. Нехай дано два числа: $\delta > 0$ і $0 < \varepsilon < 1$.

Визначення 2. Модель $X_n(t, \Lambda)$ апроксимує процес $X(t)$ з надійністю $1 - \varepsilon$ і точністю $\delta > 0$ в нормі простору $A(T)$, якщо існує нерівність:

$$P\{\|X(t) - X_n(t, \Lambda)\| > \delta\} \leq \varepsilon \quad (3.14)$$

Нехай для $D\Lambda$ та $\frac{T\Lambda}{n} \leq 1$

Теорема 1. Модель $W_n(t, \Lambda)$ апроксимує процес $w(t)$ з надійністю $1 - \varepsilon$ і точністю δ у просторі $L_2(T)$, якщо для чисел Λ і n виконуються нерівності:

$$\exp\left\{\frac{1}{2}\right\} \frac{\delta}{\sqrt{G_{2n,\Lambda}}} \exp\left\{-\frac{\delta^2}{2G_{2n,\Lambda}}\right\} \leq \varepsilon \quad (3.15)$$

Де $G_{2n,\Lambda} = \frac{T^3 \Lambda^2}{3n^2} \int_0^\Lambda g(\lambda) d\lambda + T \left(\int_\Lambda^\infty g(\lambda) d\lambda\right)$

Для чисел Λ і n ставимо:

$$\Lambda = \left(\frac{3n^2}{T^3}\right)^{\frac{1}{2H+1}} = G_{2n} = T^{\frac{3H}{H+1}} \left(1 + \frac{T}{H}\right) \left[(3n^2)^{\frac{H}{H+1}}\right] \quad (3.16)$$

У запропонованій математичній моделі апаратно-програмного модулю візуального спостереження демонструється у вигляді графа $G = (V, E)$, де $V = \{1, 2, \dots, N\}$ та $E = \{1, 2, \dots, M\}$ - множини маршрутизаторів і M ліній зв'язку між ними. Пряма лінія зв'язку m має місткість c_m (в одиницях/сек). Всі вузли в мережі MPLS розділяються на дві частини: $V^+ = \{V_i^+, i = 1, n_{LER}\}$ - множина граничних маршрутизаторів LER і $V^- = \{V_i^-, i = n_{LSR}\}$ - множина комутаторів міток LSR.

Кожен елемент множини V^- може виступати як джерело трафіку і як одержувач. Якщо маршрутизатор є джерелом трафіку, це означає, що на нього надходить трафік з суміжної мережі (IP, MPLS, ATM і т.д.). Цей трафік повинен бути доставлений до вузла одержувача, який є також точкою з'єднання з суміжними мережами. У випадку граничних маршрутизаторів LER кожен з них є як джерелом, так і одержувачем, а LSR не може функціонувати як одержувач трафіку, що приходить на нього з суміжної мережі [24].

На момент часу $t = T$, трафік інтенсивністю $\lambda(t)$, надходить на один з маршрутизаторів, який відноситься до одного з класів обслуговування q з множини Q , що включає вимоги щодо якості обслуговування (QoS), такі як максимальна допустима затримка Q_q і максимально допустимий відсоток втрат l_q . Весь вхідний трафік розділяється на потоки класів обслуговування таким чином, щоб забезпечити виконання усіх вимог класів Q (t) у повному обсязі. Таким чином, множина всіх каналів трафіку з QoS Y на лінії зв'язку m може бути визначена як:

$$Y = Y(d_y, P_y, l_y) \quad (3.17)$$

де d_y - смуга пропускання каналу; $P_y = \{p_y^1, \dots, p_y^{L_y}\}^{-1}$ - допустима множина шляхів для шляху L_y певна для кожного каналу трафіку.

У даному випадку, швидкість надходження трафіку для апаратно-програмного модулю візуального спостереження можна позначити так: у момент часу t на граничний маршрутизатор передається трафік інтенсивністю λ , який належить до класу обслуговування q , Цей трафік потрібно доставити

на вихідний маршрутизатор через будь-який шлях з множини P_y , забезпечуючи, щоб не було перевищення заданих максимально допустимих значень затримки Q_q і максимально допустимого відсотку втрат.

У момент часу t кожен вузол мережі V характеризується продуктивністю, коефіцієнтом відносних втрат і середнім часом очікування пакета в черзі. Величина втрат для всіх вузлів мережі обмежується:

$$0 \leq X_v^q(t), \sum_v^{p_y} X_v^q(t) \leq l_q \quad (3.18)$$

де $V = \overline{l, P_y, l_q}$ – заданий максимально допустимий відсоток втрат. Втрати визначаються як співвідношення кількості відкинутих даних до загальної кількості даних, які прийшли на обробку. Обмеження часу затримки для всіх вузлів мережі в апаратно-програмному модулі візуального спостереження встановлюються аналогічним чином:

$$0 \leq T_v^q(t), \sum_v^{p_y} T_v^q(t) \leq \tau_q \quad (3.19)$$

де $V = \overline{l, P_y, \tau_q}$ – задане максимально допустиме значення затримки. Вартість маршрутизації c_m призначається кожній лінії зв'язку m і може залежати від різноманітних параметрів, таких як швидкість передачі даних, довжина маршруту, надійність і інші. Вартість шляху обчислюється як сума вартості всіх ліній зв'язку, що його складають:

$$C_y^l = \sum_{m \in p_y^l} c_m \quad (3.20)$$

Якщо рівняння представляє пропускну здатність апаратно-програмного модулю візуального спостереження, який спрямовується на допустимий шлях каналу передачі у трафіку, тоді виконується:

$$\sum_{t \in T, l=1}^{L_y} x_y^l(t) = d_y \quad \forall y \in Y, \forall l \in \{1, \dots, L_y\} \quad (3.21)$$

Загальне навантаження на лінії зв'язку m визначається за формулою:

$$f_m = \sum_{\lambda_{p_y^q}(t) \in Y} \sum_{l=1}^{L_y} a_{y,l}^m x_y^l(t), \quad \forall m \in E \quad (3.22)$$

якщо шлях P у використовує лінію зв'язку m , в іншому випадку;

$x_y^l(t)$ – пропускна здатність системи передавати дані під час функціонування апаратно-програмного модулю візуального спостереження згідно вищеприведеного рівняння.

Представлена модель дозволяє розробити апаратно-програмний модуль візуального спостереження та структуру мереж із технологією MPLS для віддаленого управління [22].

У логічній моделі апаратно-програмного модулю візуального спостереження описуються її значущі функціональні модулі та класи. Модулі проєктованого апаратно-програмного модулю візуального спостереження представлені на діаграмі класів (рис. 3.1).



Рисунок 3.1 – Діаграма класів ергономіки апаратно-програмного модулю візуального спостереження

Опис модулів ергономіки передавання інформації при функціонуванні апаратно-програмного модулю візуального спостереження наведено в табл. 3.1.

Таблиця 3.1 – Опис модулів ергономіки апаратно-програмного модулю візуального спостереження

Найменування	Опис
Користувацький інтерфейс	Містить атрибути взаємодії користувача з програмою. Дозволяє вибрати параметри даних, реалізувати пошук і сортування.
Імпорт даних	Містить атрибут, що виконує імпорт даних з сайту fipam. ua
Транслятор даних	Містить атрибут, що формує імпортовані дані в таблицю.
Графічний модуль	Містить атрибут, що виробляє побудову графіка і індикатора на основі імпортованих даних.

Модель розміщення. В рамках моделі розміщення ергономіки апаратно-програмного модулю візуального спостереження формується діаграма розгортання, що містить опис фізичного простору для розміщення компонентів. Діаграма розгортання представлена на рис. 3.2.

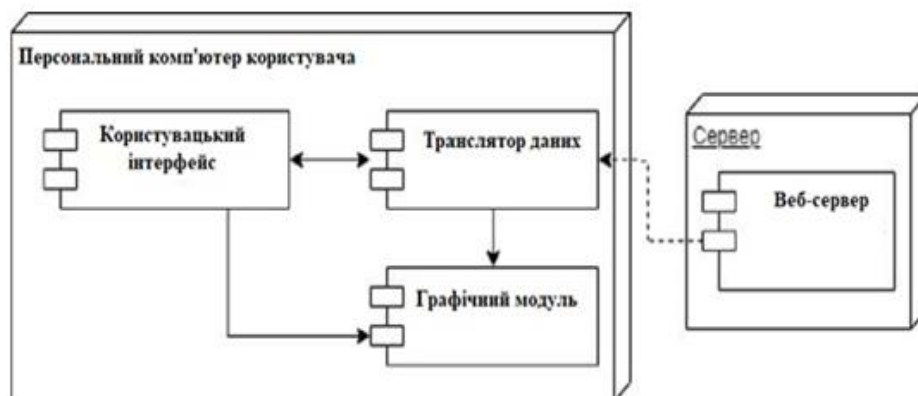


Рисунок 3.2 – Діаграма розгортання апаратно-програмного модулю візуального спостереження

Для безпосередньої реалізації ергономіки апаратно-програмного модулю візуального спостереження важливо виокремити ключові логічні і функціональні структури. Для опису елементів ергономіки такої системи використовується діаграма компонентів [26].

Оскільки основна мета дослідження полягає у створенні апаратно-програмного модулю візуального спостереження, то для досягнення цієї мети визначено такі цілі: аналіз предметної області, проектування системи, реалізація програмної частини, вдосконалення системи, підтримка продукту. Подане дерево цілей для даного дослідження можна оглянути на рис. 3.3 в такому вигляді.



Рисунок 3.3 – Зовнішній вигляд дерева цілей згідно досліджуваної проблематики в роботі

Далі в системному аналізі об'єкту дослідження виконується створення матриці порівняння критеріїв з типами систем (таблиця 3.2). Після нормалізації елементів стовпця та розрахунку середнього значення рядка формується вектор пріоритетів.

Таблиця 3.2 – Ранжування критеріїв в дереві цілей для апаратно-програмного модулю візуального спостереження

№	Назва критерію	Порівняння критеріїв						Вектор пріоритету
		1	2	3	4	5	6	
1	Актуальність	1	1 / 4	1 / 3	5	1 / 4	2	0,15
2	Точність	1	3	7	7	1	1	0,26
3	Зручність	1 / 7	4	1	5	1 / 2	1 / 7	0,114
4	Коректність	2	1	4	1 / 6	3	1	0,17
5	Доступність	1 / 3	1 / 7	1 / 5	1 / 5	1 / 8	1	0,04
6	Якість	1 / 2	7	1	6	1	8	0,33

Після цього проводиться попарне порівняння кожного критерію з типами систем. Таким чином було оцінено всі системи за кожним визначеним критерієм (таб. 3.3–3.9).

Таблиця 3.3 – Матриця попарного порівняння за критерієм «Актуальність» в дереві цілей для апаратно-програмного модулю візуального спостереження

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1 / 2	1 / 7	6	1 / 6	1 / 7	0,4
2	Інформаційно-керівні	6	1 / 3	5	1 / 4	1 / 4	0,2
3	Інформаційно-пошукові	1 / 5	1 / 6	1 / 2	1 / 4	1 / 6	0,3
4	Інтелектуальні інформаційні	4	1 / 3	5	1	1 / 2	0,5
5	СППР	5	3	5	1 / 2	1	0,45

Таблиця 3.4 – Матриця попарного порівняння за критерієм «Точність» в дереві цілей для апаратно-програмного модулю візуального спостереження

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1 / 2	6	1 / 4	1 / 5	7	0,74
2	Інформаційно-керівні	1 / 2	3	1 / 4	1 / 4	6	0,28
3	Інформаційно-пошукові	1 / 5	1 / 7	1 / 5	1 / 3	5	0,22
4	Інтелектуальні інформаційні	1 / 4	3	3	1 / 8	1 / 3	0,27
5	СППР	1 / 6	1 / 2	1 / 4	1 / 5	1 / 4	0,21

Таблиця 3.5 – Матриця порівняння для критерію «Зручність» в дереві цілей для апаратно-програмного модулю візуального спостереження

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1 / 2	1 / 4	2	1 / 5	5	0,2
2	Інформаційно-керівні	4	1 / 2	5	1 / 4	3	0,5
3	Інформаційно-пошукові	1 / 5	1 / 6	1 / 5	1 / 4	7	0,4
4	Інтелектуальні інформаційні	1 / 7	3	5	2	4	0,38
5	СППР	3	1 / 3	1 / 3	1 / 4	4	0,17

Таблиця 3.6 – Матриця порівняння для критерію «Коректність» дереві цілей для апаратно-програмного модулю візуального спостереження

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
1	Інформаційно-консультаційні	1 / 3	2	3	3	7	0,6

№	Назва системи	Порівняння систем					Вектор пріоритету
		1	2	3	4	5	
2	Інформаційно-керівні	1 / 3	4	1 / 5	4	5	0,24
3	Інформаційно-пошукові	1 / 2	3	1 / 3	3	1 / 2	0,36
4	Інтелектуальні інформаційні	1 / 2	1	1 / 2	1 / 2	3	0,22
5	СППР	1 / 2	3	4	1 / 5	1 / 4	0,28

Таблиця 3.7 – Матриця порівняння для критерію «Доступність» в дереві цілей для апаратно-програмного модулю візуального спостереження

№	Назва системи	Порівняння систем					Вектор пріоритету
		1 / 2	3	4	3	5	
1	Інформаційно-консультаційні	3	1 / 4	1/2	5	1 / 3	0,24
2	Інформаційно-керівні	1 / 7	1 / 2	1 / 3	2	3	0,22
3	Інформаційно-пошукові	3	6	4	3	4	0,6
4	Інтелектуальні інформаційні	4	3	1/4	3	1 / 3	0,6
5	СППР	3	2	1 / 3	1 / 4	1	0,49

Таблиця 3.8 – Матриця порівняння для критерію «Якість» в дереві цілей для апаратно-програмного модулю візуального спостереження

№	Назва системи	Порівняння систем					Вектор пріоритету
		2	3	3	6	3	
1	Інформаційно-консультаційні	3	7	9	1 / 5	2	0,47
2	Інформаційно-керівні	5	3	1 / 8	3	4	0,33
3	Інформаційно-пошукові	1 / 2	1 / 7	1 / 3	1 / 2	3	0,24
4	Інтелектуальні інформаційні	1 / 9	4	1 / 3	1 / 7	1/2	0,26
5	СППР	1 /	6	4	1 / 4	6	0,44

№	Назва системи	Порівняння систем					Вектор пріоритету
		2	3	3	6	3	
		7					

Далі при аналізі дерева цілей для апаратно-програмного модуля візуального спостереження була створена таблиця, в якій були зібрані всі коефіцієнти критеріїв кожного типу системи, і була визначена кінцева пріоритетність розробки у табл. 3.9.

Таблиця 3.9 – Результати методу аналітичної ієрархії в дереві цілей для апаратно-програмного модулю візуального спостереження

Критерій та Тип систем	Актуальність	Точність	Зручність	Коректність	Доступність	Якість	Пріоритет
Інформаційно-консультаційні	0,015	0,034	0,165	0,018	0,086	0,2566	0,49
Інформаційно-керівні	0,15	0,064	0,044	0,029	0,018	0,085	0,28
Інформаційно-пошукові	0,17	0,029	0,023	0,024	0,023	0,036	0,12
Інтелектуальні інформаційні	0,032	0,035	0,042	0,026	0,022	0,033	0,19
СППР	0,033	0,019	0,011	0,028	0,019	0,045	0,29

Результатом проведеного порівняння є вибір найбільше значення у інформаційно-консультаційній системі. Тобто основним пріоритетним призначенням створюваної апаратно-програмного модулю візуального спостереження є інформаційно-консультаційна спрямованість розробки.

Мета використання такого типу дерев рішень полягає у створенні моделі, придатної для прогнозування класу або значення цільової змінної шляхом вивчення простих правил прийняття рішень, отриманих з навчальних даних. У деревах рішень для передбачення мітки класу для запису ми починаємо з кореня дерева, порівнюючи значення кореневого атрибута з

атрибутом запису. В залежності від результату порівняння ми переходимо до відповідної гілки і продовжуємо рухатися до наступного вузла. Типи дерев рішень визначаються типом цільової змінної, яка може бути одного з двох видів [29]:

Контекстна діаграма або діаграма нульового рівня для апаратно-програмного модулю візуального спостереження зображена на рис. 3.4 є основою та базовою діаграмою для потоків даних.



Рисунок 3.4 – Діаграма потоків даних нульового рівня

Під час розробки контекстної діаграми було виявлено два елементи DFD на нульовому рівні, тобто на рівні контекстної діаграми:

– *користувач*. Показано зовнішній об'єкт, що використовується для представлення користувача, який взаємодіє з системою.

– *надати можливість доступу*. Показано основний процес, який виконує система, для обробки та потоку даних. Сформовані процеси, їхні потоки та бази даних зображено на рис. 3.5 саме:

- створити обліковий запис;
- сформувати список;
- надіслати сформований результат опрацювань;
- бази даних: користувачі та база даних та їх потреб.

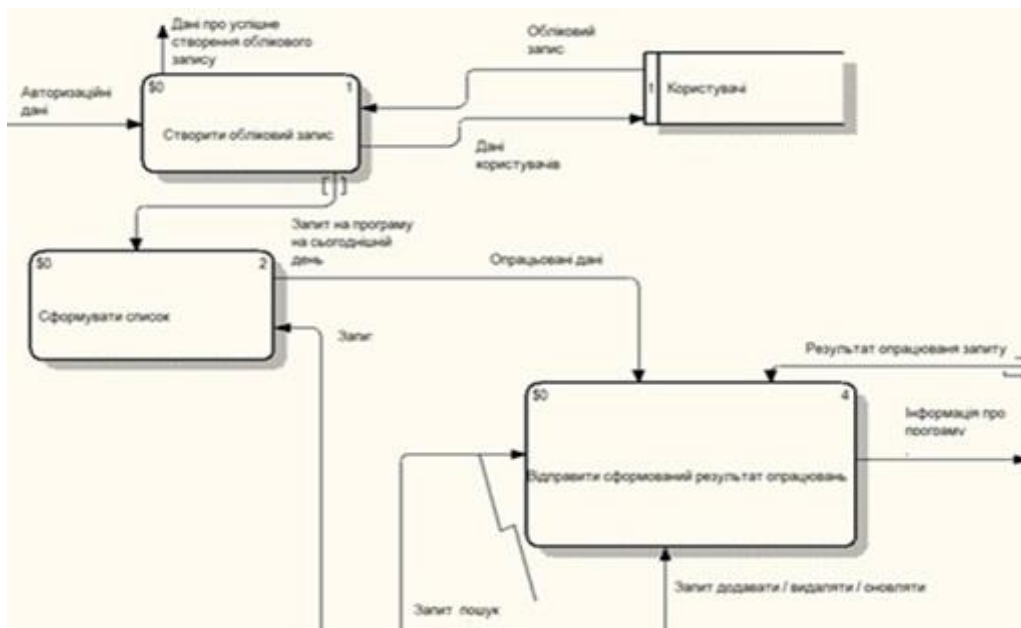


Рисунок 3.5 – Декомпозиція апаратно-програмного модулю візуального спостереження першого рівня

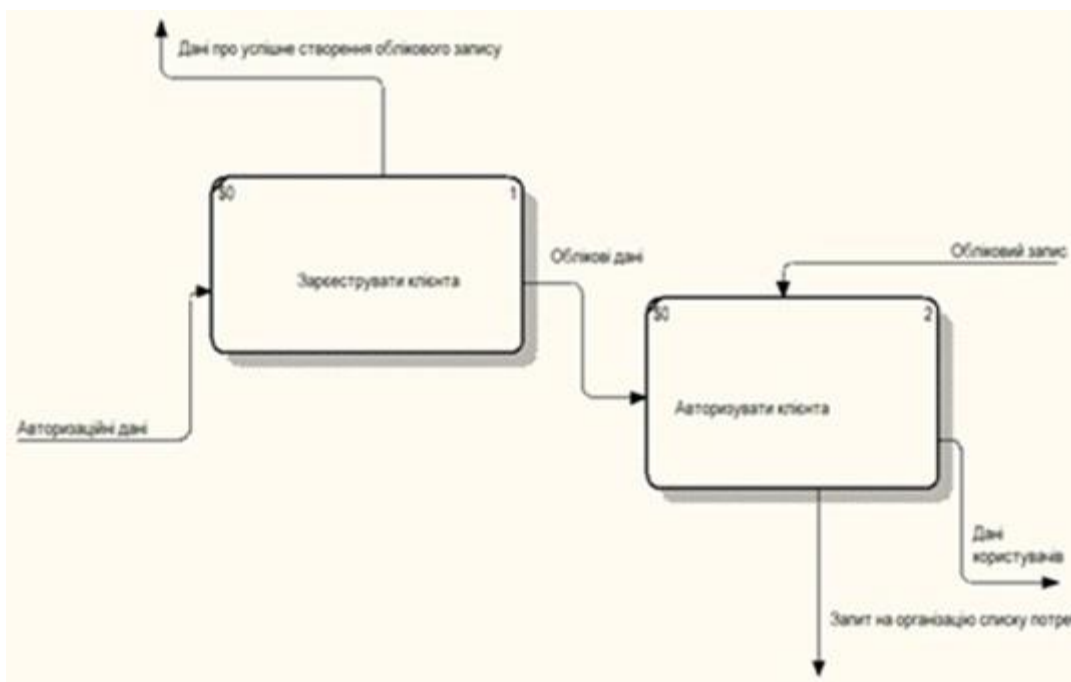


Рисунок 3.6 – Декомпозиція процесу «Створити обліковий запис» для апаратно-програмного модулю візуального спостереження

Наступним етапом є декомпозиція процесу другого рівня під назвою "Створення облікового запису". Цей процес включає такі кроки: реєстрація клієнта і авторизація клієнта.

Облікові дані, пов'язані між собою, передаються при авторизації користувача для обробки даних і надання доступу до програмного забезпечення.



Рисунок 3.7 – Декомпозиція процесу «Сформувати відеопотік»

Наступний процесом описано дії користувача в межі пошуку результатів прогнозу істинності силуету, зображено на рис. 3.8. Сформований запит на перегляд результатів відображається користувачу у вигляді білої рамки. Наступним складеним процесом є «Сформувати остаточний вигляд кадру з обведеними силуетами». Цей блок включає в себе наступні процеси: створення запиту на завантаження даних і схвалення вибраних даних [25].

Цей процес об'єднує опрацьований матеріал, а саме програми, які в даний момент можуть транслюватися. В результаті отримуємо оброблені дані та запит на перегляд інформації.



Рисунок 3.8 – Декомпозиція процесу «Відправити сформований результат опрацювань»

Дана декомпозиція зображена на рис. 3.8 складається з двох процесів:
вибрати телеканал для показу результатів уточнення силуету на телекадрі;

вибрати показ програми.

Ці процеси взаємодіють через програми, які користувач обирає та якими опрацьовуються запити. На вході отримується запит на пошук даних відеосигналу через різні канали.

Бібліотека using System; – список імен системи включає класи і базові класи, які визначають загальнозживані типи значень, довідкові типи даних, події та обробники подій, інтерфейси, атрибути і винятки для обробки.

.Collections. Generic; – список імен *System. Collections. Generic* котрий включає різноманітні інтерфейси та класи, які визначають загальні колекції, надають допоміжні засоби для створення колекцій, що забезпечують кращу безпеку та ефективність типів, ніж інші колекції сильного типу.

.Drawing; – список імен *System. Drawing* дає доступ до основних функцій графіки *GDI +*. Здебільш розширена функціональність надається в просторах імен *System. Drawing. Drawing2D, System. Drawing. Imaging ma System. Drawing. Text*.

.Windows. Forms; – список імен *System. Windows. Forms* включає класи, що дозволяють розробляти програми під управлінням Windows і повністю використовувати можливості багатифункціонального користувацького інтерфейсу, доступного в операційній системі *Microsoft Windows*.

Using Emgu. CV; – *Emgu CV* – Це абстракція платформи .NET над бібліотекою обробки зображень *OpenCV*, що дозволяє викликати її функції *OpenCV* з *.NET*

System. IO; – список імен *System. IO* включає типи, що дозволяють зчитувати та записувати дані в файли та потоки, а також типи, які підтримують основні операції з файлами та каталогами.

.Data. SqlClient; – список імен *System. Data. SqlClient* – це постачальник даних. *NET* для *SQL Server*.

.Data; – список імен *System. Data* забезпечує доступ до класів, що описують структуру *ADO. NET. ADO. NET*, надає можливість створювати елементи, котрі ефективно управляють даними із різноманітних джерел.

Система візуального спостереження має наступний (рис. 3.9) набір модулів та алгоритм роботи:

- обробка вхідних відео та отримання набору кадрів;
- сегментація кадрів, виділення областей відеоспостереження;
- розпізнавання відеосигналу з кожної виділеної області [14];
- відображення об'єкту спостереження на поточному кадрі;
- створення результуючого відео з опрацьованих кадрів;
- опрацювання та аналіз даних з кожного відеопотоку;
- виведення результатів ідентифікації та трекінгу.



Рисунок 3.9 – Алгоритм системи візуального спостереження

Виявлення об'єктів – одна з класичних проблем при візуальному спостереженні, де алгоритм визначає, які об'єкти присутні на зображенні і їх місцезнаходження. Проблема виявлення об'єктів є складнішою в порівнянні з класифікацією, яка також визначає тип об'єктів, але не вказує їх точне положення на зображенні. Крім того, класифікація не ефективна для зображень, що містять більше одного об'єкта.

Алгоритм вибору обмежувальних границь для кожного класу:

- відкинути всі поля зі значенням $C < C\text{-порогове}$ (наприклад, $C < 0,5$);

- відсортувати прогнози, починаючи з найвищого значення C ;
- визначити поле з найбільшим значенням C і показати його як результат.

Модуль трекінгу виконується згідно з наступним алгоритмом:

Система обробляє кожне з певних зображень з різних камер протягом заданого часового інтервалу, щоб їх потім групувати;

Для зображення того, як користувач взаємодіє з системою, було створено діаграму прецедентів (рис. 3.10).

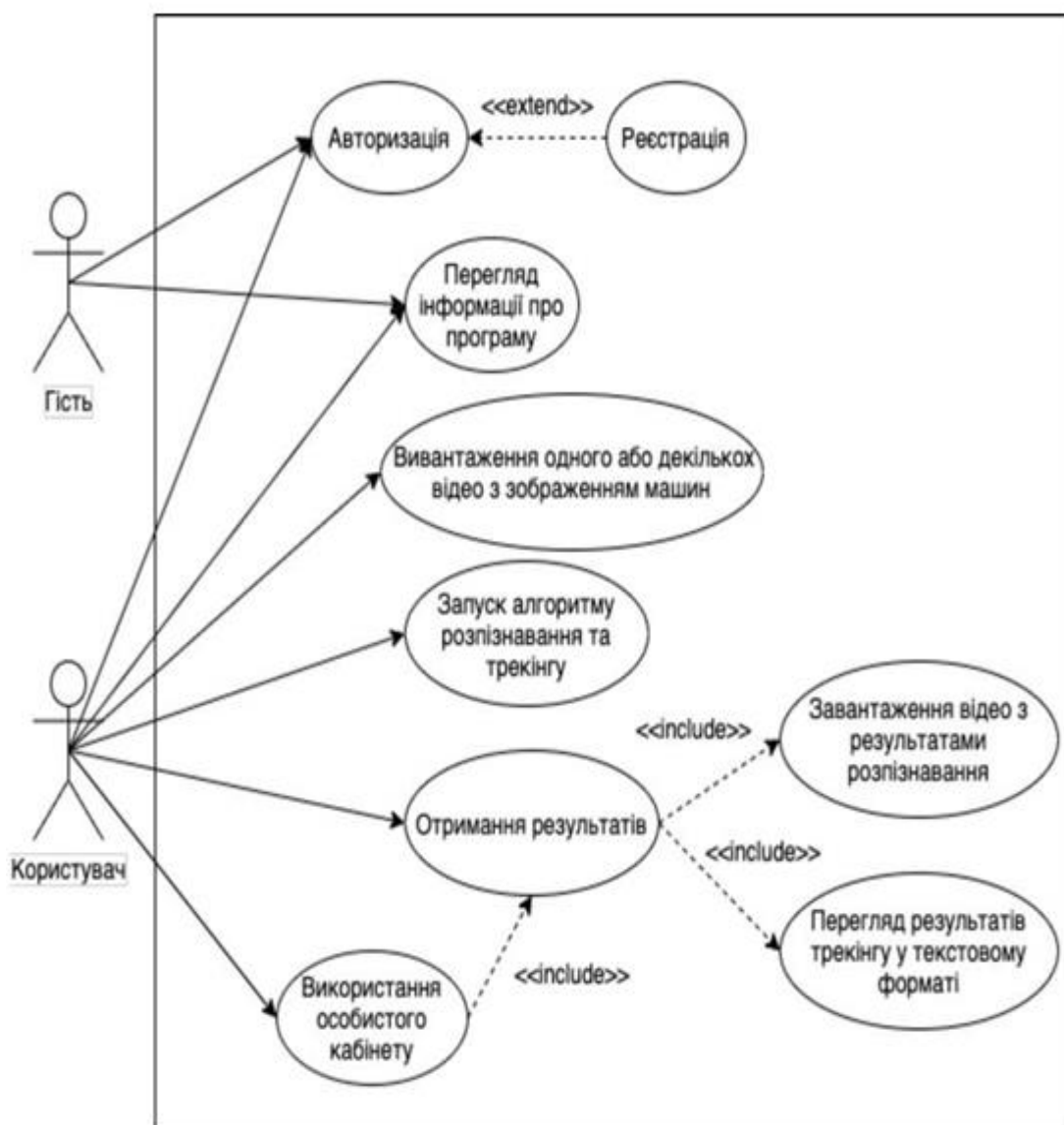


Рисунок 3.10 – Діаграми прецедентів апаратно-програмного модулю візуального спостереження

Діаграма прецедентів показує, як різні актори системи взаємодіють з нею та визначає рівень доступу для кожного з них [30].

Діаграми послідовності, що зазвичай використовуються розробниками, моделюють взаємодію об'єктів у межах одного випадку використання. Вони демонструють, як різні частини системи взаємодіють для виконання певної функції та в якому порядку ці взаємодії відбуваються. Діаграма послідовності ілюструє роботу системи в "послідовності", необхідній для досягнення певного використання (рис. 3.11).

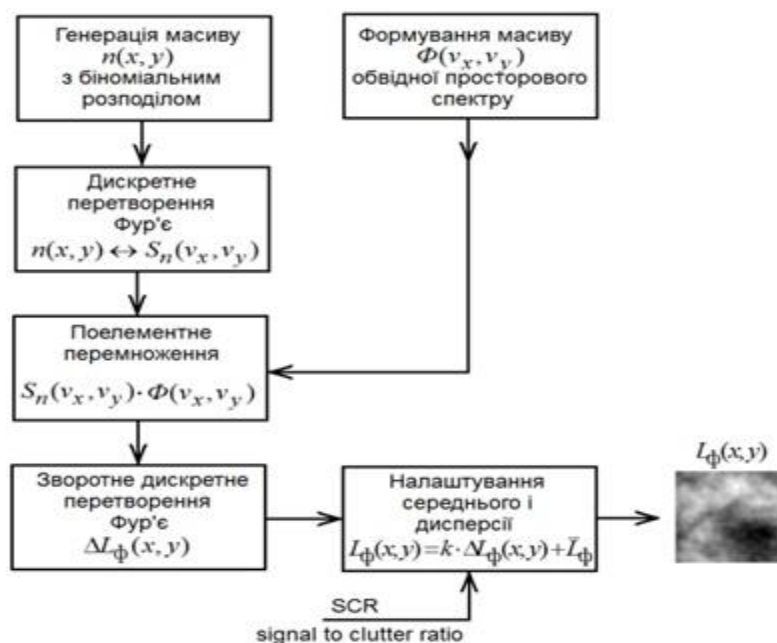


Рисунок 3.11 – Алгоритм програмної побудови моделі фону

Для початку налагодження пристрою необхідно мати наступне обладнання: (Raspberry Pi 3B), модуль камери Raspі Cam V1.3, SD-карта на 16 Гбайт, мікрофон SunFounder USB, кабель Micro USB, кабель HDMI, монітор, клавіатуру та мишку.

Далі потрібно завантажити операційну систему для Raspberry Pi. На офіційному сайті доступні дві основні операційні системи: NOOBS (система, що допомагає встановлювати потрібну операційну систему) та Raspbian.

Враховуючи вищезазначену інформацію, можна сформулювати основні вимоги до апаратно-програмного комплексу на Raspberry Pi наступним чином:

Hardware

- одноплатний комп'ютер Raspberry Pi 4;
- модуль Raspberry Pi camera module v2;
- підключення USB-Type C power supply;
- кабель HDMI;
- монітор для виводу картинки;
- корпус створений за допомогою 3d-принтера (за бажанням).

Software

- інструменти розробки алгоритму та створення каскадного класифікатора, і завантаження на плату – Microsoft Visual Studio;
- термінал SSH Putty для віддаленого підключення;
- програма Fritzing – середовище для візуалізації з'єднання схеми;
- також, Cascade training GUI.

Після цього можна підключити одноплатний комп'ютер, налаштувати його та провести тестування програмного забезпечення.

Raspbian - це офіційна операційна система для Raspberry Pi, розроблена для цього пристрою на базі ARM версії Debian 8 Jessie. Вона включає в себе ряд програм за замовчуванням, таких як офісний пакет LibreOffice, веб-браузер, поштовий клієнт, легке оточення робочого столу та інструменти для навчання програмуванню [36]. Ми вибрали дистрибутив Raspbian через його повноту і необхідність інструментів для роботи з міні-комп'ютером. Після завантаження образу ОС важливо розархівувати та записати його на SD-карту об'ємом не менше 8 Гбайт.. Записати можна програмою Win32DiskImager.

На наступному етапі ми підключаємо необхідні компоненти, такі як серводвигун, мікрофон і динамік до Raspberry Pi 3B. Світлодіод під'єднуємо за допомогою джамперів: перша ніжка вільна, друга до 6-го піна (заземлення), третя до 1-го (живлення), четверта до 12-го піна (дані).

Серводвигун підключаємо так: жовтий дріт (дані) до 26-го піну, червоний (живлення) до 2-го піну, коричневий (заземлення) до 14-го піну [6] (рис. 3.12).

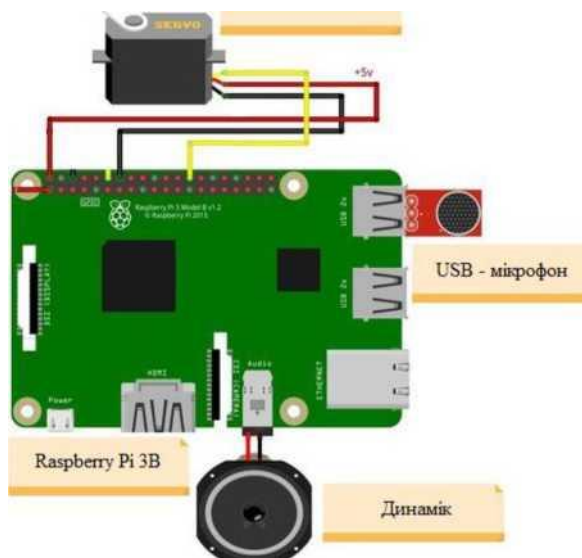


Рисунок 3.12 – Принципова схема пристрою на базі Raspberry Pi.

Після цього вставляємо SD-карту з образом операційної системи в Raspberry Pi, підключаємо монітор за допомогою HDMI-кабелю, клавіатуру, мишку та модуль камери до відповідних роз'ємів, які знаходяться між HDMI та Mini-Jack. Після цього підключаємо живлення, і плата автоматично вмикається. Після завантаження операційної системи необхідно налаштувати її [28]. Для запуску майстра налаштувань відкриваємо термінал, використовуючи комбінацію клавіш Ctrl+Alt+T, і вводим команду `sudo raspi-config` (`sudo` – права суперкористувача) (рис. 3.13).

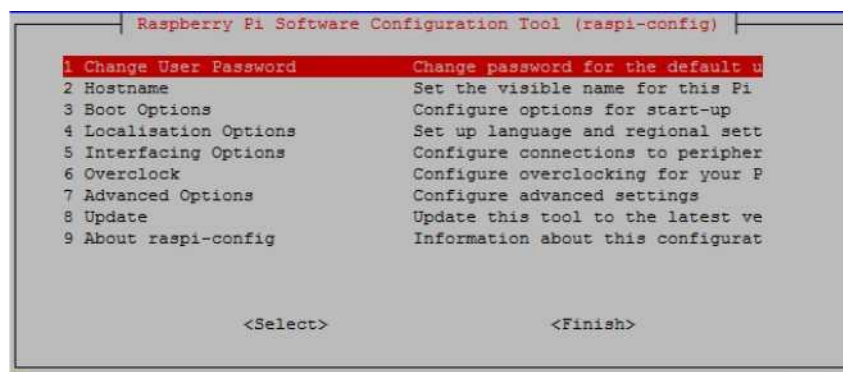


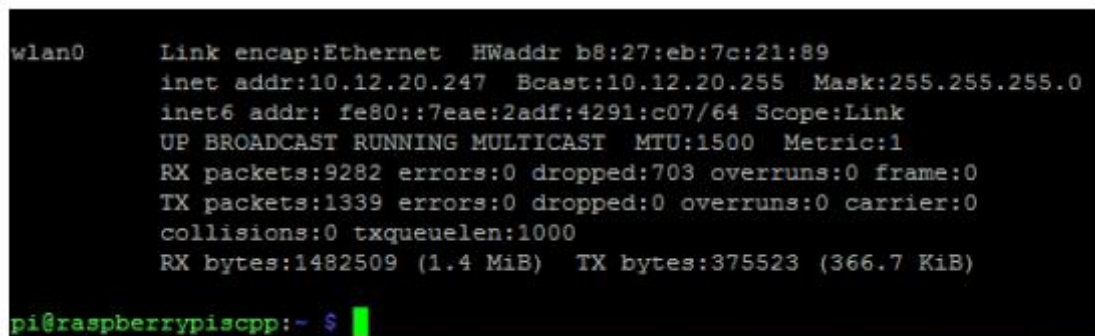
Рисунок 3.13 – Майстер налаштування ОС Raspbian

Спочатку ми встановимо пароль для облікового запису користувача. Для цього обираємо перший пункт, вводимо новий пароль двічі, і отримуємо повідомлення про успішну зміну паролю. Далі необхідно розширити кореневий розділ на всю доступну пам'ять карти. Для цього переходимо в Advanced Options - Expand Filesystem і натискаємо ОК. Потім ми змінюємо ім'я пристрою, переходячи до розділу Interfacing Options, обираємо пункт Camera і встановлюємо його в положення "Так". Далі, в розділі SSH також обираємо опцію "Так" і перезавантажуємо систему [34].

SSH – мережевий протокол, який дозволяє керувати віддалено операційною системою. Використання цього засобу значно спрощує управління комп'ютерами.

Raspberry Pi 3B дуже зручно користуватися віддалено, використовуючи програму PuTTY термінал. PuTTY – це програмне забезпечення для різноманітних протоколів віддаленого доступу (SSH, Telnet, rlogin та ін.). Можна управляти як звичайним комп'ютером, так і сервером.

Для початку, підключаємо Raspberry Pi до WiFi. В терміналі виконуємо команду ifconfig для перевірки IP-адреси нашого пристрою (див. рис. 3.14). Потім інсталуємо SSH-клієнт PuTTY на ПК, запускаємо програму, переходимо до вкладки Session і в поле IP address вводимо IP-адресу Raspberry Pi. Залишаємо порт за замовчуванням і, за бажанням, зберігаємо дану сесію: в полі Saved Sessions вказуємо ім'я сесії і натискаємо Save. Після цього вибираємо Load для запуску сесії, вводимо ім'я користувача (Rpi) і пароль.



```
wlan0    Link encap:Ethernet  HWaddr b8:27:eb:7c:21:89
        inet addr:10.12.20.247  Bcast:10.12.20.255  Mask:255.255.255.0
        inet6 addr: fe80::7eae:2adf:4291:c07/64 Scope:Link
        UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
        RX packets:9282 errors:0 dropped:703 overruns:0 frame:0
        TX packets:1339 errors:0 dropped:0 overruns:0 carrier:0
        collisions:0 txqueuelen:1000
        RX bytes:1482509 (1.4 MiB)  TX bytes:375523 (366.7 KiB)

pi@raspberrypi:~$
```

Рисунок 3.14 – IP-адреса Raspberry Pi 3B

Тепер необхідно оновити систему [29]. Для цього в терміналі набираємо `sudo apt update`, натискаємо Enter, Після цього ви побачите сповіщення про наявність пакетів для оновлення. Введіть команду `sudo apt upgrade`, щоб розпочати процес оновлення операційної системи (рис. 3.15). На цьому початкове налаштування пристрою завершено.

```
Игн http://mirrordirector.raspbian.org jessie/main Translation-ru_RU
Игн http://archive.raspberrypi.org jessie/ui Translation-ru_RU
Игн http://mirrordirector.raspbian.org jessie/main Translation-ru
Игн http://mirrordirector.raspbian.org jessie/main Translation-en
Игн http://archive.raspberrypi.org jessie/ui Translation-ru
Игн http://mirrordirector.raspbian.org jessie/non-free Translation-ru_RU
Игн http://mirrordirector.raspbian.org jessie/non-free Translation-ru
Игн http://archive.raspberrypi.org jessie/ui Translation-en
Игн http://mirrordirector.raspbian.org jessie/non-free Translation-en
Игн http://mirrordirector.raspbian.org jessie/rpi Translation-ru_RU
Игн http://mirrordirector.raspbian.org jessie/rpi Translation-ru
Игн http://mirrordirector.raspbian.org jessie/rpi Translation-en
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
All packages are up to date.
pi@raspberrypiscpp:~$ sudo apt upgrade
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Расчёт обновлений... Готово
обновлено 0, установлено 0 новых пакетов, для удаления отмечено 0 пакетов, и 0 п
акетов не обновлено.
pi@raspberrypiscpp:~$
```

Рисунок 3.15 – Завершення процесу оновлення програмного забезпечення

Після авторизації користувач отримує доступ до розширеного функціоналу системи. Для перевірки роботи камери потрібно підключити її до плати Raspberry Pi. Вимкніть живлення пристрою і відключіть його від сервера. На платі є спеціальний порт для підключення шлейфу камери, розташований біля HDMI-порту. Підніміть замок вгору і вставте шлейф у порт з синьою міткою з боку аудіороз'єму. Після цього опустіть замок, щоб зафіксувати шлейф.

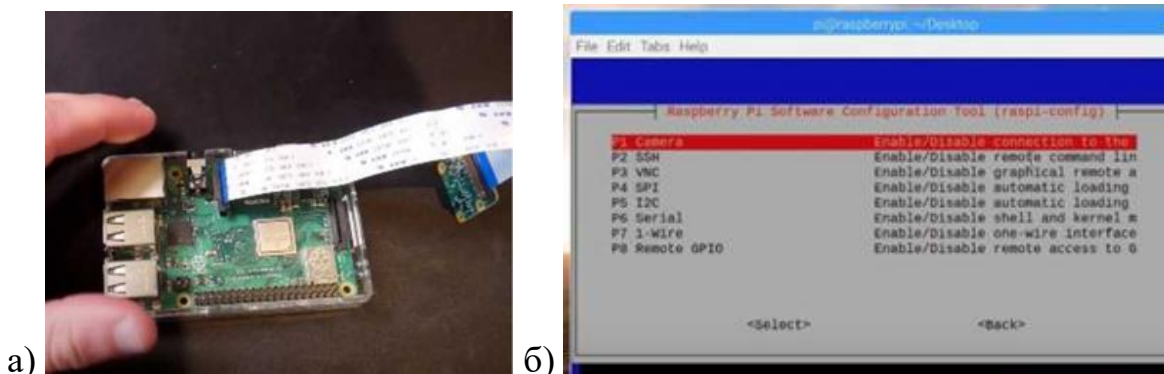


Рисунок 3.16 – Підключення шлейфу камери до плати – а) та активація камери – б)

Після цього підключіть плату до живлення і активуйте камеру через графічний інтерфейс (див. рис. 3.16а). Це також можна зробити через конфігураційний файл, аналогічно до налаштування VNC, введенням команди в терміналі `sudo raspi-config` і обирається Interface Options -> Cameras (Enable).

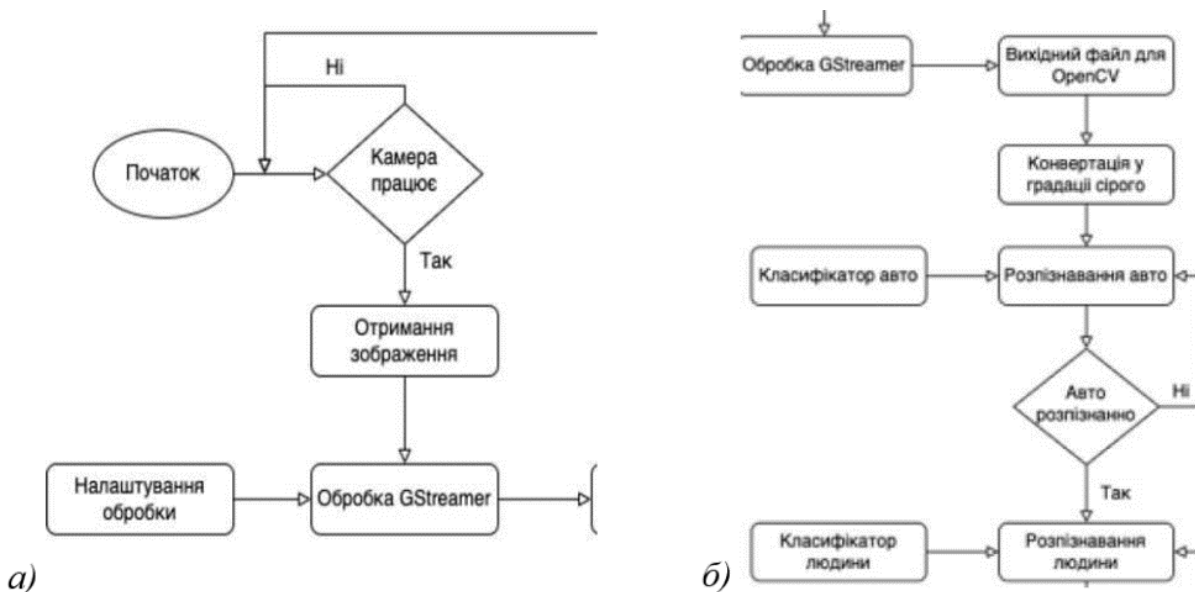


Рисунок 3.17 – Перевірка камери та обробка медіа-файлу – а) та перевірка користувача – б)

Для перевірки роботи камери можна створити невеликий скрипт на Python, в якому необхідно створити файл і вписати такий код:

```

from picamera import PiCamera
from time import sleep
camera = PiCamera()
camera.start_preview(alpha=192)
sleep(1)
    
```

```
camera.capture («/home / pi / Desktop / test.jpg»)  
camera.stop_preview()
```

Після запуску цього скрипта, камера має зробити фотографію і зберегти її на робочому столі (рис. 3.16б).

Тепер можна випробувати камеру з розпізнаванням об'єктів. Для цього буде написаний код для візуального спостереження, який використовує готовий класифікатор OpenCV. Після подачі живлення на Raspberry Pi можна запустити написаний код. Важливо, щоб камера була підключена до плати, через яку буде йти відеопотік для перевірки її працездатності. Після цього відеопотік пройде через фреймворк GStreamer з налаштуваннями для обробки потоку (рис. 3.18).

```
# Face rec cascade  
face_cascade = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')  
# To capture video from Raspbi-cam.  
cap = cv2.VideoCapture(0)  
  
while True:  
    # Read the frame  
    _, img = cap.read()  
    # Convert to grayscale  
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)  
    # Detect the faces  
    faces = face_cascade.detectMultiScale(gray, 1.1, 4)  
    # Draw the rectangle around each face  
    for (x, y, w, h) in faces:  
        cv2.rectangle(img, (x, y), (x+w, y+h), (255, 0, 0), 2)  
    # Display  
    cv2.imshow('img', img)  
    # Stop if escape key is pressed  
    k = cv2.waitKey(30) & 0xff  
    if k==27:  
        break  
# Release the VideoCapture object  
cap.release ()
```

Рисунок 3.18 – Код тестування розпізнавання відеосигналу з камери

Зображення перетворюється у відтінки сірого кольору, і починається процес визначення об'єктів у потоковому відео. Кожен об'єкт перевіряється на відповідність за допомогою власного класифікатора, призначеного для цього конкретного типу об'єкта.

Якщо об'єкт не було знайдено, процес повторюється для кожного встановленого класифікатора. Після обробки кожного кадру, зображення нормалізується і вихідне відео формується у відповідному форматі з обробленими об'єктами, які визначені на медіа-файлі.

Оскільки відео складається з послідовних кадрів, ми перетворюємо їх у градації сірого кольору за допомогою бібліотеки OpenCV і здійснюємо виявлення заданого об'єкту на кожному кадрі. Після запуску написаної програми ми отримуємо відповідний результат [35].

3.2 Аналіз можливостей щодо розширення функціоналу розроблюваного модуля

З результатів кадрової обробки модулем візуального спостереження можна виділити наступні позитивні аспекти: успішне розпізнавання рухомих об'єктів, стабільність алгоритму при змінах оточення та висока швидкодія.



Рисунок 3.19 – Приклад роботи покадрової обробки зображення камери при візуальному спостереженні



Рисунок 3.20 – Приклад роботи оцінки фону на основі визначення мінімуму або максимуму інтенсивності пікселів з камери при візуальному спостереженні

З результатів аналізу кадрової обробки видно наступні недоліки: об'єкти, які швидко рухаються, залишають за собою "шлейф"; об'єкти з однорідним забарвленням, наприклад, автомобілі, не виділяються повністю.

Якщо об'єкт не проявляє руху на наступному кадрі, він ідентифікується як фон або нерухомий об'єкт.



Рисунок 3.21- Візуалізація порогу накладеного на кадр з камери

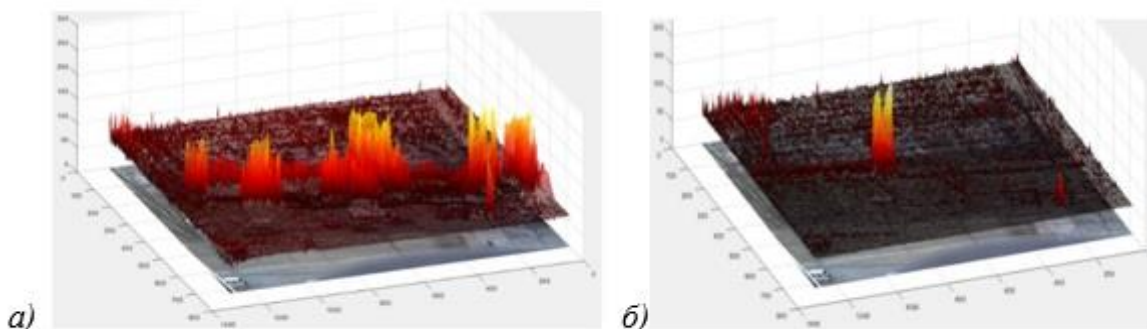


Рисунок 3.22 – Модуль різниці поточного кадру та накопиченого фону до накопичення фону – а) та після накопичення фону – б)

На рис. 3.23 показано кадр з камери, де відображено значення модуля різниці між поточним кадром та накопиченим фоном. Кольори підсвічення варіюються від прозорого через червоний до білого.

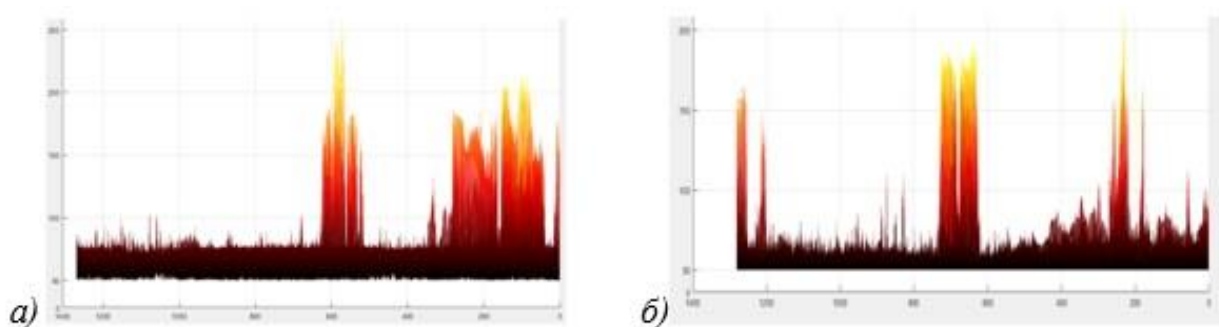


Рисунок 3.23 – Модуль різниці поточного кадру та накопиченого фону до накопичення фону – а) та після накопичення фону – б)

На рис. 3.22–3.23 провівши аналіз зміни рівня шуму після накопичення фону, було визначено, що значення порогу 20 найбільш підходить для ефективної роботи програми у визначеній сцені.

Необхідно продовжувати вдосконалювати розроблений апаратно-програмний модуль візуального спостереження, розглядаючи його потенціал у майбутньому, додавання нових модулів і вдосконалення програмного забезпечення для відповідності вимогам задач.

Насамперед, можна розпочати з легкого захисного кейсу, який захистить камеру від зовнішніх факторів, таких як вода, пил і бруд, а також від невеликих ударів. Найпростіший спосіб - це виготовлення корпусу на 3D принтері, ідеально підходять до розмірів всіх компонентів. (рис. 3.24).



Рисунок 3.24 – Варіант вдосконалення спроектованого апаратно-програмного модулю візуального спостереження в окремому корпусі

У сучасних умовах, з агресією з боку країни-терориста проти України, цей апаратно-програмний модуль може бути вдосконалений для використання у проектах з дронами, з метою розпізнавання ворожої піхоти та техніки за допомогою вказаного програмного коду. Це дозволить збільшити ефективність та швидкість реакції Збройних Сил України на окупантів у складних умовах, де їх важко виявити (рис. 3.25).



Рисунок 3.25 – Майбутні сфери застосування розробки на дроні

Доброю ідеєю буде створення портативного джерела живлення для плати, щоб мати можливість працювати з нею не лише вдома, а й у будь-якому іншому місці (рис. 3.26).

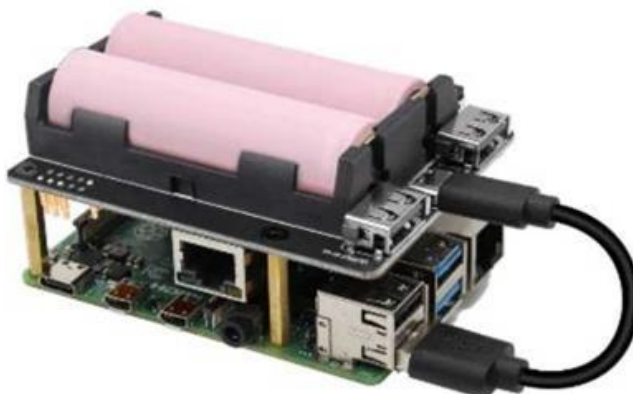


Рисунок 3.26 – Переносне живлення для плати

Ще одним кроком у розширенні функціоналу розроблюваного модуля є камери з функцією розпізнавання зображень спостереження, а також запуск

проекту зі встановлення у громадських місцях кнопок екстреного виклику поліції.

3.3 Аналіз енергоефективності пристрою

Простіші камери нижчої якості апаратно-програмного модулю візуального спостереження можуть споживати від 2 до 4 Вт, тоді як високоякісні та багатофункціональні моделі можуть споживати від 10 до 15 Вт. Важливо враховувати, що споживання електроенергії також залежить від режиму запису: постійний запис споживає більше енергії, ніж запис лише під час виявлення руху.

Для визначення споживаної потужності всієї системи камер необхідно врахувати кілька факторів, таких як кількість камер, їх тип, додаткові функції та інші підключені пристрої. Окрім індивідуального споживання кожною камерою, важливо враховувати інші елементи системи, як-от мережеві відеореєстратори (NVR) або сервери зберігання даних, які можуть споживати від 20 до 100 Вт залежно від їх характеристик та функцій.

Додатково до камер система візуального спостереження можна включати монітори для відображення зображень у реальному часі або пристрої живлення, такі як джерела живлення чи резервні батареї. Хоча споживання цих елементів зазвичай менше, ніж у камер або записуючих пристроїв, їх також слід враховувати.

Для розрахунку споживання енергії підсилювача камери візуального спостереження можна використовувати формулу:

$$P = V \times I, \quad (3.23)$$

де P – потужність у ватах, V – напруга джерела живлення, а I – струм в амперах.

Однак цей розрахунок є приблизним і може змінюватися залежно від фактичних умов експлуатації камери.

Наведемо приклад розрахування питання для камер візуального спостереження.

Початкові дані:

- кількість камер відеоспостереження – 4;
- відстань до камер 50 метрів (будемо вважати, що всі камери розташовані в безпосередній близькості один від одного);
- струм споживання кожної камери 150 мА;
- напруга живлення камери відеоспостереження 12В +/- 10%.

Розрахунок

Визначаємо сумарний струм споживання :

$$I = 150 \cdot 4 = 600 \text{ мА} = 0,6 \text{ А.}$$

Вибираємо відповідний блок живлення, дивимося параметри його вихідного напруги, наприклад 12,6+/-0,2В.

Визначаємо мінімальний рівень напруги блоку $12,6 - 0,2 = 12,4\text{В}$ і камери $12\text{В} - 10\% = 10,8$.

Максимально допустимий рівень втрат на складі:

$$U = 12,4 - 10,8 = 1,6 \text{ В.}$$

Розраховуємо максимально можливий опір лінії :

$$R = U / I = 1,6 / 0,6 = 2,7 \text{ Ом.}$$

Загальна довжина дроту $L = 50 \cdot 2 = 100$ метрів.

Максимально допустимий питомий опір:

$$R_{\text{пит}} = R / L = 2,7 / 100 = 0,027 \text{ Ом/метр.}$$

Висновки до розділу 3

В розділі розглянуто вимоги користувача до системи візуального спостереження. Наголошено, що функціональні вимоги апаратно-програмного модулю візуального спостереження визначаються тим, які завдання він повинен вирішувати і що саме має робити. А нефункціональні

вимоги визначають характеристики якості апаратно-програмного модулю візуального спостереження, такі як доступність, зручність установки, продуктивність, надійність та умови експлуатації.

Наведено математичну модель апаратно-програмного модулю візуального спостереження, що описана за допомогою методів статистичного моделювання, таких як Вінерівські та узагальнені Вінерівські процеси. Приведена діаграма класів ергономіки та опис модулів апаратно-програмного модулю візуального спостереження. За допомогою дерева цілей зроблено оцінку основного призначення розроблюваного проєкту.

Було розглянуто основні вимоги до апаратно-програмного комплексу на Raspberry Pi і представлено опис принципової схеми пристрою. Описано блок-схему алгоритму та розглянуто принцип його функціонування.

Окремо розглянуто можливості щодо розширення функціоналу розроблюваного модуля за рахунок монтажу його в корпусі, розміщення на дроні та розроблення переносного живлення для плати. Оцінка енергоефективності пристрою засвідчує значну доцільність даної системи візуального спостереження для широкого використання.

ВИСНОВКИ

На початку дослідження було розглянуто основні принципи візуального спостереження та перелічено необхідне обладнання систем відеоспостереження, яке незалежно від їх типу (аналогова, гібридна або цифрова), включає відеокамери для збору відеоінформації, пристрої обробки та запису (DVR, NVR, плати відеозахоплення, сервери з прикладним ПЗ), архівуюче обладнання та пристрої для відображення відеоінформації (монітори). Було проведено характеристику найбільш поширених одноплатних комп'ютерів. Для кожного досліджуваного типу було наведено характеристику та вказано переваги і обмеження. Також було охарактеризовано модульні камери для візуального спостереження. Окремо було проаналізовано типи камер, які найбільш підходять для внутрішнього візуального спостереження. Наголошено, що модульні камери (безкорпусні) – це одноплатні камери з мініатюрним об'єктивом, які можуть бути вбудовані в різні предмети інтер'єру або пристрої.

В аналітичній частині роботи було зроблено обґрунтування підбору мікропроцесора, мови програмування та необхідних складових проєкту. Зокрема, було обґрунтовано використання мікропроцесора Raspberry Pi 4, який забезпечує необхідну продуктивність для виконання завдань обробки відеоданих. Було наведено функціональні можливості Visual Studio Code як середовища розробки та мови програмування Python, яка відома своїми широкими можливостями для візуалізації даних та легкістю вивчення і використання. Наголошено на таких функціональних перевагах даної мови програмування, як можливість інтеграції з численними бібліотеками для аналізу та візуалізації даних, зокрема Matplotlib, Seaborn та Plotly.

Було перелічено вимоги користувача до системи візуального спостереження та наведено математичну модель апаратно-програмного модуля візуального спостереження, що описана за допомогою методів статистичного моделювання, таких як Вінерівські та узагальнені Вінерівські

процеси. Було розроблено діаграму класів ергономіки та дано опис модулів апаратно-програмного модуля візуального спостереження. Було спроектовано і описано блок-схему роботи алгоритму та розписано принцип його роботи, що дозволяє забезпечити надійну та ефективну роботу системи. Проведено тестування проекту та розглянуто можливості розширення функціоналу розроблюваного модуля, що включають додавання нових функцій та поліпшення існуючих для підвищення ефективності та надійності системи.

Таким чином, проведене дослідження дозволило створити ефективний та надійний апаратно-програмний модуль візуального спостереження, який відповідає сучасним вимогам безпеки та моніторингу. Описані функціональні можливості та обґрунтування вибору технологій забезпечують високу якість роботи системи та її відповідність потребам користувачів. Впровадження даного модуля дозволяє значно підвищити рівень безпеки на об'єктах різного типу, забезпечуючи безперервний моніторинг і своєчасне реагування на можливі інциденти.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Азарян А. А., Карабут Н. О., Козицова Т. П. Основи алгоритмізації та програмування: Навчальний посібник. Кривий Ріг: Вид-во ОктапПринт, 2014. 308 с.
2. Алгоритми та структури даних / уклад. О. В. Щербаков, Ю. Е. Парфьонов, В. М. Федорченко. Харків: ХНЕУ ім. С. Кузнеця, 2017. 58 с.
3. Архітектура комп'ютерних систем: конспект лекцій для студентів усіх форм навчання з курсу «Архітектура комп'ютерних систем» / Укладачі: Голотенко О. С. Тернопіль: Вид-во ТНТУ імені Івана Пулюя, 2016. 120 с.
4. Бавико О. Є. Синхронізація розвитку ринку інформаційно-комунікаційних технологій в Україні з глобальними трендами. *Маркетинг і менеджмент інновацій*. 2018. № 1. С. 272–282.
5. Варакін М. В. Розробка мобільних додатків під Android. УЦ «Спеціаліст» при МГТУ ім. Н. Е. Баумана, 2012. С. 40–48.
6. Ганжела С. І. Основи інформатики з елементами програмування та сучасні інформаційні технології навчання. Ч. II. Елементи програмування. Кропивницький: РВВ ЦДПУ ім. В. Винниченка, 2017. 61 с.
7. Євсєєв В. В. Проєктування мобільних роботів на базі одноплатних комп'ютерів (Raspberry Pi и мови Python 3.6) // І. Ш. Невлюдов, А. О. Андрусевич, В. В. Євсєєв: підручник. Харків, 2020. 257 с.
8. Кондрат В. В., Костенко О. І., Корнієнко О. В. Аналіз оптико-електронних засобів розвідки та напрямки їх удосконалення з метою підвищення ефективності бойового застосування ОВТ, *Збірник наукових праць Харківського національного університету Повітряних Сил*. 2018, № 2 (56), С. 66–71. doi: 10.30748 / zhups. 2018.56.08.
9. Коршець, О., Горбенко, В. (2023). Уроки застосування безпілотних літальних апаратів у російсько-українській війні. *Повітряна міць України*, 1 (4), 9–17. <https://doi.org/10.33099/2786-7714-2023-1-4-9-17/>

10. Крилов В. К. Аналіз методів ідентифікації та моделювання динамічних змін силуетів користувачів. Кваліфікаційна робота освітнього рівня «Магістр» // Крилов Вадим Кирилович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра кафедра комп'ютерних наук, група СНм – 61. Тернопіль, 2023. С. 107.

11. Кузь М. В., Соловко Я. Т. Методологія формування узагальненого критерію якості програмного забезпечення в умовах невизначеності. *Вісник Вінницького політехнічного інституту*. 2015. №5. С. 104–107.

12. Луцків А. М. Архітектури високопродуктивних систем опрацювання великих даних. *Збірник тез доповідей Науково-технічна VI Науково-технічна конференція «Інформаційні моделі, системи та технології»*, 12–13 грудня 2018 року. Т.: ТНТУ, 2018. С. 75.

13. Нікіфоров М., Пампуха І. Обґрунтування типу та вимог до оптикоелектронних систем в інтересах виконання завдань розвідки та охорони об'єктів, *Збірник наукових праць ВІКНУ*, 2017, № 55, С. 71–81.

14. Основи візуального спостереження: практ. посіб / С. С. Чернявський, М. Л. Грібов, А. А. Вознюк, А. А. Небитов. Київ, Норма права, 2024. 112 с.

15. Подчашинський Ю. О., Лугових О. О., Чепюк, Л. О. Аналіз методів обробки відеозображень з вимірювальною інформацією, отриманих від тепловізора спектральної камери. *Технічна інженерія*, 2023. № 1 (91), С. 214–221. [https://doi.org/10.26642/ten-2023-1\(91\)-214-221](https://doi.org/10.26642/ten-2023-1(91)-214-221).

16. Сервецький І. Поняття, зміст та правові засади візуального спостереження. *Наукові праці Міжрегіональної Академії управління персоналом. Юридичні науки*, 2022. №1 (60), С. 36–44. <https://doi.org/10.32689/2522-4603.2021.1.7>

17. Becker Ronald F. Dutelle Aric W. Criminal Investigation / F. Becker Ronald, W. Dutelle Aric. – Burlington, MA: Jones & Bartlett Learning, 2012. – 576 с.

18. S. Bosse, D. Maniry, T. Wiegand, and W. Samek, «A deep neural network for image quality assessment,» in 2016 IEEE International Conference on Image Processing (ICIP), pp. 3773–3777, 2016. doi: 10.1109 / ICIP. 2016.7533065.
19. P. P. Chavan, B. S. Rani, M. Murugan, and P. Chavan, «A Novel Image Compression Model by Adaptive Vector Quantization: Modified Rider Optimization Algorithm,» *Sadhana – Acad. Proc. Eng. Sci.*, vol. 45, no. 1, 2020, doi: <https://doi.org/10.1007/s12046-020-01436-9>.
20. K. Ding, K. Ma, S. Wang, and E. P. Simoncelli, «Comparison of Full-Reference Image Quality Models for Optimization of Image Processing Systems,» *Int. J. Comput. Vis.*, vol. 129, no. 4, pp. 1258–1281, 2021, doi: <https://doi.org/10.1007/s11263-020-01419-7>.
21. Image Compression Using Deep Learning: Methods and Techniques. (2022). *Iraqi Journal of Science*, 63 (3), 1299–1312. <https://doi.org/10.24996/ijs.2022.63.3.34>.
22. Feng, Y., Zhang, Y., Zhou, Z. et al. Memristor-based storage system with convolutional autoencoder-based image compression network. *Nat Commun* 15, 1132 (2024). <https://doi.org/10.1038/s41467-024-45312-0>.
23. A. J. Hussain, A. Al-Fayadh, and N. Radi, Image compression techniques: A survey in loss-less and lossy algorithms, vol. 300. 2018. <https://doi.org/10.1016/j.neucom.2018.02.094>.
24. Jalilian E, Hofbauer H, Uhl A. Iris Image Compression Using Deep Convolutional Neural Networks. *Sensors*. 2022; 22 (7): 2698. <https://doi.org/10.3390/s22072698> / Ma, S., Zhang, X., Jia, C., Zhao, Z., Wang, S., & Wang, S. (2019). Image and Video Compression With Neural Networks: A Review. *IEEE Transactions on Circuits and Systems for Video Technology*, 30, 1683–1698. doi: 10.1109 / TCSVT. 2019.2910119.
25. Hess Kären M. Criminal Investigation / M. Hess Kären, H. Orthmann Christine. – Clifton Park, NY: Delmar, 2013. – 761 с.

26. Raspberry Pi – перший запуск: [Електронний ресурс] – Режим доступу: <https://gekelectronics.org/raspberry-pi/raspberry-pi-pervyj-zapusk.html>
27. R. K. Netalkar, H. Barman, R. Subba, K. V. Preetam, and U. S. N. Raju, «Distributed Com-pression and Decompression for Big Image Data: LZW and Huffman Coding,» *J. Electron. Imaging*, vol. 30, no. 5, 2021, doi: <https://doi.org/10.1117/1.JEI.30.5.053015>.
28. Atta Rawheyaa E., Kasem Hossam M., Attia Mahmoud. A Comparison Study for Image Compression Based on Compressive Sensing [J]. Eleventh international conference on graphics and image processing (ICGIP 2019), 2020,11373.
29. Sadeeq, H. T., Hameed, T. H., Abdi, A. S., & Abdulfatah, A. N. (2021). Image Compression Using Neural Networks: A Review. *International Journal of Online and Biomedical Engineering (iJOE)*, 17 (14), pp. 135–153. <https://doi.org/10.3991/ijoe.v17i14.26059>.
30. Srihari K., Ramesh R., Udayakumar E., Dhiman G. An Innovative Approach for Face Recognition Using Raspberry Pi. *Artificial Intelligence Evolution*. 2020. V. 1 (2). P. 63–144.
31. A. J. Qasim, R. Din, F. Qasim, and A. Alyousuf, «Review on Techniques and File Formats of Image Compression,» *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 2, pp. 602–610, 2020, doi: <https://doi.org/10.11591/eei.v9i2.2085>.
32. X. Zhang, W. Lin, and S. Wang, «Fine-Grained Quality Assessment for Compressed Images,» *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1163–1175, 2019, doi: <https://doi.org/10.1109/TIP.2018.2874283>.
33. Shupeizhang, Chenqiu Zhao, and Anup Basu. 2024. Principal Component Approximation Network for Image Compression. *ACM Trans. Multimedia Comput. Commun. Appl.* 20, 5, Article 121 (May 2024), 20 pages. <https://doi.org/10.1145/3637490>.

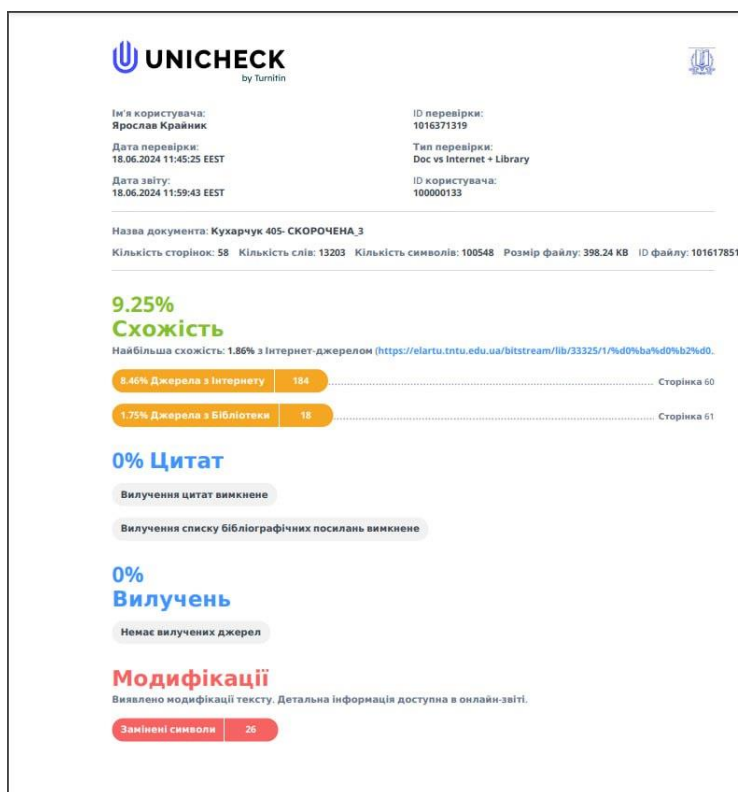
34. F. Yang, L. Herranz, Y. Cheng, and M. G. Mozerov, «Slimmable Compressive Autoencoders for Practical Neural Image Compression,» pp. 4998–5007.
35. Yibo Yang, Stephan Mandt and Lucas Theis (2023), «An Introduction to Neural Data Compression», Foundations and Trends in Computer Graphics and Vision: Vol. 15, No. 2, pp 113–200. doi: 10.1561 / 0600000107.

ДОДАТОК А
Довідка
про перевірку на унікальність пояснювальної записки
бакалаврської кваліфікаційної роботи на тему:
«Апаратно-програмний модуль візуального спостереження»

студента спеціальності 123 «Комп'ютерна інженерія», 405 групи
Кухарчук Ростислав Денисович
прізвище, ім'я, по-батькові

Перевірку тексту здійснено сервісом: онлайн-сервіс Unicheck

Результат перевірки тексту бакалаврської кваліфікаційної роботи: схожість складає 9,25%.



Здобувач:

_____ Р. Д. Кухарчук
підпис ініціали, прізвище

Дата: «___» _____ 2024 р.

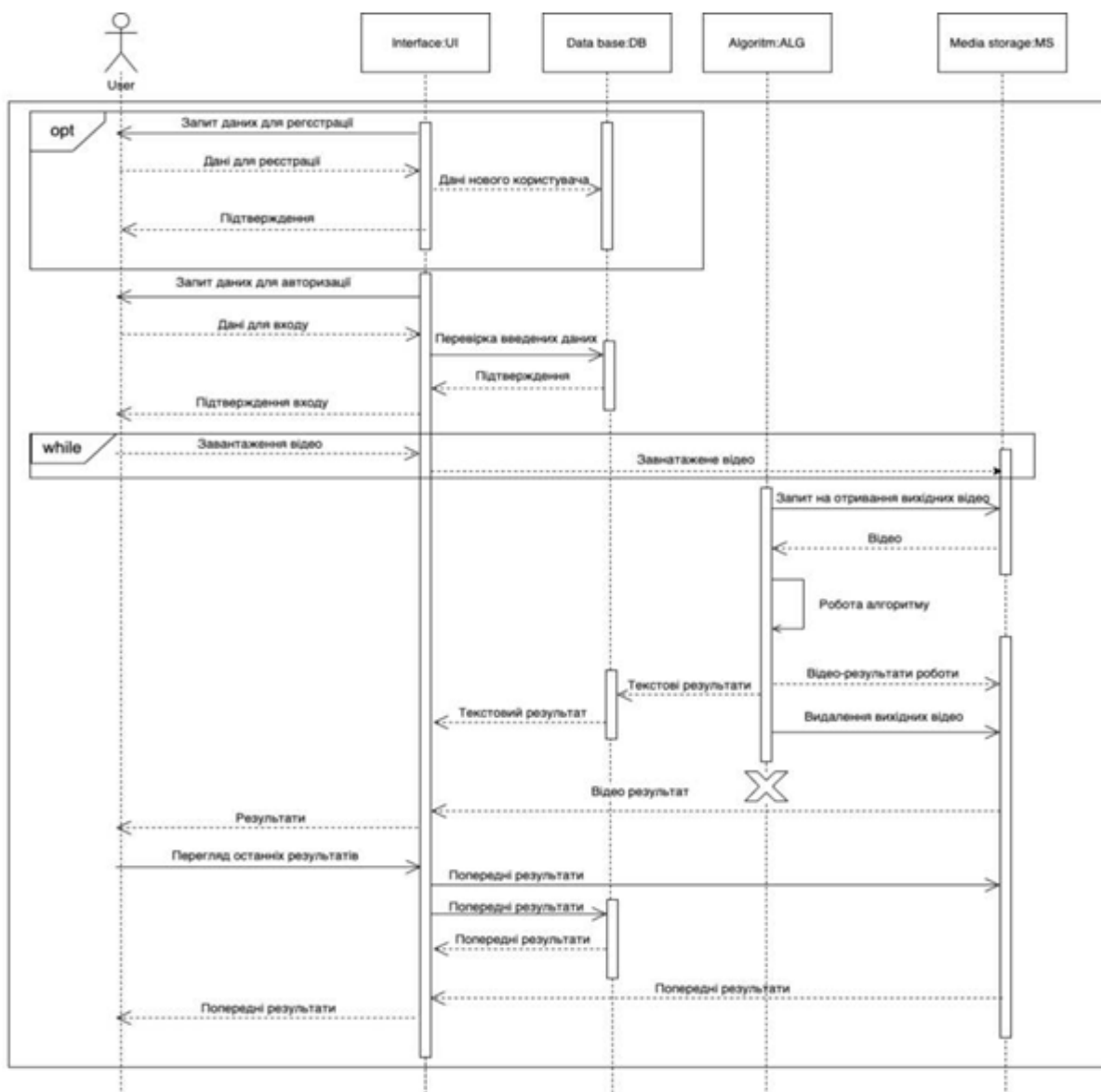
Керівник:

канд. техн. наук, доцент

_____ Я. М. Крайник
підпис ініціали, прізвище

ДОДАТОК Б

Діаграми послідовностей



ДОДАТОК В

Код програми покадрової обробки зображення при візуальному спостереженні

```
vebca = webcam (1);
vebca. Resolution='1280x720';
gray_range = 18;
medfilt_m = 4;
medfilt_n = 4;
test_size = snapshot (vebca);
size_1 = size (test_size);
prev_frame = zeros (size_1 (1), size_1 (2));
disp_img = zeros (size_1 (1), size_1 (2));
disp_img_red = test_size;
disp_img_filtered = disp_img;
img = disp_img;
sum_img = zeros (size_1 (1), size_1 (2));
disp_img_filtered_2x2 = zeros (size_1 (1), size_1 (2));
disp_img_filtered_1 = zeros (size_1 (1), size_1 (2));
figure (1) subplot (2,2,1); P (1) = imshow (img);
title('Зображення з камери') subplot (2,2,2);
P (2) = imshow (disp_img_red);
title('Зображення з камери з виділенням рухомих об'єктів') subplot (2,2,3);
P (3) = imshow (disp_img);
title('Оброблене зображення') subplot (2,2,4);
P (4) = imshow (disp_img_filtered);
title('Оброблене зображення з медіанною фільтрацією') while (1) img = snapshot
(vebca);
current_frame = im2gray (img);
for i = 1: size_1 (1)
for j = 1: size_1 (2)
if current_frame (i, j) <= (prev_frame (i, j) + gray_range) &&
current_frame (i, j) >= (prev_frame (i, j) - gray_range)
sum_img (i, j) = 0;
else
sum_img (i, j) = 255;
end
end
end
disp_img_filtered(:, :, 1) = medfilt2 (sum_img, [medfilt_m medfilt_n]);
disp_img_red = img;
disp_img_red (disp_img_filtered == 255) = 255;
prev_frame = current_frame;
disp_img = sum_img;
sum_img = zeros (size_1 (1), size_1 (2));
set (P(1), 'CData', img);
set (P(2), 'CData', disp_img_red);
set (P(3), 'CData', disp_img);
set (P(4), 'CData', disp_img_filtered);
end

using System;
using System. Collections. Generic;
using System. Drawing;
using System. Windows. Forms;
using Emgu. CV;
using Emgu. CV. Structure;
```

```
using Emgu. CV. CvEnum;  
using System. IO;  
using System. Diagnostics;  
using System. Data. SqlClient;  
using System. Data;  
using System. Net. Mail;  
using System. Net;  
using System. Net. Mime;  
namespace RecSystem  
{  
public partial class FrmPrincipal: Form
```