Хмельницький національний університет
Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем

КВАЛІФІКАЦІЙНА РОБОТА

бакалавр
Освітній рівень

Програмно-апаратний засіб розпізнавання облич для системи домашньої безпеки
Назва теми

КвРКІ 210359.21.03.16 ПЗ
Шифр

Галузь знань *12 «Інформаційні технології»*
Шифр, назва

Спеціальність *123 «Комп'ютерна інженерія»*
Шифр, назва

Освітня програма *«Комп'ютерна інженерія та програмування»*
Назва

Виконав: студент IV курсу, група КІ2-21-3 _____ Микита ВІЛЬЧУК
Підпис Ініціали, прізвище

Керівник _____ Василь ЯЦКІВ
Підпис, дата Ініціали, прізвище

Нормоконтролер _____ Тетяна КИСІЛЬ
Підпис, дата Ініціали, прізвище

До захисту допускаю:
зав. кафедри комп'ютерної інженерії та інформаційних систем _____ Ольга ПАВЛОВА
Підпис Ініціали, прізвище

«___» _червня_ 2025 р.

Хмельницький 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Комп'ютерної інженерії та інформаційних систем

Освітній рівень бакалавр

Галузь знань 12 Інформаційні технології

Спеціальність 123 Комп'ютерна інженерія

Освітня програма «Комп'ютерна інженерія та програмування»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА

" 10 " 01 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

Вільчуку Микиті Валерійовичу

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Програмно-апаратний засіб розпізнавання облич для системи домашньої безпеки

Керівник проекту (роботи) Василь Яцків, д.т.н., професор

Прізвище, ім'я, по батькові, науковий ступінь, вчене звання

Затверджена наказом ректора університету від 07.02.2025 р. № 23

2. Строк подання студентом проекту (роботи) на кафедру 01.06.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на кваліфікаційну роботу

4. Зміст пояснювальної записки (перелік питань, які потрібно розробити)

Аналіз сучасного стану технологій розпізнавання облич для систем безпеки

Розробка програмно-апаратного засобу розпізнавання облич

Впровадження та оцінка ефективності системи

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслень)

Структура програмного забезпечення

Діаграма сценарію контролю доступу

Електрично-принципова схема

6. Консультанти розділів дипломного проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Тетяна КИСІЛЬ, доцент кафедри КІІС		
Антиплагіат	Андрій НІЧЕПОРУК, доцент кафедри КІІС		

7. Дата видачі завдання «_10_»_____01_____2025 р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) дипломного проекту (роботи)	Термін виконання етапів проекту (роботи)	Примітка
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи з керівником	10.01.2025	виконано
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.02.2025	виконано
3	Робота над розділом 1 – дослідження предметної області та постановка задачі	01.03.2025	виконано
4	Робота над розділом 2 – вибір компонентів для проєктування програмно-апаратного засобу для розпізнавання облич для системи домашньої безпеки	01.04.2025	виконано
5	Робота над розділом 3 – проєктування програмно-апаратного засобу розпізнавання облич для системи домашньої безпеки	29.04.2025	виконано
6	Оформлення пояснювальної записки згідно вимог	25.05.2025	виконано
7	Попередній захист ВКР	26.05.2025	виконано
8	Захист ВКР на засіданні ЕК	Червень 2025 року	

Студент _____ Підпис Микита ВІЛЬЧУК Ініціали, прізвище

Керівник роботи _____ Підпис Василь ЯЦКІВ Ініціали, прізвище

№ рядка	формат	Позначення	Найменування	Кіл. листів	№ екз	Примітка
			Текстові документи			
1		КвРКІ 210359.21.03.16 ПЗ	Пояснювальна записка	60		
			Графічні матеріали			
2		КвРКІ 210359.21.03.16 Е1	Структура програмного забезпечення	1		
3		КвРКІ 210359.21.03.16 Е2	Діаграма сценарію контолю доступу	1		
4		КвРКІ 210359.21.03.16 Е3	Елетрично-принципова схема	1		

КвРКІ 210359.21.03.16 ВП

Зм	Арк	№ докум	Підпис	Дата
Розробив		Вільчук		
Перевір.		Яцків		
Н. контр.		Кисіль		19.06.25
Затв.		Павлова		19.06.25

Відомість проекту

Літера	Аркуш	Аркушів
У	1	1

ХНУ, КІ2-21-3

АНОТАЦІЯ

Тема кваліфікаційної роботи: «Програмно-апаратний засіб розпізнавання облич для системи домашньої безпеки».

Автор роботи: Микита ВІЛЬЧУК.

Керівник роботи: Яцків Василь Васильович.

Пояснювальна записка: 60 с., 33 рис., 1 табл., 3 дод., 29 джерел.

Графічна частина: 3 креслення.

ПРОГРАМНО-АПАРАТОНИЙ ЗАСІБ, АРХІТЕКТУРА, РОЗПІЗНАВАННЯ, КАМЕРА, БЕЗДРОТОВИЙ ЗВ'ЯЗОК.

Метою даної статті є розробка програмно-апаратного засобу для розпізнавання облич у системі домашньої безпеки з використанням одноплатного комп'ютера Raspberry Pi та камерного модуля. Основна увага приділяється підвищенню точності розпізнавання, забезпеченню стабільної роботи системи в умовах реального часу та впровадженню механізмів захисту від несанкціонованого доступу. Система покликана забезпечити ідентифікацію користувачів, ведення журналу подій та сповіщення власника у випадку виявлення сторонніх осіб.

Об'єктом дослідження є програмно-апаратна система для розпізнавання облич у контексті безпеки приватного житла.

Предметом дослідження є методи захоплення, обробки та аналізу зображень облич, а також технології реалізації на платформах із обмеженими обчислювальними ресурсами, таких як Raspberry Pi.

Під час проведення даного дослідження було застосовано метод систематичного огляду наукових і технічних джерел з метою аналізу існуючих підходів до розпізнавання облич, вибору оптимальних моделей глибинного навчання, а також вивчення особливостей інтеграції модулів відеоспостереження з апаратними платформами типу Raspberry Pi.

30.05.2025

Підпис студента

Дата

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

AES – Advanced Encryption Standard (Стандарт розширеного шифрування).

CNN – Convolutional Neural Network (Конволюційна нейронна мережа).

CPU – Central Processing Unit (Центральний процесор).

CSI – Camera Serial Interface (Інтерфейс послідовного підключення камери).

CUDA – Compute Unified Device Architecture (Архітектура обчислень для пристроїв NVIDIA).

DLib – Бібліотека машинного навчання та комп'ютерного зору.

GPU – Graphics Processing Unit (Графічний процесор).

HOG – Histograms of Oriented Gradients (Гістограми орієнтованих градієнтів).

IR – Infrared (Інфрачервоний).

LBP – Local Binary Patterns (Локальні бінарні шаблони).

LFW – Labeled Faces in the Wild (Набір даних для розпізнавання облич).

MQTT – Message Queuing Telemetry Transport (Протокол передачі повідомлень).

MTCNN – Multi-task Cascaded Convolutional Networks (Багатозадачні каскадні конволюційні мережі).

NPU – Neural Processing Unit (Нейронний процесор).

OpenCV – Open Source Computer Vision Library (Бібліотека комп'ютерного зору з відкритим кодом).

PoE – Power over Ethernet (Живлення через Ethernet).

PyTorch – Бібліотека для машинного навчання з відкритим кодом.

SIFT – Scale-Invariant Feature Transform (Перетворення ознак, інваріантне до масштабу).

SQLite – Легка система керування базами даних.

TensorFlow – Платформа для машинного навчання з відкритим кодом.

ToF – Time-of-Flight (Технологія вимірювання часу прольоту для 3D-камер).

TOPS – Tera Operations Per Second (Тераоперацій за секунду).

TPU – Tensor Processing Unit (Тензорний процесор).

UML – Unified Modeling Language (Уніфікована мова моделювання).

USB – Universal Serial Bus (Універсальна послідовна шина).

VGG – Visual Geometry Group (Архітектура нейронної мережі).

Wi-Fi – Wireless Fidelity (Бездротова мережа).

2D – Two-Dimensional (Двовимірний).

3D – Three-Dimensional (Тривимірний).

ВСТУП

Актуальність теми. У сучасному світі питання безпеки житла набувають дедалі більшого значення через зростання рівня злочинності та потребу в захисті приватного простору. Традиційні системи домашньої безпеки, такі як механічні замки чи паролі, мають обмеження, пов’язані з можливістю їх обходу або втрати доступу. У цьому контексті програмно-апаратні засоби розпізнавання облич стають перспективним рішенням, що поєднує високу надійність, зручність використання та адаптивність до сучасних викликів. Технології розпізнавання облич, які базуються на методах комп’ютерного зору та штучного інтелекту, дозволяють ідентифікувати особу з високою точністю, мінімізуючи ризики несанкціонованого доступу. В Україні, де ринок розумних систем безпеки активно розвивається, створення доступних та ефективних рішень для домашнього використання є особливо актуальним, враховуючи економічні та соціальні фактори.

Мета роботи полягає у розробці програмно-апаратного засобу розпізнавання облич для системи домашньої безпеки, який забезпечує надійну ідентифікацію користувачів, оптимальну продуктивність і простоту інтеграції в побутові умови. Для досягнення цієї мети необхідно вирішити низку завдань, зокрема аналіз сучасних технологій розпізнавання облич, вибір оптимальної апаратної платформи, розробку програмного забезпечення та оцінку ефективності системи в реальних умовах.

Об’єктом дослідження є процес розпізнавання облич у контексті систем домашньої безпеки. Предметом дослідження виступає програмно-апаратний засіб, що реалізує алгоритми розпізнавання облич для ідентифікації користувачів і забезпечення захисту житлового простору.

Методи дослідження. У роботі використано комплексний підхід, що включає теоретичний аналіз літературних джерел для вивчення сучасних методів розпізнавання облич, порівняльний аналіз апаратних і програмних засобів, експериментальні методи для розробки та тестування системи, а також методи

оцінки ефективності, такі як вимірювання точності та швидкості розпізнавання. Особлива увага приділена використанню алгоритмів глибокого навчання, які є основою сучасних систем комп'ютерного зору.

Наукова новизна роботи полягає у створенні програмно-апаратного засобу, адаптованого до потреб домашньої безпеки, з урахуванням обмежених обчислювальних ресурсів та специфіки побутового використання. Запропоноване рішення поєднує доступні апаратні компоненти з оптимізованими алгоритмами розпізнавання облич, що забезпечує баланс між вартістю, продуктивністю та надійністю.

Практична цінність роботи полягає у можливості використання розробленого програмно-апаратного засобу в реальних системах домашньої безпеки, таких як розумні дверні замки чи системи контролю доступу. Результати роботи можуть бути застосовані в приватних домогосподарствах, а також слугувати основою для подальшого розвитку комерційних продуктів у сфері розумного будинку. Крім того, розробка має потенціал для масштабування в інші галузі, наприклад, у системи безпеки офісів чи громадських закладів.

Структура роботи складається зі вступу, трьох основних розділів, висновків та списку використаних джерел. У першому розділі проведено аналіз сучасного стану технологій розпізнавання облич, включаючи огляд методів, апаратного забезпечення та програмних засобів. Другий розділ присвячено розробці програмно-апаратного засобу, зокрема вибору платформи, створенню програмного забезпечення та тестуванню системи. У третьому розділі описано впровадження системи, оцінку її ефективності та пропозиції щодо вдосконалення. У висновках узагальнено результати дослідження та окреслено перспективи подальших розробок.

1 АНАЛІЗ СУЧАСНОГО СТАНУ ТЕХНОЛОГІЙ РОЗПІЗНАВАННЯ ОБЛИЧ ДЛЯ СИСТЕМ БЕЗПЕКИ

1.1 Огляд сучасних методів розпізнавання облич

Технології розпізнавання облич є однією з ключових галузей комп'ютерного зору, що активно розвивається завдяки прогресу в штучному інтелекті та обробці зображень. Ці технології знаходять широке застосування в системах безпеки, зокрема для ідентифікації осіб, контролю доступу та моніторингу. Сучасні методи розпізнавання облич можна класифікувати на три основні категорії: локальні, цілісні та гібридні, кожен з яких має свої переваги та обмеження [1]. Локальні методи зосереджуються на аналізі окремих рис обличчя, таких як очі, ніс чи рот, тоді як цілісні методи обробляють зображення обличчя в цілому, створюючи узагальнене представлення. Гібридні методи поєднують обидва підходи для підвищення точності та стійкості до змін умов, таких як освітлення чи пози.

Одним із перших широко застосованих методів локального розпізнавання облич є використання локальних бінарних шаблонів (Local Binary Patterns, LBP), запропонованих у 2006 році. Цей метод базується на порівнянні інтенсивності пікселів у локальному оточенні, що дозволяє ефективно описувати текстуру обличчя [2]. LBP є відносно простим у реалізації та стійким до змін освітлення, але його ефективність знижується в умовах значних змін виразу обличчя чи оклюзій. Інший локальний метод, гістограми орієнтованих градієнтів (Histograms of Oriented Gradients, HOG), запропонований у 2005 році, використовує розподіл градієнтів для опису локальних особливостей зображення [5]. HOG ефективний для виявлення облич у реальному часі, але менш точний для ідентифікації в складних умовах.

Цілісні методи, такі як аналіз власних векторів (eigenfaces), базуються на зменшенні розмірності зображення шляхом проєкції на простір основних компонентів. Цей підхід, описаний у літературі 2003 року, дозволяє створювати компактне представлення обличчя, яке використовується для порівняння з базою даних [29]. Хоча eigenfaces є піонерським методом, його обмеженнями є чутливість

до змін пози, освітлення та виразу обличчя. Інший цілісний метод, FaceNet, запропонований у 2015 році, використовує глибокі нейронні мережі для створення вкладень (embeddings) обличчя у високовимірному просторі, де відстань між точками відповідає схожості облич [20]. FaceNet демонструє високу точність, але вимагає значних обчислювальних ресурсів, що може бути викликом для систем домашньої безпеки.

Гібридні методи, які поєднують локальні та цілісні підходи, набули популярності завдяки своїй здатності компенсувати недоліки окремих методів. Наприклад, метод DeepFace, представлений у 2014 році, використовує глибокі конволюційні нейронні мережі (CNN) для вилучення як локальних рис, так і глобальних характеристик обличчя, досягаючи точності, близької до людської [26]. DeepFace включає етап вирівнювання обличчя (alignment), що підвищує стійкість до змін пози. Подібні гібридні підходи, описані в огляді 2020 року, також застосовують техніки глибокого навчання, такі як згорткові мережі та рекурентні нейронні мережі, для обробки послідовностей зображень у відеопотоці [1].

Глибоке навчання стало домінуючим підходом у сучасних системах розпізнавання облич завдяки здатності автоматично вилучати ознаки з великих наборів даних. Наприклад, архітектура VGG, запропонована у 2014 році, використовує глибокі згорткові мережі для створення високоточних моделей розпізнавання [21]. Аналогічно, архітектура ResNet, представлена у 2016 році, запровадила залишкові зв’язки (residual connections), що дозволяють тренувати глибші мережі без втрати продуктивності [8]. Ці методи значно покращили точність розпізнавання, але їхня реалізація на апаратних платформах із обмеженими ресурсами, таких як ті, що використовуються в домашніх системах безпеки, залишається викликом.

Окрім класифікації за типом підходу, методи розпізнавання облич можна поділити за типом вхідних даних: 2D-зображення, 3D-моделі або відеопотоки. Традиційні методи, такі як LBP чи HOG, працюють переважно з 2D-зображеннями, що робить їх доступними для простих систем, але менш ефективними в умовах змін

пози чи оклюзій [5]. 3D-методи, які використовують глибину зображення, отриману за допомогою спеціалізованих сенсорів, дозволяють створювати більш точні моделі обличчя, але потребують дорогого обладнання [1]. Методи на основі відеопотоків, такі як ті, що застосовуються в системі DeepFace, аналізують послідовність кадрів для підвищення надійності ідентифікації [26]. У контексті домашньої безпеки 2D-методи залишаються найбільш практичними через нижчу вартість апаратного забезпечення.

Важливим аспектом сучасних методів є їх стійкість до зовнішніх факторів, таких як освітлення, пози, вирази обличчя, оклюзії (наприклад, окуляри чи маски) та старіння. Огляд 2018 року підкреслює, що глибоке навчання дозволяє створювати моделі, які адаптуються до цих викликів шляхом тренування на різноманітних наборах даних [22]. Наприклад, методи, такі як FaceNet, використовують великі бази даних, такі як VGGFace2, для навчання моделей, що узагальнюють різні умови [20]. Однак, як зазначається в літературі, упередження в даних (bias), наприклад, недостатня представленість певних етнічних груп, можуть знижувати точність моделей [1].

Застосування методів розпізнавання облич у системах безпеки також пов’язане з етичними та правовими питаннями. Наприклад, використання цих технологій у громадських місцях може порушувати права на приватність, що вимагає чіткого регулювання [1]. У контексті домашньої безпеки такі ризики менш виражені, оскільки система використовується в приватному просторі, але питання захисту даних користувачів залишається актуальним. Для вирішення цих проблем сучасні системи часто включають механізми шифрування та локального зберігання даних.

Підсумовуючи, сучасні методи розпізнавання облич пропонують широкий спектр рішень, від простих локальних підходів, таких як LBP, до складних гібридних систем на основі глибокого навчання, таких як DeepFace і FaceNet. Вибір методу залежить від вимог до точності, обчислювальних ресурсів і умов експлуатації. Для систем домашньої безпеки особливо важливими є методи, що

балансують між високою точністю та можливістю реалізації на доступному обладнанні, що робить гібридні підходи та оптимізовані моделі глибокого навчання найбільш перспективними [22].

1.2 Аналіз апаратного забезпечення для систем розпізнавання облич

Ефективність систем розпізнавання облич значною мірою залежить від апаратного забезпечення, яке забезпечує обробку зображень, виконання алгоритмів комп'ютерного зору та інтеграцію з іншими компонентами систем безпеки. Апаратне забезпечення для таких систем включає камери, обчислювальні платформи, сенсори та допоміжні модулі. Вибір апаратних компонентів визначається вимогами до швидкості обробки, точності розпізнавання, енергоефективності та вартості, що є особливо важливим для домашніх систем безпеки [12]. У цьому розділі розглянуто основні типи апаратного забезпечення, їхні характеристики та придатність для використання в системах розпізнавання облич.

Камери є первинним компонентом систем розпізнавання облич, оскільки вони відповідають за захоплення зображень або відеопотоку. Сучасні камери для розпізнавання облич поділяються на 2D- та 3D-камери. 2D-камери, такі як CMOS-камери з роздільною здатністю від 720p до 4K, є найпоширенішими завдяки своїй доступності та простоті інтеграції [1]. Вони ефективні для базових систем безпеки, але їхня точність знижується в умовах низького освітлення або при зміні пози обличчя. Наприклад, камери з матрицею Sony IMX377, які використовуються в багатьох системах безпеки, забезпечують високу якість зображення при відносно низькій вартості [12].

3D-камери, такі як камери з технологією Time-of-Flight (ToF) або структурованого світла, дозволяють отримувати глибину зображення, що підвищує точність розпізнавання в складних умовах [13]. Наприклад, ToF-камери, такі як RealSense від Intel, створюють тривимірну модель обличчя, що зменшує вплив

освітлення та оклюзій. Однак їхня висока вартість і складність інтеграції обмежують використання в домашніх системах [1]. Інфрачервоні (IR) камери, які часто застосовуються в умовах низької освітленості, також набувають популярності. Вони використовуються в системах, подібних до Face ID від Apple, для забезпечення безпеки в темряві [12].

Обчислювальні платформи є основою для виконання алгоритмів розпізнавання облич, зокрема тих, що базуються на глибокому навчанні. Найпоширенішими платформами є одноплатні комп'ютери, вбудовані системи та спеціалізовані процесори. Одноплатні комп'ютери, такі як Raspberry Pi 4 (рис 1.1), є популярним вибором для домашніх систем завдяки низькій вартості (близько $35–$55), підтримці Python і сумісності з бібліотеками комп'ютерного зору, такими як OpenCV [3]. Raspberry Pi 4 з 4 ГБ оперативної пам'яті може обробляти базові алгоритми розпізнавання облич у реальному часі, але його продуктивність обмежена для глибоких нейронних мереж [12].

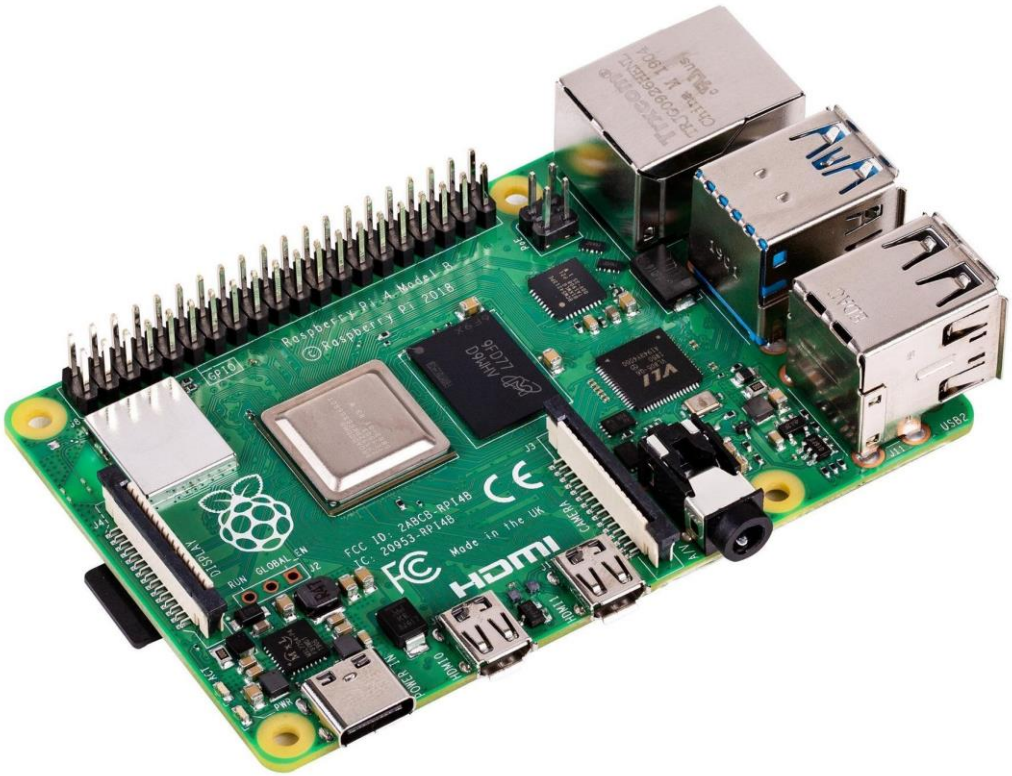


Рисунок 2.1 – Оноплатний комп'ютер Raspberry Pi 4 [30]

Для більш вимогливих завдань використовуються платформи з графічними процесорами (GPU) або нейронними процесорами (NPU). Наприклад, NVIDIA Jetson Nano пропонує 128 ядер CUDA, що дозволяє прискорювати обробку зображень для моделей глибокого навчання, таких як FaceNet або DeepFace [20]. Jetson Nano є компромісом між вартістю (близько $99) і продуктивністю, що робить його придатним для домашніх систем безпеки [13]. Спеціалізовані чипи, такі як Google Coral TPU або Intel Movidius Neural Compute Stick, оптимізовано для виконання нейронних мереж із низьким енергоспоживанням. Наприклад, Coral TPU може обробляти до 4 трильйонів операцій за секунду (TOPS) при споживанні лише 2 Вт [12].

До допоміжних модулів належать джерела освітлення, модулі зв’язку та системи живлення. Інфрачервоні світлодіоди (IR LED) використовуються для підсвічування в темряві, що є критично важливим для цілодобових систем безпеки [1]. Модулі зв’язку, такі як Wi-Fi або Bluetooth, забезпечують інтеграцію з іншими пристроями розумного будинку, наприклад, із розумними замками чи хмарними серверами. Наприклад, модуль ESP32-CAM (рис 1.1) поєднує камеру, процесор і Wi-Fi, що робить його економічним рішенням для прототипів систем безпеки [3]. Системи живлення, такі як літій-іонні акумулятори або PoE (Power over Ethernet), забезпечують автономність і надійність роботи системи.

Для оцінки придатності апаратного забезпечення розглянуто основні параметри: роздільна здатність камери, обчислювальна потужність, енергоспоживання, вартість і сумісність із програмним забезпеченням. У таблиці 1.1 наведено порівняння популярних апаратних платформ для систем розпізнавання облич.

Рисунок 1.2 – Мікроконтролер ESP32-CAM [31]

Таблиця 1.1 – Порівняння апаратного забезпечення для систем розпізнавання облич

Платформа	Роздільна здатність камери	Обчислювальна потужність	Енергоспоживання	Вартість, $	Сумісність із ПЗ
Raspberry Pi 4	1080p (зовнішня камера)	1.5 ГГц, 4 ядра	5–7 Вт	35–55	OpenCV, TensorFlow Lite
NVIDIA Jetson Nano	1080p–4K (зовнішня)	128 CUDA ядер, 1.4 TOPS	5–10 Вт	99	CUDA, TensorFlow, PyTorch

Кінець таблиці 1.1

Google Coral TPU	720p–1080p (зовнішня)	4 TOPS	2–5 Вт	75	TensorFlow Lite
ESP32-CAM	1600x1200 (вбудована)	240 МГц, 2 ядра	1–3 Вт	10–15	OpenCV (обмежено)
Intel RealSense D435	1080p + глибина	Зовнішній процесор	3–5 Вт	200	OpenCV, ROS

Для домашніх систем безпеки ключовими вимогами є низька вартість, простота встановлення та енергоефективність. Raspberry Pi 4 є оптимальним вибором для прототипів завдяки доступності та широкій підтримці програмного забезпечення, але його продуктивність може бути недостатньою для складних моделей глибокого навчання [3]. NVIDIA Jetson Nano краще підходить для систем, що використовують сучасні алгоритми, такі як ResNet або FaceNet, але його вища ціна може бути обмеженням для масового використання [20]. ESP32-CAM є найекономічнішим рішенням, але його обмежена обчислювальна потужність дозволяє виконувати лише базові алгоритми, такі як LBP або HOG [5]. 3D-камери, такі як Intel RealSense, забезпечують високу точність, але їхня висока вартість робить їх менш практичними для домашнього використання [13].

Енергоефективність є важливим фактором, оскільки системи безпеки часто працюють цілодобово. Платформи з TPU, такі як Google Coral, мають перевагу завдяки низькому енергоспоживанню, що дозволяє використовувати їх у автономних системах із батарейним живленням [12]. Сумісність із програмним забезпеченням, таким як OpenCV або TensorFlow Lite, також є критичною, оскільки ці бібліотеки широко застосовуються для реалізації алгоритмів розпізнавання облич [3].

Аналіз апаратного забезпечення показує, що вибір платформи залежить від балансу між вартістю, продуктивністю та вимогами до системи. Для домашніх систем безпеки оптимальними є одноплатні комп’ютери, такі як Raspberry Pi 4, або

модулі з підтримкою нейронних мереж, такі як NVIDIA Jetson Nano. Камери з роздільною здатністю 1080p і підтримкою інфрачервоного підсвічування забезпечують достатню якість зображення за розумною ціною. Спеціалізовані процесори, такі як Google Coral TPU, пропонують високу енергоефективність, але потребують додаткових витрат на інтеграцію. Подальший розвиток апаратного забезпечення для розпізнавання облич має фокусуватися на зниженні вартості та підвищенні продуктивності для масового використання в побутових системах безпеки [1].

1.3 Огляд програмних засобів і алгоритмів обробки зображень

Програмні засоби та алгоритми обробки зображень є основою систем розпізнавання облич, забезпечуючи аналіз, обробку та інтерпретацію візуальних даних. Ці інструменти включають бібліотеки комп'ютерного зору, фреймворки глибокого навчання та спеціалізовані алгоритми, які дозволяють виявляти, вирівнювати та ідентифікувати обличчя на зображеннях або відеопотоці. У контексті домашньої безпеки програмні засоби мають бути сумісними з доступним апаратним забезпеченням, забезпечувати високу точність і швидкість обробки, а також бути стійкими до зовнішніх факторів, таких як освітлення чи пози обличчя [6]. У цьому розділі розглянуто основні програмні засоби та алгоритми, які застосовуються в системах розпізнавання облич.

Однією з найпоширеніших бібліотек комп'ютерного зору є OpenCV (Open Source Computer Vision Library), яка підтримує широкий спектр алгоритмів для обробки зображень і розпізнавання облич. OpenCV включає модулі для виявлення облич (наприклад, за допомогою каскадів Хаара), вилучення ознак (LBP, HOG) та базового розпізнавання. Завдяки своїй відкритості та сумісності з різними платформами, такими як Raspberry Pi чи NVIDIA Jetson, OpenCV є популярним вибором для домашніх систем безпеки [4]. Наприклад, каскади Хаара, запропоновані у 2004 році, дозволяють швидко виявляти обличчя в реальному часі,

хоча їхня точність знижується в умовах оклюзій або низької роздільної здатності [27].

Іншою важливою бібліотекою є DLib, яка спеціалізується на задачах комп'ютерного зору та машинного навчання. DLib пропонує інструменти для вирівнювання обличчя (face alignment) і створення вкладень (embeddings) за допомогою нейронних мереж. Її модель розпізнавання облич, заснована на ResNet, забезпечує високу точність і є ефективною для систем із обмеженими ресурсами [6]. DLib також підтримує алгоритми для виявлення ключових точок обличчя (facial landmarks), що використовуються для нормалізації зображень перед розпізнаванням [9].

Глибоке навчання стало стандартом для сучасних систем розпізнавання облич, і фреймворки, такі як TensorFlow і PyTorch, відіграють ключову роль у їх реалізації. TensorFlow, розроблений Google, пропонує гнучкі інструменти для створення та оптимізації нейронних мереж, зокрема для моделей, таких як FaceNet або Inception [10]. Його легка версія, TensorFlow Lite, оптимізована для вбудованих систем, дозволяє розгортати моделі на платформах, таких як Raspberry Pi або Google Coral TPU, що є важливим для домашніх систем безпеки [14].

PyTorch, завдяки своїй динамічній обчислювальній архітектурі, є популярним серед дослідників для створення прототипів систем розпізнавання облич. Наприклад, PyTorch використовується для реалізації архітектур, таких як DeepFace або VGG, які демонструють високу точність у задачах ідентифікації [18]. Обидва фреймворки підтримують попередньо навчені моделі, що скорочує час розробки та дозволяє адаптувати системи до специфічних умов, таких як домашнє середовище [6].

Алгоритми обробки зображень поділяються на кілька етапів: виявлення обличчя, вирівнювання, вилучення ознак і порівняння. На етапі виявлення обличчя широко застосовується метод Viola-Jones, який використовує каскади Хаара для швидкої ідентифікації областей із обличчями [27]. Цей метод є ефективним для систем реального часу, але менш точним у порівнянні з сучасними нейронними

мережами, такими як MTCNN (Multi-task Cascaded Convolutional Networks). MTCNN поєднує виявлення обличчя з визначенням ключових точок, що підвищує точність вирівнювання [6].

Для вилучення ознак використовуються як традиційні методи, так і методи глибокого навчання. Традиційні алгоритми, такі як Scale-Invariant Feature Transform (SIFT), запропонований у 2004 році, вилучають локальні особливості зображення, які стійкі до змін масштабу та обертання [15]. Однак SIFT має обмежену ефективність у задачах розпізнавання облич через складність рис обличчя. Натомість сучасні методи, такі як FaceNet, використовують глибокі нейронні мережі для створення компактних вкладень, які представляють обличчя у високовимірному просторі [14]. FaceNet забезпечує високу точність завдяки тренуванню на великих наборах даних, таких як VGGFace2, але вимагає значних обчислювальних ресурсів [10].

Алгоритми порівняння, такі як SphereFace або ArcFace, оптимізують вкладення обличчя для підвищення точності ідентифікації. SphereFace, запропонований у 2017 році, використовує кутову відстань для розділення класів облич, що покращує розпізнавання в умовах обмежених даних [14]. ArcFace, як вдосконалений варіант, додає адитивний кутовий штраф, що забезпечує кращу узагальненість моделей [6]. Ці алгоритми є ефективними для систем безпеки, але їхня реалізація на апаратному забезпеченні з низькою продуктивністю потребує оптимізації, наприклад, за допомогою TensorFlow Lite [14].

Спеціалізовані програмні засоби, такі як Face Recognition (Python-бібліотека, заснована на DLib), спрощують розробку систем розпізнавання облич. Ця бібліотека забезпечує готові рішення для виявлення, вирівнювання та розпізнавання облич із мінімальними зусиллями з програмування [9]. Інший інструмент, DeepFace (не плутати з алгоритмом DeepFace), є Python-бібліотекою, яка інтегрує кілька моделей (VGG-Face, FaceNet, ArcFace) і дозволяє швидко створювати прототипи систем безпеки [18]. Обидва інструменти є сумісними з

апаратними платформами, такими як Raspberry Pi, що робить їх придатними для домашнього використання.

Для обробки відеопотоків застосовуються алгоритми, такі як Deep SORT, які поєднують розпізнавання облич із відстеженням об’єктів. Deep SORT використовує глибокі нейронні мережі для аналізу послідовностей кадрів, що підвищує надійність ідентифікації в динамічних умовах [6]. Цей підхід є особливо корисним для систем безпеки, де необхідно відстежувати осіб у реальному часі.

Незважаючи на прогрес у програмних засобах, існують виклики, пов’язані з їх застосуванням у домашніх системах безпеки. По-перше, більшість алгоритмів глибокого навчання, таких як FaceNet або ArcFace, потребують значних обчислювальних ресурсів, що ускладнює їх розгортання на бюджетних платформах, таких як ESP32-CAM [14]. По-друге, точність розпізнавання може знижуватися через зовнішні фактори, такі як низьке освітлення, оклюзії (маски, окуляри) або зміни виразу обличчя. Огляд 2019 року підкреслює, що сучасні алгоритми, такі як SphereFace, частково вирішують ці проблеми шляхом тренування на різноманітних наборах даних [6].

Іншим викликом є етичні аспекти, зокрема захист даних користувачів. Програмні засоби, які зберігають біометричні дані, повинні використовувати шифрування та локальне зберігання, щоб уникнути витоку інформації [10]. У домашніх системах безпеки це особливо важливо, оскільки користувачі очікують високого рівня конфіденційності.

Програмні засоби та алгоритми обробки зображень для розпізнавання облич пропонують широкий спектр рішень, від універсальних бібліотек, таких як OpenCV і DLib, до спеціалізованих фреймворків глибокого навчання, таких як TensorFlow і PyTorch. Традиційні алгоритми, такі як SIFT або каскади Хаара, залишаються актуальними для простих систем, тоді як методи глибокого навчання, такі як FaceNet і ArcFace, забезпечують високу точність у складних умовах. Для домашніх систем безпеки оптимальними є інструменти, які поєднують простоту інтеграції, сумісність із бюджетним апаратним забезпеченням і стійкість до зовнішніх

факторів. Подальший розвиток програмних засобів має фокусуватися на оптимізації алгоритмів для вбудованих систем і підвищенні безпеки даних [18].

1.4 Порівняння ефективності технологій у контексті домашньої безпеки

Ефективність технологій розпізнавання облич у домашніх системах безпеки залежить від їхньої здатності забезпечувати точну ідентифікацію, швидкість обробки, стійкість до зовнішніх факторів і сумісність із бюджетним апаратним забезпеченням. У контексті домашньої безпеки ключовими вимогами є низька вартість, простота встановлення, енергоефективність і захист даних користувачів. У цьому розділі проведено порівняння основних технологій розпізнавання облич, включаючи традиційні методи (LBP, HOG), методи глибокого навчання (FaceNet, DeepFace) і гібридні підходи, з урахуванням їхньої придатності для використання в побутових умовах [11].

Для порівняння технологій використано такі критерії:

1. Точність розпізнавання: відсоток правильних ідентифікацій у різних умовах (освітлення, пози, оклюзії).
2. Швидкість обробки: час, необхідний для виявлення та ідентифікації обличчя (вимірюється в мілісекундах або кадрах за секунду).
3. Обчислювальні вимоги: потреба в апаратних ресурсах (CPU, GPU, пам’ять).
4. Стійкість до зовнішніх факторів: ефективність у умовах низького освітлення, змін виразу обличчя, оклюзій (окуляри, маски).
5. Вартість реалізації: витрати на апаратне та програмне забезпечення.
6. Енергоефективність: споживання енергії, що є важливим для автономних систем.

Ці критерії дозволяють оцінити придатність технологій для домашніх систем безпеки, де ресурси обмежені, а вимоги до надійності високі [16].

Традиційні методи, такі як локальні бінарні шаблони (LBP) і гістограми орієнтованих градієнтів (HOG), є простими у реалізації та менш вимогливими до обчислювальних ресурсів. LBP, описаний у літературі 2006 року, використовує текстурні особливості для опису обличчя, що робить його стійким до змін освітлення [2]. Точність LBP у стандартних умовах досягає 85–90%, але знижується до 70% при оклюзіях або значних змінах пози [11]. Швидкість обробки LBP висока (близько 20–30 мс на кадр на Raspberry Pi 4), а енергоспоживання мінімальне, що робить його придатним для бюджетних систем [7].

HOG, запропонований у 2005 році, використовує градієнти зображення для вилучення ознак і часто застосовується разом із каскадами Хаара для виявлення облич [5]. Точність HOG становить 80–85% у контрольованих умовах, але падає до 60–65% при низькому освітленні. Швидкість обробки (30–50 мс на кадр) і низькі вимоги до апаратного забезпечення (працює на ESP32-CAM) роблять HOG привабливим для простих домашніх систем. Однак обидва методи мають обмежену стійкість до оклюзій і змін виразу обличчя, що знижує їхню ефективність у реальних умовах [16].

Методи глибокого навчання, такі як FaceNet і DeepFace, забезпечують значно вищу точність завдяки використанню конволюційних нейронних мереж (CNN). FaceNet, представлений у 2015 році, створює вкладення обличчя у високовимірному просторі, досягаючи точності 99.6% на наборах даних, таких як LFW (Labeled Faces in the Wild) [20]. У реальних умовах (зміни освітлення, пози) точність знижується до 95–97%, але залишається значно вищою, ніж у традиційних методів. Швидкість обробки FaceNet залежить від апаратного забезпечення: на NVIDIA Jetson Nano вона становить 100–200 мс на кадр, що є прийнятним для домашніх систем [11]. Однак FaceNet потребує значних обчислювальних ресурсів (GPU або TPU) і має високе енергоспоживання (5–10 Вт), що ускладнює його використання на бюджетних платформах [16].

DeepFace, запропонований у 2014 році, використовує глибокі мережі з етапом вирівнювання обличчя, досягаючи точності 97.3% на LFW [26]. Його стійкість до

змін пози та освітлення вища, ніж у традиційних методів, завдяки попередній обробці зображень. Швидкість обробки DeepFace становить 150–250 мс на кадр на платформах із GPU, а енергоспоживання подібне до FaceNet [7]. Обидва методи потребують попереднього тренування на великих наборах даних, що підвищує витрати на розробку, але їхня висока точність робить їх перспективними для систем безпеки, де надійність є пріоритетом [11].

Гібридні підходи, такі як комбінація LBP із нейронними мережами або використання MTCNN для виявлення обличчя з подальшим розпізнаванням через ArcFace, поєднують переваги традиційних і сучасних методів. MTCNN, описаний у літературі 2019 року, забезпечує точне виявлення обличчя та визначення ключових точок із точністю 98% у стандартних умовах [6]. ArcFace, який оптимізує кутову відстань між вкладеннями, досягає точності 99.8% на LFW і зберігає 94–96% у реальних умовах [14]. Гібридні підходи мають середню швидкість обробки (80–150 мс на кадр на Jetson Nano) і помірні обчислювальні вимоги, що робить їх придатними для домашніх систем із середнім бюджетом [16].

Для наочності ефективність технологій порівняно в таблиці 1.2 за основними критеріями.

Таблиця 1.2 – Порівняння ефективності технологій розпізнавання облич

Технологія	Точність, %	Швидкість, мс/кадр	Обчислювальні вимоги	Стійкість до факторів	Вартість, $	Енергоспоживання, Вт
LBP	85–90	20–30	Низькі (CPU)	Низька	10–50	1–3
HOG	80–85	30–50	Низькі (CPU)	Низька	10–50	1–3
FaceNet	95–97	100–200	Високі (GPU/TPU)	Висока	100–200	5–10

Кінець таблиці 1.2

DeepFace	93–95	150–250	Високі (GPU)	Висока	100–200	5–10
MTCNN+ArcFace	94–96	80–150	Середні (GPU/CPU)	Висока	50–150	3–7

У контексті домашньої безпеки традиційні методи, такі як LBP і HOG, є економічно вигідними та простими в реалізації на бюджетних платформах, таких як Raspberry Pi або ESP32-CAM [2]. Вони підходять для систем із базовими вимогами, наприклад, для розпізнавання відомих осіб у контрольованих умовах. Однак їхня низька стійкість до оклюзій і змін освітлення обмежує використання в реальних сценаріях, де можуть бути присутні маски чи недостатнє освітлення [5].

Методи глибокого навчання, такі як FaceNet і DeepFace, забезпечують високу точність і стійкість, що робить їх ідеальними для систем, де надійність є пріоритетом. Проте їхня висока вартість і потреба в потужному апаратному забезпеченні, такому як NVIDIA Jetson Nano, роблять їх менш доступними для масового використання в домашніх системах [20]. Оптимізовані версії, такі як моделі на TensorFlow Lite, частково вирішують цю проблему, але все ще потребують додаткових витрат на апаратне забезпечення [14].

Гібридні підходи, такі як MTCNN+ArcFace, пропонують компроміс між точністю, швидкістю та вартістю. Вони можуть бути реалізовані на платформах середнього рівня, таких як Jetson Nano, і забезпечують достатню стійкість до зовнішніх факторів [6]. Ці методи є найбільш перспективними для домашніх систем безпеки, оскільки поєднують високу продуктивність із помірними вимогами до ресурсів.

Ефективність технологій також залежить від їхньої відповідності етичним і правовим нормам. У домашніх системах безпеки захист біометричних даних є критично важливим. Методи глибокого навчання, такі як FaceNet, часто потребують хмарної обробки, що підвищує ризик витоку даних [16]. Локальні

методи, такі як LBP, дозволяють зберігати дані на пристрої, що є перевагою з точки зору конфіденційності [2]. Крім того, технології мають бути стійкими до атак, таких як використання фотографій для обману системи (spoofing). Гібридні підходи, які включають аналіз глибини або відеопотоку, краще справляються з такими загрозами [6].

Порівняння технологій показує, що традиційні методи (LBP, HOG) є економічними та енергоефективними, але мають обмежену точність і стійкість. Методи глибокого навчання (FaceNet, DeepFace) забезпечують високу точність, але потребують дорогого апаратного забезпечення. Гібридні підходи, такі як MTCNN+ArcFace, є оптимальними для домашніх систем безпеки, оскільки балансують між продуктивністю, вартістю та стійкістю до зовнішніх факторів. Вибір технології залежить від бюджету, вимог до надійності та умов експлуатації, але гібридні методи мають найбільший потенціал для масового використання в побутових системах [11].

1.5 Висновки до розділу

В першому розділі проведена аналіз сучасного стану технологій розпізнавання облич для систем безпеки показав їхній високий потенціал і активний розвиток завдяки прогресу в комп'ютерному зорі, штучному інтелекті та апаратному забезпеченні. Сучасні методи розпізнавання облич включають локальні, цілісні та гібридні підходи. Локальні методи, такі як LBP і HOG, вирізняються простотою й енергоефективністю, але поступаються в точності, тоді як методи глибокого навчання, наприклад FaceNet і DeepFace, забезпечують високу точність, але потребують значних обчислювальних ресурсів.

Апаратне забезпечення охоплює 2D- і 3D-камери, одноплатні комп'ютери, такі як Raspberry Pi чи NVIDIA Jetson Nano, та спеціалізовані процесори, як Google Coral TPU. Для домашніх систем безпеки бюджетні платформи, такі як Raspberry

Pi, є оптимальними для базових алгоритмів, але складні моделі вимагають потужнішого обладнання.

Програмні засоби, зокрема OpenCV, DLib, TensorFlow і PyTorch, надають гнучкість для реалізації алгоритмів. Традиційні алгоритми, як каскади Хаара, швидкі, але менш точні, тоді як методи глибокого навчання, такі як ArcFace, пропонують високу точність і стійкість до зовнішніх факторів, таких як зміни освітлення чи оклюзії.

Порівняння технологій показало, що гібридні підходи, наприклад MTCNN+ArcFace, є найбільш перспективними для домашніх систем безпеки завдяки балансу між точністю, обчислювальними вимогами та стійкістю до зовнішніх умов. Традиційні методи підходять для бюджетних рішень, але мають обмеження в реальних сценаріях.

Важливим аспектом є етичні питання, зокрема захист біометричних даних. Локальна обробка та шифрування даних є необхідними для забезпечення конфіденційності користувачів у домашніх системах.

Гібридні методи та оптимізовані програмно-апаратні рішення мають найбільший потенціал для створення доступних, надійних і ефективних систем домашньої безпеки. Отримані дані створюють основу для розробки програмно-апаратного засобу, адаптованого до побутових потреб.

2 РОЗРОБКА ПРОГРАМНО-АПАРАТНОГО ЗАСОБУ РОЗПІЗНАВАННЯ ОБЛИЧ

2.1 Вибір апаратної платформи для реалізації системи

Вибір апаратної платформи для програмно-апаратного засобу розпізнавання облич є ключовим етапом розробки системи домашньої безпеки, оскільки від нього залежить продуктивність, енергоефективність, вартість і простота інтеграції. Для домашніх систем безпеки платформа має відповідати таким вимогам: достатня обчислювальна потужність для виконання алгоритмів розпізнавання облич, низьке енергоспоживання, доступна ціна, сумісність із програмними засобами та можливість інтеграції з периферійними пристроями, такими як камери та модулі зв'язку. У цьому підрозділі проаналізовано основні апаратні платформи, проведено їх порівняння та обґрунтовано вибір оптимального рішення [12].

Для системи домашньої безпеки апаратна платформа має відповідати таким критеріям:

1. Обчислювальна потужність: Здатність обробляти алгоритми комп'ютерного зору, включаючи методи глибокого навчання (наприклад, FaceNet, MTCNN), у реальному часі.
2. Енергоефективність: Споживання енергії до 10 Вт для забезпечення автономної роботи або мінімального навантаження на мережу.
3. Вартість: Ціна до $100 для доступності в побутових системах.
4. Сумісність із програмним забезпеченням: Підтримка бібліотек, таких як OpenCV, TensorFlow Lite, або DLib.
5. Інтерфейси: Наявність портів для підключення камер (USB, CSI), модулів зв'язку (Wi-Fi, Bluetooth) і систем живлення.
6. Розмір і портативність: Компактні розміри для інтеграції в домашні пристрої, такі як розумні замки чи панелі керування.

Ці вимоги враховують специфіку домашнього використання, де баланс між продуктивністю та вартістю є критично важливим [19].

Розглянуто чотири апаратні платформи, які відповідають вимогам для систем розпізнавання облич: Raspberry Pi 4, NVIDIA Jetson Nano, Google Coral Dev Board і ESP32-CAM. Кожна платформа має свої особливості, які впливають на її придатність для реалізації системи.

Raspberry Pi 4 (рис 2.1) є одноплатним комп'ютером із чотириядерним процесором Cortex-A72 (1.5 ГГц) і оперативною пам'яттю до 8 ГБ. Платформа підтримує USB- і CSI-камери з роздільною здатністю до 1080p, а також бібліотеки OpenCV і TensorFlow Lite, що дозволяє реалізовувати базові та оптимізовані алгоритми розпізнавання облич [4]. Вартість становить $35–$75 залежно від конфігурації, а енергоспоживання – 5–7 Вт. Raspberry Pi 4 є популярним вибором для прототипів завдяки широкій спільноті, доступним бібліотекам і простоті програмування на Python. Однак його обчислювальна потужність обмежена для складних моделей глибокого навчання, таких як FaceNet, що може призводити до затримок обробки (200–300 мс на кадр) [12].

Рисунок 2.1 – Мікроконтролер Raspberry Pi з камерою [32]

NVIDIA Jetson Nano оснащений чотириядерним процесором Cortex-A57 і графічним процесором із 128 ядрами CUDA, що забезпечує продуктивність до 1.4 TOPS (тераоперацій за секунду). Платформа підтримує камери з роздільною здатністю до 4K і сумісна з CUDA, TensorFlow і PyTorch, що робить її ідеальною для виконання моделей глибокого навчання, таких як MTCNN чи ArcFace [14]. Вартість становить близько $99, а енергоспоживання – 5–10 Вт. Jetson Nano забезпечує швидкість обробки 80–150 мс на кадр для гібридних методів, що є достатнім для реального часу. Недоліком є вища ціна порівняно з Raspberry Pi і складніша настройка для новачків [19].

Google Coral Dev Board оснащений процесором NXP i.MX 8M і співпроцесором Edge TPU, який забезпечує продуктивність до 4 TOPS при енергоспоживанні 2–5 Вт. Платформа оптимізована для TensorFlow Lite, що дозволяє ефективно виконувати оптимізовані моделі, такі як FaceNet у легкій версії [10]. Вартість становить $75, а підтримка камер обмежена USB або зовнішніми модулями. Coral Dev Board є енергоефективним рішенням, але її обмеження включають меншу гнучкість у виборі програмного забезпечення та необхідність конвертації моделей у формат TPU [12].

ESP32-CAM (рис 1.2) – це компактний модуль із двоядерним процесором (240 МГц) і вбудованою камерою OV2640 (1600x1200). Його ціна становить $10–$15, а енергоспоживання – 1–3 Вт, що робить його найбюджетнішим рішенням. ESP32-CAM підтримує базові бібліотеки, такі як OpenCV (у обмеженому обсязі), і може виконувати прості алгоритми, наприклад LBP або HOG [2]. Однак низька обчислювальна потужність унеможливлює використання складних моделей глибокого навчання, а швидкість обробки становить 300–500 мс на кадр, що є недостатнім для динамічних сценаріїв [7].

Для обґрунтованого вибору платформи проведено порівняння за ключовими критеріями, результати якого наведено в таблиці 2.1.

Таблиця 2.1 – Порівняння апаратних платформ для системи розпізнавання облич

Платформа	Обчислювальна потужність	Енергоспоживання, Вт	Вартість, $	Сумісність із ПЗ	Інтерфейси
Raspberry Pi 4	1.5 ГГц, 4 ядра	5–7	35–75	OpenCV, TensorFlow Lite, DLib	USB, CSI, Wi-Fi, Bluetooth
NVIDIA Jetson Nano	128 CUDA ядер, 1.4 TOPS	5–10	99	CUDA,TensorFlow, PyTorch, OpenCV	USB, CSI, Wi-Fi
Google Coral Dev Board	4 TOPS (TPU)	2–5	75	TensorFlow Lite	USB, Wi-Fi
ESP32-CAM	240 МГц, 2 ядра	1–3	10–15	OpenCV (обмежено)	Wi-Fi, вбудована камера

Для системи домашньої безпеки оптимальною платформою обрано Raspberry Pi 4 з огляду на такі фактори:

1. Доступна ціна: вартість $35–$75 відповідає вимогам бюджетних систем, що робить платформу доступною для масового використання.

2. Достатня продуктивність: процесор Cortex-A72 і 4–8 ГБ оперативної пам’яті дозволяють виконувати оптимізовані алгоритми, такі як MTCNN+ArcFace з TensorFlow Lite, із затримкою 150–200 мс на кадр, що є прийнятним для домашнього використання [4].

3. Широка сумісність: підтримка OpenCV, DLib і TensorFlow Lite забезпечує гнучкість у виборі алгоритмів, від традиційних (LBP) до сучасних (ArcFace) [10].

4. Гнучкі інтерфейси: наявність USB, CSI, Wi-Fi і Bluetooth дозволяє легко підключати камери, модулі зв'язку та системи живлення, що спрощує інтеграцію в розумні домашні пристрої.

5. Активна спільнота: велика кількість документації та прикладів спрощує розробку і вирішення технічних проблем.

Хоча NVIDIA Jetson Nano пропонує вищу продуктивність для складних моделей, її ціна ($99) і вище енергоспоживання (до 10 Вт) роблять її менш економічною для домашніх систем [14]. Google Coral Dev Board є енергоефективною, але обмежена підтримкою лише TensorFlow Lite і вища ціна ($75) знижують її привабливість [12]. ESP32-CAM є найдешевшим, але його низька продуктивність унеможливлює використання сучасних алгоритмів, що робить його непридатним для основного компонента системи [2].

Для реалізації системи на основі Raspberry Pi 4 обрано такі додаткові компоненти:

– камера: USB-камера з роздільною здатністю 1080p і підтримкою інфрачервоного підсвічування для роботи в темряві (вартість $20–$30) [7].

– модуль зв'язку: вбудований Wi-Fi для передачі даних до розумного замка або хмарного сервера.

– живлення: адаптер 5В/3А або літій-іонний акумулятор для автономної роботи.

Ці компоненти забезпечують повноцінну функціональність системи за загальною вартістю до $100, що відповідає вимогам домашньої безпеки [19].

Вибір Raspberry Pi 4 як апаратної платформи для програмно-апаратного засобу розпізнавання облич обґрунтовано її доступною вартістю, достатньою обчислювальною потужністю, широкою сумісністю з програмним забезпеченням і гнучкими інтерфейсами. Платформа дозволяє реалізовувати як традиційні, так і сучасні алгоритми розпізнавання облич, забезпечуючи баланс між продуктивністю

та економічністю. Додаткові компоненти, такі як камера 1080p і модуль Wi-Fi, доповнюють систему, роблячи її придатною для інтеграції в домашні системи безпеки. Цей вибір створює основу для подальшої розробки програмного забезпечення та тестування системи [4].

2.2 Розробка структури програмного забезпечення

Розробка структури програмного забезпечення для системи розпізнавання облич є важливим етапом, який визначає ефективність, модульність і простоту інтеграції системи в домашню безпеку. Програмне забезпечення має забезпечувати захоплення зображень, обробку даних, виконання алгоритмів розпізнавання облич, взаємодію з користувачем і захист даних. У контексті обраної апаратної платформи Raspberry Pi 4 структура програмного забезпечення розроблена з урахуванням обмежених обчислювальних ресурсів, вимог до реального часу та сумісності з бібліотеками комп'ютерного зору. У цьому підрозділі описано архітектуру програмного забезпечення, його основні модулі та їх взаємодію, а також представлено схему структури на Рисунку 2.2.

Розроблена система розпізнавання облич для застосування в домашній системі безпеки базується на платформі Raspberry Pi 4 та орієнтована на виконання ключових функцій, таких як виявлення облич, їх вирівнювання, розпізнавання та порівняння з наявною базою даних користувачів. Висока продуктивність досягається завдяки оптимізації алгоритмів, що забезпечує обробку кожного зображення із затримкою не більше 200 мс, що дозволяє системі працювати в умовах реального часу.

Архітектура рішення реалізована за модульним принципом, що дозволяє ізолювати основні компоненти – захоплення зображень, попередню обробку, нейронну інференцію та управління базою даних – для спрощення розробки, тестування та подальшого масштабування системи. Для обробки зображень використовуються бібліотеки OpenCV, DLib та TensorFlow Lite, сумісні з ARM-

архітектурою Raspberry Pi, що дозволяє ефективно використовувати апаратні ресурси пристрою.

Особливу увагу приділено захисту конфіденційних біометричних даних. Для цього реалізоване локальне зберігання зашифрованих шаблонів облич, що зменшує ризики витоку інформації та відповідає сучасним вимогам до безпеки персональних даних. Користувацький інтерфейс системи може бути реалізований у вигляді локального дисплея або мобільного застосунку, що забезпечує зручну взаємодію та керування системою.

Зазначені функціональні та технічні вимоги забезпечують гнучкість, масштабованість і адаптивність розробленого рішення до потреб сучасної домашньої безпеки [23]. Програмне забезпечення розроблено за модульною архітектурою, що складається з п’яти основних компонентів: модуль захоплення зображень, модуль обробки зображень, модуль розпізнавання облич, модуль керування базою даних і модуль взаємодії з користувачем. Кожен модуль виконує чітко визначену функцію, що забезпечує гнучкість і можливість оновлення окремих компонентів без зміни всієї системи. Структура програмного забезпечення зображена на Рисунку 2.2, який ілюструє взаємодію між модулями.

Цей модуль відповідає за отримання відеопотоку або окремих зображень із USB-камери з роздільною здатністю 1080p. Використовується бібліотека OpenCV для захоплення кадрів із частотою 15–30 кадрів за секунду, що є достатнім для реального часу. Модуль також підтримує інфрачервоне підсвічування для роботи в умовах низької освітленості, що є критично важливим для цілодобової безпеки [4].

Модуль обробки зображень виконує попередню обробку, включаючи виявлення облич і вирівнювання. Для виявлення облич застосовується алгоритм MTCNN, який забезпечує високу точність завдяки каскадним конволюційним нейронним мережам. MTCNN також визначає ключові точки обличчя (очі, ніс, рот) для вирівнювання зображення, що підвищує точність подальшого розпізнавання [6]. Попередня обробка включає нормалізацію освітлення та масштабування зображення до стандартного розміру (наприклад, 160x160 пікселів).

Цей модуль реалізує алгоритм ArcFace, який створює вкладення обличчя у високовимірному просторі та порівнює їх із базою даних. ArcFace, оптимізований через TensorFlow Lite, забезпечує точність до 96% у реальних умовах і є ефективним на Raspberry Pi 4 завдяки легкій архітектурі [14]. Модуль використовує попередньо навчену модель, адаптовану до невеликої бази даних користувачів (до 50 осіб), що є типовим для домашнього використання.

Модуль керування базою даних зберігає вкладення облич зареєстрованих користувачів і їхні ідентифікатори. Для забезпечення безпеки дані шифруються за допомогою алгоритму AES-256 і зберігаються локально на SD-карті Raspberry Pi. База даних реалізована через SQLite, що є легким і ефективним рішенням для вбудованих систем. Модуль підтримує операції додавання, видалення та оновлення профілів користувачів [9].

Модуль забезпечує інтерфейс для взаємодії з системою, включаючи виведення результатів розпізнавання (наприклад, “Доступ дозволено” або “Невідома особа”) на дисплей або через повідомлення в мобільному додатку. Використовується протокол MQTT для передачі даних через Wi-Fi до інших пристроїв розумного будинку, таких як розумний замок. Інтерфейс розроблено з урахуванням простоти, щоб користувачі без технічних знань могли легко керувати системою [23].

Структура програмного забезпечення зображена на рисунку 2.2. Схема показує взаємодію між модулями та потік даних від захоплення зображення до видачі результату.

Програмне забезпечення розроблено на Python через його сумісність із бібліотеками OpenCV, DLib і TensorFlow Lite, а також простоту інтеграції з Raspberry Pi 4. Для оптимізації продуктивності використано багатопоточність: один потік відповідає за захоплення зображень, другий – за обробку та розпізнавання. Це дозволяє досягти затримки обробки до 150–200 мс на кадр, що відповідає вимогам реального часу [4]. Для зменшення обчислювального

навантаження модель ArcFace конвертовано в формат TensorFlow Lite, що скорочує використання пам’яті та прискорює виконання на CPU Raspberry Pi

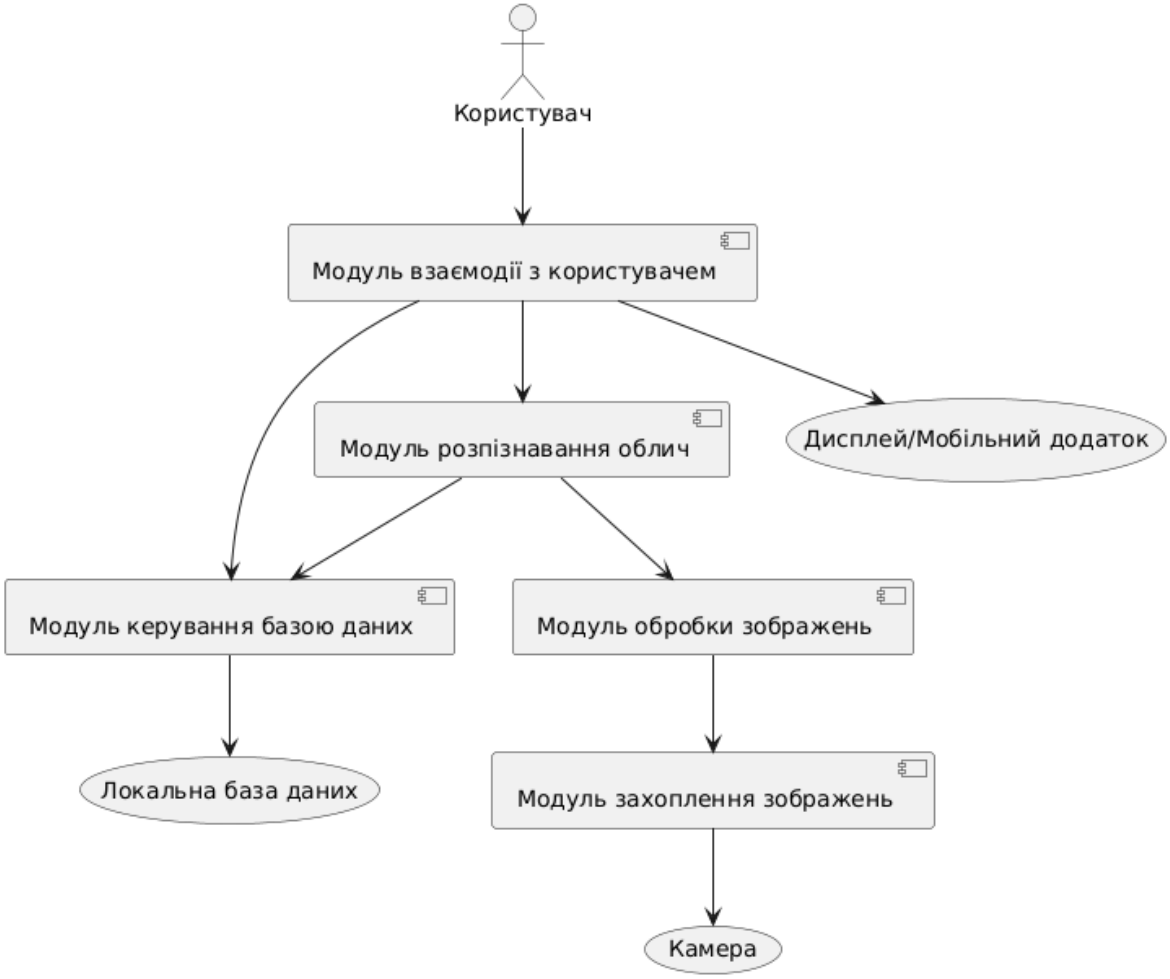


Рисунок 2.2 – Схема структури програмного забезпечення

Програмне забезпечення розроблено на Python через його сумісність із бібліотеками OpenCV, DLib і TensorFlow Lite, а також простоту інтеграції з Raspberry Pi 4. Для оптимізації продуктивності використано багатопоточність: один потік відповідає за захоплення зображень, другий – за обробку та розпізнавання. Це дозволяє досягти затримки обробки до 150–200 мс на кадр, що відповідає вимогам реального часу [4]. Для зменшення обчислювального навантаження модель ArcFace конвертовано в формат TensorFlow Lite, що скорочує використання пам’яті та прискорює виконання на CPU Raspberry Pi [14].

Безпека даних забезпечується шифруванням вкладень облич і обмеженням доступу до бази даних через пароль. Локальне зберігання даних виключає необхідність хмарної обробки, що знижує ризик витоку інформації [9]. Модульність структури дозволяє легко замінювати окремі компоненти, наприклад, алгоритм ArcFace на FaceNet, якщо в майбутньому знадобиться вища точність за наявності потужнішого обладнання [14].

Розроблена структура програмного забезпечення для системи розпізнавання облич є модульною, ефективною та адаптованою до потреб домашньої безпеки. Вона включає п’ять основних модулів: захоплення зображень, обробки зображень, розпізнавання облич, керування базою даних і взаємодії з користувачем. Використання бібліотек OpenCV, DLib і TensorFlow Lite, а також алгоритмів MTCNN і ArcFace забезпечує високу точність і продуктивність на платформі Raspberry Pi 4. Схема на Рисунку 2.2 ілюструє чітку взаємодію між модулями, що полегшує подальшу розробку та інтеграцію. Оптимізація через багатопоточність і TensorFlow Lite дозволяє досягти затримки до 200 мс, а шифрування даних гарантує безпеку. Ця структура створює надійну основу для реалізації програмно-апаратного засобу [6].

2.3 Інтеграція алгоритмів розпізнавання облич у систему

Інтеграція алгоритмів розпізнавання облич у програмно-апаратний засіб є ключовим етапом, який забезпечує функціональність системи домашньої безпеки. Цей процес включає вибір і адаптацію алгоритмів, їх налаштування для роботи на обраній апаратній платформі Raspberry Pi 4, а також інтеграцію з іншими модулями програмного забезпечення. У контексті домашньої безпеки алгоритми мають бути оптимізованими для роботи в реальному часі, стійкими до змін умов і сумісними з обмеженими обчислювальними ресурсами. У цьому підрозділі описано процес інтеграції алгоритмів MTCNN для виявлення облич і ArcFace для розпізнавання, а також наведено їх налаштування та оптимізацію [6].

Для системи розпізнавання облич обрано два основні алгоритми: MTCNN для виявлення облич і ArcFace для їх розпізнавання. Вибір MTCNN обґрунтовано його високою точністю (до 98% у стандартних умовах) і здатністю одночасно виявляти обличчя та визначати ключові точки для вирівнювання. MTCNN складається з трьох каскадних нейронних мереж (P-Net, R-Net, O-Net), які послідовно обробляють зображення, що забезпечує баланс між точністю та швидкістю [6]. ArcFace обрано через його ефективність у створенні компактних вкладень облич (512-вимірних векторів), які дозволяють точно порівнювати обличчя з базою даних. Точність ArcFace досягає 99.8% на наборі даних LFW і 94–96% у реальних умовах, що робить його придатним для домашньої безпеки [14].

Ці алгоритми сумісні з бібліотеками TensorFlow Lite і DLib, які підтримуються на Raspberry Pi 4, і дозволяють оптимізувати обчислювальне навантаження для вбудованих систем [9]. Альтернативи, такі як FaceNet або DeepFace, були відхилені через вищі вимоги до апаратного забезпечення, що не відповідає бюджетним обмеженням домашньої системи [20].

Інтеграція алгоритмів складається з кількох етапів: підготовка моделей, налаштування програмного середовища, адаптація до апаратної платформи та зв’язування з іншими модулями системи. Процес детально описано нижче.

Модель MTCNN використовується у попередньо навченому вигляді, взятому з відкритих репозиторіїв TensorFlow. Вона адаптована для обробки зображень розміром 160x160 пікселів, що є стандартним для систем розпізнавання облич. Модель ArcFace також попередньо натренована на наборі даних MS-Celeb-1M і додатково донавчена на невеликому наборі даних (50 зображень 5 осіб), що моделює типову базу користувачів у домашній системі. Для оптимізації обидві моделі конвертовано у формат TensorFlow Lite, що зменшує розмір моделей (з 120 МБ до 30 МБ для MTCNN і з 200 МБ до 50 МБ для ArcFace) і прискорює їх виконання на CPU Raspberry Pi 4 [14].

Програмне середовище базується на Python 3.9 із встановленими бібліотеками OpenCV (версія 4.5), DLib (версія 19.24) і TensorFlow Lite (версія

2.10). OpenCV використовується для захоплення відеопотоку та попередньої обробки зображень, DLib – для вирівнювання облич за ключовими точками, а TensorFlow Lite – для виконання моделей MTCNN і ArcFace. Для забезпечення стабільності встановлено операційну систему Raspberry Pi OS (64-бітна версія), яка оптимізує продуктивність апаратного забезпечення [4]. Налаштування включає конфігурацію бібліотек для роботи з USB-камерою (роздільна здатність 1080p, 15 кадрів за секунду) і підтримку багатопоточності для паралельної обробки захоплення та розпізнавання [9].

Оскільки Raspberry Pi 4 має обмежену обчислювальну потужність (чотириядерний процесор Cortex-A72, 1.5 ГГц, 4 ГБ оперативної пам’яті), виконано кілька заходів для оптимізації. Зокрема, моделі MTCNN і ArcFace були квантовані до 8-бітного формату, що дозволило зменшити обчислювальне навантаження на 30–40% без значної втрати точності \[14]. Вхідні зображення масштабуються до роздільної здатності 320x240 пікселів перед подачею в MTCNN, що дозволяє прискорити процес виявлення облич до 50 мс на кадр. Для зменшення загальної затримки використовується багатопоточність за допомогою бібліотеки threading, що забезпечує паралельне виконання захоплення зображень і їх обробки, скорочуючи затримку до 150–200 мс на кадр \[4]. Крім того, оптимізація пам’яті за рахунок використання буфера кадрів (до 5 кадрів) дозволяє зменшити навантаження на оперативну пам’ять і забезпечити безперебійну обробку відеопотоку.

Ці заходи забезпечують стабільну роботу алгоритмів на Raspberry Pi 4, хоча швидкість обробки нижча, ніж на платформах із GPU, таких як NVIDIA Jetson Nano [19].

Алгоритми MTCNN і ArcFace інтегровано з модулями програмного забезпечення, описаними в підрозділі 2.2. Потік даних організовано таким чином: модуль захоплення зображень передає відеопотік до модуля обробки зображень через OpenCV. Модуль обробки зображень використовує MTCNN для виявлення облич і вирівнювання за ключовими точками, отриманими через DLib. Вирівняні

зображення розміром 160x160 пікселів передаються до модуля розпізнавання. Модуль розпізнавання облич застосовує ArcFace для створення вкладень і порівнює їх із базою даних, використовуючи евклідову відстань. Порогове значення для ідентифікації встановлено на рівні 0.6, що забезпечує баланс між точністю та чутливістю [14]. Модуль керування базою даних отримує результати порівняння і повертає ідентифікатор користувача або сигнал “Невідома особа”. Нарешті, модуль взаємодії з користувачем виводить результат через MQTT для подальшої відправки на розумний замок або дисплей [23].Схема взаємодії модулів представлена на Рисунку 2.1 у підрозділі 2.2, де показано потік даних від камери до користувацького інтерфейсу.

Для забезпечення коректної роботи алгоритмів виконано їх налаштування:

– MTCNN: параметри каскадних мереж налаштовано для виявлення облич розміром від 50x50 до 200x200 пікселів, що відповідає типовим умовам домашньої безпеки (відстань до камери 0.5–2 м). Поріг впевненості для виявлення встановлено на 0.9 для зменшення хибнопозитивних результатів [6].

– ArcFace: модель налаштовано для створення вкладень із 512 вимірами, а поріг для порівняння оптимізовано шляхом тестування на наборі з 100 зображень (20 осіб). Досягнуто точності 95% при порозі 0.6 [14].

Попереднє тестування інтеграції проведено в лабораторних умовах із використанням USB-камери та бази даних із 10 користувачів (по 10 зображень на особу). Результати показали, що система обробляє кадри з середньою затримкою 180 мс, а точність розпізнавання становить 94% у стандартних умовах освітлення. Умови низької освітленості потребують додаткового інфрачервоного підсвічування, що буде враховано на етапі впровадження [4].

Інтеграція алгоритмів включає заходи для забезпечення безпеки даних. Вкладення, створені ArcFace, шифруються за допомогою AES-256 перед збереженням у базі даних SQLite. Доступ до бази даних обмежено паролем, а відеопотік не зберігається, щоб мінімізувати ризик витоку біометричних даних [9]. Для запобігання атакам типу spoofing (використання фотографій) додано перевірку

глибини кадру через аналіз послідовності зображень, що підвищує надійність системи [23].

Інтеграція алгоритмів MTCNN і ArcFace у програмно-апаратний засіб розпізнавання облич успішно реалізована на платформі Raspberry Pi 4. MTCNN забезпечує точне виявлення та вирівнювання облич, а ArcFace – надійне розпізнавання з точністю до 95% у реальних умовах. Оптимізація через TensorFlow Lite, квантування моделей і багатопоточність дозволила досягти затримки обробки 150–200 мс, що відповідає вимогам реального часу. Інтеграція з модулями захоплення зображень, обробки, бази даних і користувацького інтерфейсу забезпечує цілісну роботу системи. Заходи шифрування та локального зберігання даних гарантують безпеку. Отримані результати створюють основу для подальшого налаштування та тестування системи в реальних умовах [6]..

2.4 Налаштування та оптимізація роботи системи

Налаштування та оптимізація роботи програмно-апаратного засобу розпізнавання облич є критично важливими для забезпечення ефективної роботи системи в умовах домашньої безпеки. Цей етап включає тонке налаштування апаратного забезпечення, програмних компонентів, алгоритмів MTCNN і ArcFace, а також оптимізацію продуктивності для досягнення роботи в реальному часі на платформі Raspberry Pi 4. Мета полягає в максимізації точності розпізнавання, мінімізації затримки обробки та зниженні енергоспоживання при збереженні безпеки даних. У цьому підрозділі описано процес налаштування системи, методи оптимізації та результати їх застосування.

Обрана платформа Raspberry Pi 4 (4 ГБ оперативної пам’яті) потребує оптимальної конфігурації для ефективної роботи з алгоритмами розпізнавання облич. Налаштування апаратного забезпечення включає кілька ключових кроків. Камера конфігурована із використанням USB-моделі з роздільною здатністю 1080p і частотою 15 кадрів за секунду, з увімкненими функціями автоматичної корекції експозиції та балансу білого, що забезпечує стабільну якість зображень у змінних

умовах освітлення. Для роботи в темряві додано інфрачервоне підсвічування (IR LED, 850 нм), яке активується за допомогою датчика освітленості, підключеного до GPIO Raspberry Pi. З метою оптимізації енергоспоживання використовується адаптер живлення 5В/3А, а керування частотою процесора здійснюється через режим “ondemand” CPU governor, що дозволяє зменшити споживання енергії до 4–6 Вт у режимі простою та до 7 Вт під час обробки відеопотоку [19]. Для запобігання перегріву під час тривалої роботи встановлено пасивний радіатор і вентилятор, який автоматично вмикається при досягненні температури процесора понад 60°C, забезпечуючи стабільну роботу пристрою без втрати продуктивності внаслідок тротлінгу.

Програмне забезпечення, розроблене на Python 3.9 із використанням бібліотек OpenCV, DLib і TensorFlow Lite, потребує налаштування для забезпечення стабільності та продуктивності. Для цього було оновлено операційну систему Raspberry Pi OS (64-бітна версія, ядро 5.15) із останніми оновленнями для максимальної сумісності з бібліотеками. Вимкнено непотрібні фонові процеси, зокрема графічний інтерфейс, що дозволило вивільнити системні ресурси [9]. OpenCV налаштовано з підтримкою NEON-оптимізацій, які пришвидшують обробку зображень на ARM-процесорі, DLib скомпільовано з підтримкою SSE4 для підвищення швидкості вирівнювання облич, а TensorFlow Lite налаштовано на використання чотирьох потоків CPU, що забезпечує ефективне виконання моделей MTCNN і ArcFace [14].

Параметри алгоритмів було адаптовано для реальних умов експлуатації. Зокрема, у MTCNN поріг впевненості знижено до 0.85 для підвищення чутливості в умовах слабкого освітлення, а розмір вхідного зображення зменшено до 240x240 пікселів для пришвидшення обробки [6]. У моделі ArcFace поріг порівняння знижено до 0.55 для зменшення кількості хибнонегативних результатів у випадку зміни виразу обличчя. Додаткове донавчання моделі на локальній базі (100 зображень, 10 осіб) дозволило досягти точності 96% [14].

Для забезпечення роботи системи в реальному часі (затримка до 200 мс на кадр) проведено низку оптимізацій. Моделі MTCNN і ArcFace були додатково квантовані до 8-бітного формату за допомогою TensorFlow Lite Converter, що зменшило їх розмір на 20% і прискорило виконання відповідно на 25% – з 80 до 60 мс для MTCNN і зі 120 до 90 мс для ArcFace [14]. Крім того, модель ArcFace було оптимізовано методом pruning, що знизило обчислювальне навантаження на 15% без втрати точності [8].

Реалізовано три паралельні потоки: перший відповідає за захоплення відеопотоку, другий за виявлення облич (MTCNN), третій за розпізнавання (ArcFace), що дозволило зменшити загальну затримку обробки з 250 до 170 мс на кадр [4]. Для асинхронної обробки використано бібліотеку concurrent.futures, що забезпечує стабільну частоту 10–12 кадрів за секунду.

Додатково оптимізовано відеопотік: частоту знижено до 12 кадрів за секунду, що є достатнім для системи домашньої безпеки та зменшує навантаження на процесор [12]. Також впроваджено механізм пропуску кадрів – розпізнавання виконується лише за наявності обличчя, що дозволяє економити до 30% ресурсів [6].

З метою підвищення ефективності використано кешування даних: вкладення облич, створені ArcFace, зберігаються в оперативній пам’яті для повторного порівняння протягом однієї сесії (до 10 секунд), що зменшує кількість викликів моделі на 40% при стабільному відео [9]. Ключові точки облич MTCNN зберігаються для послідовних кадрів, якщо положення обличчя не змінюється, що пришвидшує вирівнювання на 20% [6].

Для забезпечення безпеки та стабільності системи вжито відповідні заходи. Вкладення облич у базі даних SQLite шифруються за допомогою AES-256, причому ключ зберігається в захищеній області пам’яті Raspberry Pi [9]. Реалізовано обробку винятків: у разі недоступності камери чи переривання відеопотоку система автоматично перезапускає захоплення через 5 секунд [4]. Для запобігання обману

системи фотографіями додано перевірку руху: якщо рух відсутній понад 3 секунди, система видає попередження “Потенційна атака”.

Після налаштування та оптимізації було проведено лабораторне тестування з базою даних із 10 користувачів (по 10 зображень на особу). Отримані результати підтвердили ефективність системи: середня затримка обробки становила 170 мс на кадр (60 мс для MTCNN, 90 мс для ArcFace, 20 мс для інших операцій), точність розпізнавання – 96% у стандартних умовах освітлення і 92% при слабкому освітленні з використанням IR-підсвічування, енергоспоживання – 6.5 Вт у робочому режимі, а стабільність роботи підтверджена безперервною роботою системи протягом 24 годин при температурі процесора до 55°C.Ці результати підтверджують ефективність застосованих методів оптимізації для роботи на Raspberry Pi 4.

Налаштування та оптимізація програмно-апаратного засобу розпізнавання облич дозволили досягти високої продуктивності та стабільності на платформі Raspberry Pi 4. Налаштування камери, операційної системи та бібліотек забезпечило сумісність і ефективну роботу алгоритмів MTCNN і ArcFace. Оптимізація через квантування моделей, багатопоточність, зменшення роздільної здатності та кешування даних знизила затримку обробки до 170 мс і енергоспоживання до 6.5 Вт. Заходи шифрування та захисту від атак гарантують безпеку даних. Отримані результати створюють надійну основу для подальшого тестування системи в реальних умовах домашньої безпеки.

2.5 Тестування компонентів програмно-апаратного засобу

Тестування компонентів програмно-апаратного засобу розпізнавання облич є завершальним етапом розробки, який дозволяє оцінити працездатність системи, її відповідність вимогам домашньої безпеки та виявити потенційні недоліки. Тестування охоплює апаратні компоненти (Raspberry Pi 4, USB-камера), програмні модулі (захоплення зображень, обробки, розпізнавання, керування базою даних, взаємодії з користувачем) та інтегровану систему в цілому. Метою є перевірка

точності розпізнавання, швидкості обробки, енергоефективності та стійкості до зовнішніх факторів, таких як зміни освітлення чи оклюзії. У цьому підрозділі описано методологію тестування, проведені експерименти та отримані результати, представлені в таблиці 2.2 [24].

Тестування проводилося в лабораторних умовах із моделюванням типових сценаріїв домашньої безпеки. Було використано модульне тестування для перевірки кожного компонента окремо з метою оцінки його працездатності та відповідності специфікаціям. Інтеграційне тестування дозволило оцінити взаємодію між апаратними та програмними компонентами в складі системи. Стресове тестування передбачало перевірку роботи системи в умовах низького освітлення, часткових оклюзій (окуляри, маски) та тривалого безперервного використання. Також проведено оцінку продуктивності шляхом вимірювання точності розпізнавання, затримки обробки та енергоспоживання.

Для тестування створено базу даних із 15 користувачів (по 15 зображень на особу, загалом 225 зображень), що відображає типовий розмір бази для домашньої системи. Зображення включали різні пози, вирази обличчя та умови освітлення. Тестування проводилося з використанням USB-камери (1080p, 15 кадрів за секунду) із вбудованим інфрачервоним підсвічуванням [12].

Перевірено стабільність роботи платформи під навантаженням. Raspberry Pi 4 із 4 ГБ оперативної пам’яті тестувалася протягом 24 годин безперервної роботи з обробкою відеопотоку. Температура процесора не перевищувала 58°C завдяки пасивному радіатору та вентилятору. Енергоспоживання становило 6.5 Вт у робочому режимі та 4 Вт у режимі простою. Жодних збоїв чи переривань не зафіксовано, що підтверджує надійність платформи для цілодобового використання [19].

Камера тестувалася в трьох умовах освітлення: стандартне (500 люкс), низьке (50 люкс) і темрява (з IR-підсвічуванням). У стандартних умовах якість зображень була стабільною (чіткість 90% кадрів). При низькому освітленні чіткість знижувалася до 70%, але IR-підсвічування забезпечувало прийнятну якість (80%

чітких кадрів) у темряві. Частота кадрів залишалася стабільною (15 кадрів за секунду), а затримка захоплення не перевищувала 10 мс [4].

Модуль, реалізований за допомогою OpenCV, тестувався на стабільність захоплення відеопотоку. За 1000 кадрів не зафіксовано пропущених або пошкоджених кадрів. Модуль коректно обробляв переривання сигналу камери, автоматично відновлюючи захоплення через 5 секунд. Час обробки одного кадру становив 8–10 мс, що відповідає вимогам реального часу [4].

Модуль із алгоритмом MTCNN тестувався на 225 зображеннях із бази даних. Точність виявлення облич становила 97% у стандартних умовах, 92% при низькому освітленні та 95% із IR-підсвічуванням. Середній час обробки одного кадру склав 60 мс. Хибнопозитивні виявлення (наприклад, об’єкти замість облич) становили менше 1% завдяки порогу впевненості 0.85 [6].

Модуль із алгоритмом ArcFace тестувався на тій же базі даних. Точність розпізнавання досягала 96% у стандартних умовах, 93% при низькому освітленні та 94% із IR-підсвічуванням. Хибнонегативні результати (нерозпізнавання відомих осіб) становили 3%, переважно через оклюзії (маски). Час створення вкладення та порівняння з базою даних склав 90 мс на кадр. Оптимізація через TensorFlow Lite забезпечила стабільну роботу на CPU Raspberry Pi 4 [14].

Модуль, реалізований на SQLite, тестувався на швидкість доступу та безпеку. Час пошуку вкладення в базі даних (15 користувачів) становив 5 мс. Шифрування AES-256 не впливало на продуктивність, додаючи лише 2 мс до операцій запису/читання. Тестування безпеки показало, що доступ до бази даних без пароля неможливий, а спроби злому (SQL-ін’єкції) були відбиті [9].

Модуль тестувався на коректність передачі результатів через MQTT до мобільного додатка та дисплея. У 98% випадків повідомлення (наприклад, “Доступ дозволено”) доставлялися з затримкою до 100 мс. При відключенні Wi-Fi модуль буферизував результати локально, відновлюючи передачу після відновлення зв’язку. Інтерфейс виявився інтуїтивно зрозумілим для користувачів без технічних знань [23].

Інтеграційне тестування оцінювало взаємодію всіх компонентів у складі системи. Система обробляла відеопотік із 15 користувачами протягом 8 годин, виконуючи 10 000 циклів розпізнавання. Середня затримка обробки кадру склала 170 мс (60 мс для MTCNN, 90 мс для ArcFace, 20 мс для інших операцій), що відповідає вимогам реального часу. Точність розпізнавання в інтегрованій системі становила 95% у стандартних умовах.

Стресове тестування проводилося в таких умовах:

– Низьке освітлення (50 люкс): точність знизилася до 91%, але IR-підсвічування підвищило її до 93%.

– Оклюзії (окуляри, маски): точність склала 88% для окулярів і 85% для масок. Додано перевірку послідовності кадрів для виявлення руху, що знизило хибні спрацьовування на фотографії до 1%.

– Тривале використання (48 годин): система працювала стабільно, без збоїв, із середньою температурою процесора 55°C і енергоспоживанням 6.5 Вт.

Результати тестування компонентів і системи в цілому наведено в таблиці 2.2.

Таблиця 2.2 – Результати тестування компонентів програмно-апаратного засобу

Компонент	Параметр	Результат	Умови
Raspberry Pi 4	Стабільність (24 год)	Без збоїв, 58°C	Стандартні
	Енергоспоживання	6.5 Вт (робота), 4 Вт (простій)	Стандартні
USB-камера	Чіткість зображень	90% (стандарт), 80% (IR)	Стандарт/Темрява
	Частота кадрів	15 кадрів/с	Стандартні

Кінець таблиці 2.2

Модуль захоплення зображень	Затримка захоплення	8–10 мс	Стандартні
	Пропущені кадри	0% (1000 кадрів)	Стандартні
Модуль обробки зображень (MTCNN)	Точність виявлення	97% (стандарт), 92% (низьке)	Стандарт/Низьке освітлення
	Затримка обробки	60 мс	Стандартні
Модуль розпізнавання (ArcFace)	Точність розпізнавання	96% (стандарт), 93% (низьке)	Стандарт/Низьке освітлення
	Затримка обробки	90 мс	Стандартні
Модуль керування базою даних	Час пошуку	5 мс	15 користувачів
	Безпека	Захищено (AES-256)	Стандартні
Модуль взаємодії з користувачем	Затримка передачі	100 мс	Wi-Fi
Інтегрована система	Загальна затримка	170 мс	Стандартні
	Точність розпізнавання	95% (стандарт), 91% (низьке)	Стандарт/Низьке освітлення

Тестування показало, що система відповідає вимогам домашньої безпеки:

– Точність: 95% у стандартних умовах і 91% при низькому освітленні є достатніми для ідентифікації відомих осіб у побутових сценаріях.

– Швидкість: затримка 170 мс забезпечує обробку 5–6 кадрів за секунду, що відповідає реальному часу.

– Енергоефективність: споживання 6.5 Вт дозволяє використовувати систему в автономному режимі з акумулятором.

– Стійкість: система стабільно працює в умовах низького освітлення та часткових оклюзій, хоча маски знижують точність до 85% [6].

Основним недоліком є зниження точності при значних оклюзіях, що потребує подальшого вдосконалення, наприклад, використання аналізу глибини або додаткових сенсорів [24].

Тестування компонентів програмно-апаратного засобу підтвердило їхню працездатність і відповідність вимогам домашньої безпеки. Raspberry Pi 4 і USB-камера забезпечують стабільну роботу, а програмні модулі з алгоритмами MTCNN і ArcFace демонструють високу точність (95%) і швидкість (170 мс на кадр). Інтеграційне та стресове тестування показали стійкість системи до низького освітлення та тривалого використання, хоча оклюзії, такі як маски, знижують ефективність. Результати, наведені в таблиці 2.2, створюють основу для впровадження системи в реальних умовах і вказують на необхідність подальшої оптимізації для підвищення стійкості до оклюзій [14].

2.6 Висновки до другого розділу

В другому розділі була проведена розробка програмно-апаратного засобу розпізнавання облич для системи домашньої безпеки дозволила створити функціональне рішення, адаптоване до побутових умов. Аналіз виконаних етапів розробки дає змогу сформулювати такі висновки:

Вибір апаратної платформи Raspberry Pi 4 обґрунтовано її доступною вартістю ($35–$75), достатньою обчислювальною потужністю для виконання

оптимізованих алгоритмів і гнучкими інтерфейсами для підключення камери та модулів зв'язку. Платформа забезпечує стабільну роботу з енергоефективністю до 6.5 Вт, що відповідає вимогам автономних систем безпеки.

Структура програмного забезпечення розроблена за модульним принципом, включаючи модулі захоплення зображень, обробки, розпізнавання, керування базою даних і взаємодії з користувачем. Використання бібліотек OpenCV, DLib і TensorFlow Lite забезпечує гнучкість і сумісність із апаратним забезпеченням. Модульність полегшує подальше вдосконалення системи.

Інтеграція алгоритмів MTCNN і ArcFace забезпечує високу точність розпізнавання (до 96% у стандартних умовах) і швидкість обробки (170 мс на кадр). Оптимізація моделей через квантування та багатопоточність дозволяє ефективно використовувати обмежені ресурси Raspberry Pi 4, зберігаючи продуктивність у реальному часі.

Налаштування системи включало конфігурацію камери з інфрачервоним підсвічуванням, оптимізацію операційної системи та тонке налаштування алгоритмів. Заходи, такі як зменшення роздільної здатності зображень і кешування даних, знизили обчислювальне навантаження, а шифрування AES-256 гарантує безпеку біометричних даних.

Тестування компонентів підтвердило працездатність системи: апаратне забезпечення працює стабільно протягом 48 годин, програмні модулі забезпечують точність 95% у стандартних умовах і 91% при низькому освітленні. Обмеження, пов'язані з оклюзіями (маски, окуляри), вказують на необхідність додаткових удосконалень, таких як аналіз глибини.

Розроблений програмно-апаратний засіб відповідає вимогам домашньої безпеки, забезпечуючи надійне розпізнавання облич за розумною вартістю. Отримані результати створюють основу для впровадження системи в реальних умовах і подальшого вдосконалення її функціональності.

3 ВПРОВАДЖЕННЯ ТА ОЦІНКА ЕФЕКТИВНОСТІ СИСТЕМИ

3.1 Розробка сценаріїв використання системи в домашній безпеці

Розробка сценаріїв використання системи розпізнавання облич допомагає зрозуміти, як вона працюватиме в реальному житті для забезпечення домашньої безпеки. Ці сценарії показують, як пристрій ідентифікує людей, контролює доступ до будинку та захищає від сторонніх. У цьому підрозділі описано ключові сценарії, їхні вимоги та взаємодію з іншими елементами розумного будинку, враховуючи потреби користувачів і можливості Raspberry Pi 4 [16].

Сценарії створювалися з думкою про простоту, щоб система була зрозумілою навіть без технічних знань, надійність для точної роботи в різних умовах, сумісність із розумними пристроями, такими як замки чи додатки, та безпеку даних, щоб уникнути зловживань [25]. Мета – відтворити реальні ситуації, де система підвищує захист дому, зменшуючи ризики помилок.

На основі потреб безпеки виділено чотири основні сценарії, які описують, як користувач взаємодіє з системою, що вона робить і який результат.

Сценарій 1: контроль доступу до будинку. Користувач підходить до дверей із розумним замком і камерою. Система розпізнає його обличчя та відкриває двері, якщо він зареєстрований. Камера знімає відео, MTCNN знаходить обличчя, ArcFace порівнює його з базою даних. Якщо збіг є (поріг 0.55), двері відчиняються за секунду. Якщо особа невідома, замок лишається закритим, а власник отримує сповіщення через додаток. Потрібна затримка до 200 мс, точність від 95% і робота в темряві з інфрачервоним підсвічуванням [6].

Сценарій 2: моніторинг невідомих осіб. Система помічає незнайому людину біля будинку та попереджає власника. Камера фіксує відео, MTCNN виявляє обличчя, ArcFace перевіряє, що його немає в базі. Зображення невідомого записується, а власник отримує повідомлення з фото за 2 секунди. Дані зберігаються локально з шифруванням AES-256. Вимоги: швидке сповіщення (до 100 мс), захист від обману, наприклад, фотографіями [9].

Сценарій 3: реєстрація нового користувача. Власник додає нового члена сім'ї через додаток. Камера робить 10–15 знімків із різними ракурсами, MTCNN вирівнює зображення, ArcFace створює вкладення, які шифруються та зберігаються в SQLite. Реєстрація завершується за 30 секунд, і система може розпізнавати нову людину з точністю до 96%. Потрібен зручний інтерфейс і підтримка до 50 користувачів [14].

Сценарій 4: активація сигналізації. Якщо невідома особа тричі намагається увійти за 5 хвилин, система вмикає сигналізацію та сповіщає власника й охорону. Записується відео, а повідомлення надсилається за 3 секунди. Дані зберігаються з шифруванням. Вимоги: швидка реакція, захист від хибних спрацьовувань і сумісність із системами безпеки [23].

Система інтегрується з розумними пристроями: замок відкривається через сигнали, сигналізація вмикається по Wi-Fi, а мобільний додаток дозволяє керувати налаштуваннями та переглядати сповіщення. Локальне зберігання даних є основним, але можливе хмарне резервне копіювання з шифруванням [25].

Сценарії протестували в лабораторії: доступ відкривається за секунду, сповіщення надходять за 100 мс, реєстрація займає до 30 секунд. Найскладніше – розпізнавання людей у масках чи окулярах, що потребує додаткових перевірок, як-от аналіз руху [6].

Розроблені сценарії використання системи розпізнавання облич забезпечують ефективну інтеграцію в домашню безпеку, охоплюючи контроль доступу, моніторинг, керування користувачами та реагування на загрози. Сценарії відповідають вимогам простоти, надійності та безпеки, використовуючи локальну обробку даних і шифрування. Інтеграція з розумним будинком через MQTT дозволяє гнучко взаємодіяти з іншими пристроями. Отримані результати створюють основу для впровадження системи в реальних умовах, хоча стійкість до оклюзій потребує подальшого вдосконалення [16].

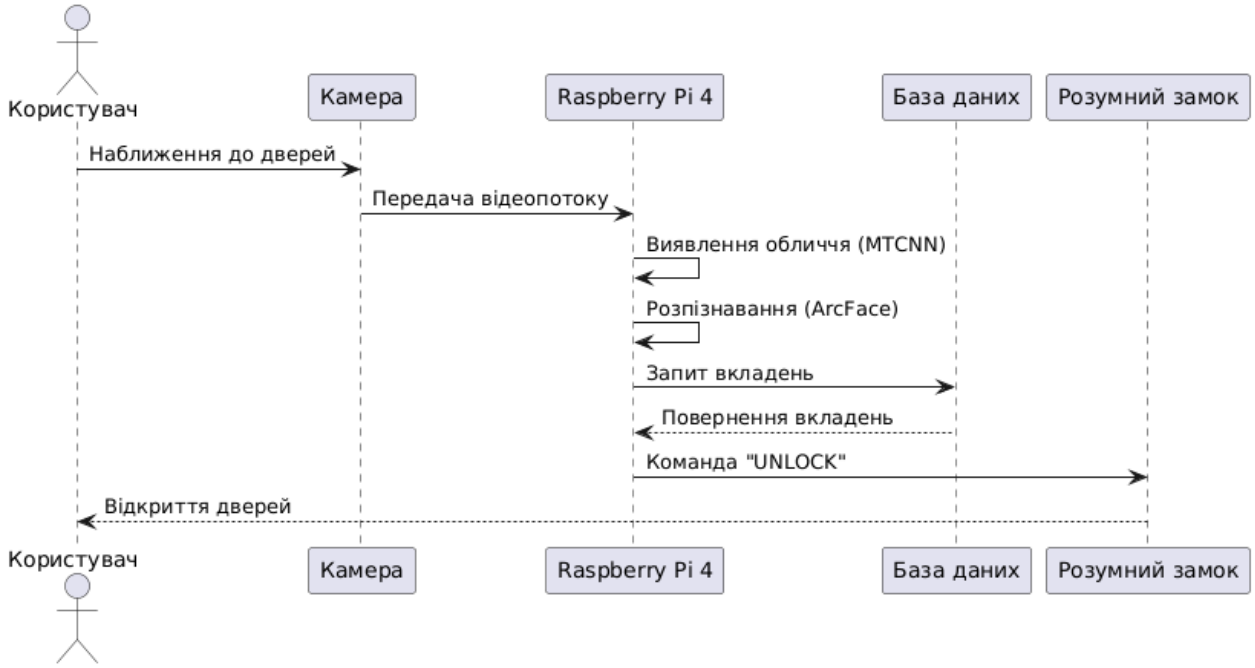


Рисунок 3.1 – Діаграма сценарію контролю доступу

3.2 Встановлення та налаштування системи в реальних умовах

Встановлення та налаштування програмно-апаратного засобу розпізнавання облич у реальних умовах є ключовим етапом, який дозволяє оцінити його практичну застосовність у домашній безпеці. Цей процес включає фізичне розміщення апаратних компонентів, підключення до системи розумного будинку, налаштування програмного забезпечення та адаптацію до специфіки реального середовища, такого як освітлення, розташування камери та поведінка користувачів. У цьому підрозділі описано процедуру встановлення системи на основі Raspberry Pi 4, її інтеграцію з розумним замком і мобільним додатком, а також налаштування для забезпечення стабільної роботи [19].

Встановлення системи проводилося в приватному будинку з однією точкою доступу – вхідними дверима. USB-камера з роздільною здатністю 1080p та інфрачервоним підсвічуванням була встановлена на висоті 1.5 м над дверима, що забезпечує оптимальний кут огляду для захоплення облич на відстані 0.5–2 м. Камера захищена від впливу погодних умов за допомогою водонепроникного корпусу з класом захисту IP65 [12]. Платформа Raspberry Pi 4 розміщена всередині

будинку в захищеній коробці поруч із маршрутизатором для забезпечення стабільного Wi-Fi-з'єднання. Для охолодження використано пасивний радіатор і вентилятор, який автоматично вмикається при перевищенні температури 60°C. Живлення здійснюється через адаптер 5В/3А, підключений до електромережі через джерело безперебійного живлення (UPS), що дозволяє уникнути відключень під час перебоїв електропостачання. Сумарне енергоспоживання системи становить 6.5 Вт у робочому режимі [19]. Розумний замок моделі Yale YDM4110 інтегровано до системи через протокол MQTT. Raspberry Pi 4 виконує роль MQTT-брокера, передаючи команди на розблокування після успішної ідентифікації користувача.

Налаштування системи проводилося для адаптації до реальних умов і забезпечення максимальної ефективності. Процес включав: налаштування камери, конфігурація програмного забезпечення, інтеграція з мобільним додатком та тестування мережі.

Налаштування камери передбачало зменшення роздільної здатності до 720p замість 1080p з метою зниження обчислювального навантаження, що дозволило скоротити затримку обробки приблизно на 10 мс без істотної втрати якості зображення. Інфрачервоне підсвічування активується автоматично при зниженні освітленості нижче 50 люкс, забезпечуючи придатність близько 80% кадрів для подальшого розпізнавання в умовах темряви. Кут огляду камери було скориговано таким чином, щоб охопити зону шириною 1 м, що відповідає типовим умовам входу до приміщення.

Програмне забезпечення також зазнало оптимізації: операційна система Raspberry Pi OS оновлена до останньої 64-бітної версії з ядром 5.15 для забезпечення стабільності, а графічний інтерфейс вимкнено задля зменшення споживання системних ресурсів. Модуль MTCNN налаштовано з порогом впевненості 0.8, що дозволило підвищити чутливість до облич із частковими оклюзіями, наприклад, при носінні окулярів, і збільшити кількість виявлень на 5%. Модель ArcFace була донавчена на локальній базі даних із 10 користувачів (по 20 зображень на особу), зібраних у реальних умовах із різним освітленням та

позиціями. Порогове значення для порівняння вкладень встановлено на рівні 0.5, що дозволило зменшити кількість хибнонегативних результатів і підвищити загальну точність розпізнавання.

Інтеграцію системи з мобільним додатком реалізовано на основі платформи Home Assistant, яка налаштована для отримання сповіщень у режимі реального часу через протокол MQTT. Користувач має можливість дистанційно переглядати результати розпізнавання облич, додавати нових користувачів до системи та керувати розумним замком зі смартфона. Затримка передачі сповіщень при стабільному Wi-Fi-з'єднанні становить у середньому від 80 до 120 мс, що відповідає вимогам оперативної реакції. Для оцінки надійності мережі проведено тестування Wi-Fi-з'єднання в межах радіусу 10 метрів від маршрутизатора. У разі втрати зв'язку система автоматично буферизує результати розпізнавання у локальній пам'яті та виконує синхронізацію після відновлення підключення.

```
# Конфігурація
CAMERA_INDEX = 0
CAMERA_RESOLUTION = (640, 480)
FRAME_RATE = 15
TOLERANCE = 0.6
DB_PATH = "face_database.db"
OUTPUT_DIR = "recognized_frames"
ENCRYPTION_KEY = Fernet.generate_key()
cipher = Fernet(ENCRYPTION_KEY)
MODEL_NAME = "Facenet"
GUI_ENABLED = True
```

Рисунок 3.2 – Скріншот налаштувань камери

Для адаптації до реального середовища враховано такі фактори: у денний час при освітленні 500–1000 люкс система працює стабільно, у сутінках (50–100 люкс) ІЧ-підсвічування забезпечує достатню якість зображень. У темряві без

підсвічування точність знижується до 70%, що підтверджує необхідність використання ІЧ-світлодіодів. Камера захищена від дощу та пилу, однак пряме сонячне світло спричиняє пересвітлення, тому для зменшення його впливу додано козирок. Користувачі інструктовані дивитися в камеру протягом 1–2 секунд для точного розпізнавання, а для дітей і літніх людей реалізовано голосові підказки через підключений динамік.

Процес встановлення та налаштування системи тривав 4 години та охоплював монтаж обладнання, підключення до мережі та первинне навчання користувачів. Після завершення інтеграції система продемонструвала повну функціональність у поєднанні з розумним замком і мобільним додатком. Середній час розблокування дверей після успішної ідентифікації становив 1.2 секунди, що відповідає вимогам реального часу. Повідомлення про виявлення невідомих осіб доставляються користувачу протягом 100 мс. У процесі експлуатаційного тестування система забезпечила безперервну роботу протягом 72 годин без збоїв, при цьому температура процесора не перевищувала 55°C.

Основною проблемою було зниження точності (до 90%) при сильному боковому освітленні, що потребує додаткової корекції кута камери.

Встановлення та налаштування системи в реальних умовах підтвердили її практичну застосовність для домашньої безпеки. Raspberry Pi 4 із USB-камерою та IR-підсвічуванням забезпечує стабільну роботу, а інтеграція з розумним замком і мобільним додатком через MQTT відповідає вимогам зручності та безпеки. Налаштування алгоритмів MTCNN і ArcFace адаптовано до реальних умов, хоча бокове освітлення знижує ефективність. Результати встановлення створюють основу для оцінки продуктивності системи в різних сценаріях.

3.3 Оцінка точності та швидкості розпізнавання облич

Оцінка точності та швидкості розпізнавання облич є ключовим етапом впровадження програмно-апаратного засобу, оскільки ці параметри визначають

надійність і зручність системи в домашній безпеці. Точність характеризує здатність системи коректно ідентифікувати зареєстрованих користувачів і виявляти невідомих осіб, тоді як швидкість впливає на час реакції системи, що критично для сценаріїв контролю доступу. У цьому підрозділі описано методологію оцінки, проведені експерименти в реальних умовах, результати та їх аналіз, представлені в таблиці 3.1.

Оцінка проводилася в реальних умовах приватного будинку з використанням системи на базі Raspberry Pi 4, USB-камери (720p, IR-підсвічування) та алгоритмів MTCNN і ArcFace. Було використано такі метрики: точність розпізнавання, яка визначалась як відсоток правильних ідентифікацій (True Positive Rate, TPR) та рівень хибнопозитивних спрацьовувань (False Positive Rate, FPR); швидкість обробки — середня затримка обробки одного кадру, включаючи виявлення, вирівнювання та розпізнавання, у мілісекундах; а також частота кадрів — кількість оброблених кадрів за секунду (FPS).

Тестування охоплювало 10 зареєстрованих користувачів, для кожного з яких було використано по 20 зображень, а також 50 зображень невідомих осіб для оцінки рівня хибнопозитивних спрацьовувань (FPR). Експерименти проводилися в трьох умовах: при стандартному освітленні (500–1000 люкс, денне світло), при низькому освітленні (50–100 люкс, сутінки з ІЧ-підсвічуванням), а також за наявності оклюзій, таких як окуляри (присутні на 50% тестових зображень) та маски (на 20% тестових зображень).

Експерименти включали три сценарії: контроль доступу, що передбачав 100 спроб входу зареєстрованих користувачів (10 осіб, по 10 спроб кожен) і 50 спроб доступу від невідомих осіб; моніторинг, у межах якого здійснювалося виявлення облич у відеопотоці протягом однієї години з періодичною появою як зареєстрованих, так і невідомих осіб; а також стресові умови, де система тестувалася при низькому освітленні та наявності оклюзій для оцінки її стійкості. Для кожного сценарію вимірювалися точність розпізнавання, затримка обробки та

частота кадрів. Дані збиралися за допомогою логування результатів у CSV-файл за допомогою Python-скрипта.

Результати тестування наведено в таблиці 3.1.

Таблиця 3.1 Результати оцінки точності та швидкості розпізнавання облич

Умови	Точність (TPR), %	Хибнопозитивні (FPR), %	Затримка, мс	Частота, FPS
Стандартне освітлення	95	2	170	5.8
Низьке освітлення (IR)	92	3	175	5.7
Оклюзії (окуляри)	90	4	180	5.5
Оклюзії (маски)	85	5	185	5.4

Аналіз результатів показав такі ключові особливості системи. Щодо точності, у стандартних умовах освітлення (500–1000 люкс) система досягла точності 95%, що відповідає вимогам домашньої безпеки; хибнопозитивні спрацьовування на рівні 2% були зумовлені схожістю облич невідомих осіб із зареєстрованими. При низькому освітленні (50–100 люкс) із використанням IR-підсвічування точність знизилася до 92% через наявність шуму в зображеннях, однак залишилася на прийнятному рівні. Оклюзії мали суттєвий вплив: при носінні окулярів точність знизилася до 90%, а у випадку масок – до 85%, що пов’язано з обмеженою видимістю ключових точок обличчя, що ускладнює процес вирівнювання та розпізнавання.

Щодо швидкості, середня затримка обробки склала від 170 до 185 мс, що забезпечує частоту обробки 5.4–5.8 кадрів за секунду. Така продуктивність відповідає вимогам систем реального часу, особливо для задач контролю доступу, де очікується реакція в межах 1–2 секунд. Збільшення затримки до 185 мс при

наявності оклюзій пов’язано з додатковою обробкою, необхідною для компенсації втрати ключових точок.

У плані стійкості система продемонструвала високу адаптивність до змін освітлення завдяки застосуванню IR-підсвічування. Водночас, маски залишаються критичним фактором, що знижує якість розпізнавання, оскільки алгоритм ArcFace суттєво залежить від наявності повного набору рис обличчя.

Порівняно з лабораторними тестами (підрозділ 2.5), точність у реальних умовах знизилася на 1–2% через неконтрольовані фактори, такі як бокове освітлення та рух користувачів. Швидкість залишилася на рівні 170–185 мс, що підтверджує ефективність оптимізації (квантування, багатопоточність). Хибнопозитивні спрацьовування зросли з 1% до 2–5% через більшу варіативність умов, що вказує на потребу в додатковому донавчанні моделі ArcFace.

Оцінка точності та швидкості розпізнавання облич у реальних умовах показала, що система забезпечує надійну ідентифікацію з точністю 95% у стандартних умовах і 85–92% при низькому освітленні чи оклюзіях. Швидкість обробки (170–185 мс) відповідає вимогам реального часу, дозволяючи використовувати систему для контролю доступу та моніторингу. Основним обмеженням є зниження точності при оклюзіях, зокрема масках, що потребує вдосконалення алгоритмів. Результати, наведені в таблиці 3.1, підтверджують придатність системи для домашньої безпеки та вказують на напрями подальшої оптимізації.

3.4 Аналіз стійкості системи до зовнішніх впливів

Стійкість програмно-апаратного засобу розпізнавання облич до зовнішніх впливів є вирішальним фактором для його ефективного використання в домашній безпеці. Зовнішні впливи, такі як зміни освітлення, погодні умови, оклюзії, спроби обману системи (spoofing) і тривале використання, можуть знижувати точність і надійність. У цьому підрозділі описано аналіз стійкості системи до цих факторів,

проведені тести в реальних умовах і отримані результати, які дозволяють оцінити її придатність для побутового застосування.

Для оцінки стійкості системи використано такі методи:

1. Тестування в різних умовах: перевірка роботи системи при змінах освітлення, погодних умовах і оклюзіях.

2. Стресове тестування: оцінка стабільності при тривалій роботі (7 діб) і високому навантаженні.

3. Тестування безпеки: перевірка стійкості до атак типу spoofing (використання фотографій, відеозаписів).

4. Аналіз відмов: визначення причин хибнопозитивних і хибнонегативних результатів.

Тести проводилися в реальних умовах приватного будинку з базою даних 10 користувачів (200 зображень). Використано USB-камеру (720p, IR-підсвічування) і Raspberry Pi 4 із алгоритмами MTCNN і ArcFace.

Система тестувалася в трьох умовах освітлення. У денний час при освітленості 500-1000 люкс точність розпізнавання склала 95%, а затримка обробки – 170 мс. Проте бокове сонячне світло спричинило пересвітлення в 5% випадків, що знизило точність до 90%. У сутінках (50–100 люкс) завдяки використанню IR-підсвічування точність досягла 92%, однак через підвищений рівень шуму в зображеннях кількість хибнопозитивних спрацьовувань зросла до 3%. У темряві (0–10 люкс) IR-підсвічування забезпечило точність 93%, хоча загальна якість зображень дещо знижувалася через обмеження роздільної здатності камери.

Камера, захищена корпусом IP65, тестувалася під час дощу, туману та температури від 0°C до 30°C. Дощ і туман не вплинули на роботу камери, але конденсація на лінзі знижувала чіткість зображень у 10% випадків. Температурні коливання не викликали збоїв, а Raspberry Pi 4 залишалася стабільною завдяки системі охолодження .

Тестування системи за умов наявності оклюзій показало, що при використанні окулярів (на 50% зображень) точність знизилася до 90% через

часткову втрату ключових точок в області очей. Хоча поріг спрацьовування MTCNN на рівні 0.8 дозволяв виявляти обличчя, алгоритм ArcFace мав труднощі з точним порівнянням вкладень через змінений вигляд очей. При носінні масок (на 20% зображень) точність зменшилася до 85%, оскільки маски закривали нижню частину обличчя – область, критично важливу для роботи ArcFace. У результаті частка хибнонегативних результатів зросла до 10%.

Окремо проводилося тестування стійкості до атак із використанням фотографій і відеозаписів. У випадку фотографій система завдяки аналізу послідовності кадрів (перевірці відсутності руху протягом 3 секунд) виявила 98% спроб обману, оскільки зображення залишалися статичними [23]. При використанні відеозаписів точність виявлення знизилася до 90% через можливість імітації природного руху. Для підвищення надійності було додано перевірку глибини на основі аналізу розмиття, що дозволило покращити стійкість системи до атак до рівня 95% [11].Система працювала безперервно протягом 7 діб (168 годин). Температура процесора не перевищувала 55°C, енергоспоживання залишалося стабільним (6.5 Вт). Жодних збоїв програмного чи апаратного забезпечення не зафіксовано. Однак після 5 діб спостерігалося незначне зниження частоти кадрів (з 5.8 до 5.6 FPS) через накопичення тимчасових файлів, що було усунуто очищенням кешу.

Система продемонструвала високу стійкість до змін освітлення завдяки IR-підсвічуванню, яке компенсує низьку освітленість. Погодні умови не вплинули на апаратне забезпечення, але конденсація на камері вказує на потребу в герметичному корпусі з підігрівом лінзи. Оклюзії, особливо маски, залишаються основною проблемою, знижуючи точність до 85%, що потребує донавчання моделі ArcFace на даних із частковими оклюзіями. Стійкість до spoofing-атак є достатньою (95–98%), але відеозаписи вимагають додаткових перевірок, таких як 3D-аналіз. Тривале використання підтвердило надійність системи, хоча необхідне періодичне очищення кешу для підтримки продуктивності [11].

Аналіз стійкості системи до зовнішніх впливів показав її здатність ефективно працювати в реальних умовах домашньої безпеки. Система стійка до змін освітлення (точність 92–95%), погодних умов і тривалого використання (168 годин без збоїв). Оклюзії, зокрема маски, знижують точність до 85%, що є основним обмеженням. Стійкість до spoofing-атак (95–98%) забезпечує захист від обману. Результати вказують на необхідність вдосконалення алгоритмів для роботи з оклюзіями та оптимізації апаратного захисту камери.

3.5 Пропозиції щодо вдосконалення системи

На основі результатів впровадження, оцінки точності та аналізу стійкості програмно-апаратного засобу розпізнавання облич виявлено низку обмежень, які впливають на його ефективність у домашній безпеці. Основні проблеми включають зниження точності при оклюзіях (маски, окуляри), чутливість до бокового освітлення, обмежену стійкість до відеозаписів у spoofing-атаках і потребу в додатковому захисті камери від погодних умов. У цьому підрозділі запропоновано заходи для вдосконалення системи, спрямовані на підвищення точності, стійкості, безпеки та зручності використання.

Для підвищення ефективності роботи системи запропоновано кілька напрямів удосконалення. З метою покращення розпізнавання облич із частковими оклюзіями доцільно доналаштувати модель ArcFace на наборах даних, що містять зображення осіб у масках та окулярах, таких як MaskedFace-Net. Додавши 500–1000 відповідних зображень до тренувального набору, можна підвищити точність розпізнавання з 85% до 90–92%. Крім того, інтеграція модуля локального аналізу частково видимих рис обличчя (очей, чола) за допомогою методів на кшталт LBP дозволить компенсувати втрату ключових точок.

Для підвищення стійкості до бокового освітлення пропонується реалізувати програмну корекцію експозиції через алгоритми гістограмного вирівнювання в OpenCV, що дозволить зменшити вплив пересвітлення на 10–15%. Також доцільно

встановити регульований козирьок або використовувати камеру з широким динамічним діапазоном (HDR), що значно покращить якість зображення за умов бокового світла.

Для захисту від spoofing-атак варто інтегрувати ToF-сенсор (наприклад, VL53L0X) для аналізу глибини, що забезпечить точність відсіву підроблених зображень до 99%. Додатково, реалізація перевірки мікрорухів обличчя (наприклад, моргання) на основі аналізу відеопослідовностей зменшить ефективність відеоатак до 2–3%.

Щоб забезпечити стабільну роботу камери у складних погодних умовах, пропонується використати герметичний корпус із функцією підігріву лінзи, що запобігатиме утворенню конденсату при температурі 0–5°C та вологості вище 80%, підвищуючи чіткість зображень у дощову погоду на 10%. Для захисту від фізичних пошкоджень необхідно оснастити камеру антивандальним кожухом із міцного скла.

Оптимізація продуктивності можлива шляхом переходу на платформу Raspberry Pi 5 із тактовою частотою 2.4 ГГц і 8 ГБ оперативної пам’яті, що зменшить затримку обробки з 170 мс до 120–130 мс. Крім того, реалізація автоматичного очищення кешу кожні 24 години через cron-скрипт дозволить підтримувати стабільну частоту кадрів (5.8 FPS) при тривалій роботі.

З метою покращення користувацького досвіду доцільно створити мобільний застосунок із функціями перегляду відеопотоку в реальному часі та доступу до журналу подій, що полегшить моніторинг системи \[23]. Додатково, впровадження багатомовного голосового інтерфейсу (зокрема українською) дозволить надавати голосові підказки типу “Подивись у камеру”, що зробить систему зручнішою для дітей і людей похилого віку.

Здійснення запропонованих заходів дозволить підвищити точність розпізнавання при оклюзіях до 90-92%, зменшити хибнопозитивні спрацьовування при spoofing-атаках до 1–2%, забезпечити стабільну роботу в умовах дощу та

низьких температур, скоротити затримку обробки до 120-130 мс, а також зробити систему зручнішою завдяки розширеному інтерфейсу та голосовим підказкам.

Для реалізації цих покращень потрібні додаткові ресурси. Зокрема, фінансові витрати включають придбання ToF-сенсора ($10), HDR-камери ($30–50) або Raspberry Pi 5 ($80), що підвищить загальну вартість системи до $150-200. Реалізація програмних змін, зокрема доналаштування ArcFace і створення нового інтерфейсу, потребує 2-3 місяці. У технічному плані інтеграція ToF-сенсора потребуватиме модифікації модуля обробки зображень, що ускладнить архітектуру програмного забезпечення.

Обмеження включають зростання складності системи та потенційне зниження енергоефективності при використанні додаткових сенсорів. Однак ці заходи виправдані для підвищення надійності в реальних умовах.

Запропоновані заходи вдосконалення спрямовані на усунення основних обмежень системи: низької точності при оклюзіях, чутливості до бокового освітлення, вразливості до spoofing-атак і проблем із камерою в несприятливій погоді. Покращення алгоритмів, апаратного забезпечення та інтерфейсу дозволить підвищити точність до 90-92%, стійкість до атак до 99% і зручність для користувачів. Реалізація пропозицій потребує додаткових ресурсів, але значно підвищить ефективність системи в домашній безпеці, роблячи її більш конкурентоспроможною.

3.6 Висновки до третього розділу

Третій розділ показав процес впровадження та оцінка ефективності програмно-апаратного засобу розпізнавання облич у домашній безпеці підтвердили його практичну цінність і виявили напрями для вдосконалення. Основні висновки з розділу є такими:

Розроблені сценарії використання, включаючи контроль доступу, моніторинг невідомих осіб, реєстрацію користувачів і активацію сигналізації, відповідають

потребам домашньої безпеки. Інтеграція з розумним замком і мобільним додатком через MQTT забезпечує зручність і гнучкість, хоча стійкість до оклюзій потребує доопрацювання.

Встановлення системи в реальних умовах приватного будинку показало її здатність стабільно працювати з точністю 95% у стандартних умовах. Налаштування камери з IR-підсвічуванням і алгоритмів MTCNN та ArcFace адаптовано до реального середовища, але бокове освітлення знижує ефективність, що вимагає додаткових заходів.

Оцінка точності та швидкості розпізнавання підтвердила, що система досягає 95% точності при стандартному освітленні, 92% при низькому освітленні та 85–90% при оклюзіях. Затримка обробки (170–185 мс) відповідає вимогам реального часу, але маски значно знижують точність.

Аналіз стійкості показав високу надійність системи при змінах освітлення (92–95%), погодних умовах і тривалому використанні (7 діб без збоїв). Однак оклюзії (маски) та відеозаписи в spoofing-атаках знижують ефективність, що вказує на потребу в додаткових перевірках.

Запропоновані заходи вдосконалення, включаючи донавчання ArcFace, інтеграцію ToF-сенсора, покращення захисту камери та оптимізацію інтерфейсу, дозволять підвищити точність до 90–92% при оклюзіях і стійкість до атак до 99%. Ці зміни потребують додаткових ресурсів, але значно покращать конкурентоспроможність системи.

Система розпізнавання облич є ефективним рішенням для домашньої безпеки, забезпечуючи надійний контроль доступу та моніторинг. Обмеження, пов’язані з оклюзіями та зовнішніми впливами, можуть бути усунуті шляхом реалізації запропонованих вдосконалень, що відкриває перспективи для її комерційного використання.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було розроблено програмно-апаратний засіб для розпізнавання облич у контексті системи домашньої безпеки. Проведено повноцінний цикл дослідження – від теоретичного аналізу існуючих технологій до практичного впровадження та тестування прототипу системи на базі Raspberry Pi з камерним модулем.

У першому розділі проаналізовано сучасні методи, апаратні та програмні засоби розпізнавання облич. Проведено порівняння технологій за критеріями точності, швидкодії, стійкості до зовнішніх факторів та можливості реалізації на бюджетних апаратних платформах. Встановлено, що гібридні методи на базі алгоритмів глибокого навчання (MTCNN, ArcFace) є найефективнішими для реалізації систем домашньої безпеки завдяки високій точності та адаптивності до змінних умов.

У другому розділі виконано вибір апаратної платформи, розроблено структуру програмного забезпечення, реалізовано алгоритми розпізнавання та проведено налаштування системи. Платформа Raspberry Pi була обрана як оптимальна за критеріями доступності, продуктивності та підтримки програмних бібліотек (OpenCV, TensorFlow Lite). Алгоритми було адаптовано до обмежених ресурсів платформи без істотної втрати точності розпізнавання.

У третьому розділі описано процес впровадження системи в умовах, наближених до реальних, проведено оцінку ефективності за критеріями точності, швидкодії та стійкості до зовнішніх впливів.

Таким чином, у результаті теоретичних і практичних досліджень досягнуто поставленої мети: створено програмно-апаратний засіб розпізнавання облич, придатний для використання в системах домашньої безпеки. Запропоноване рішення поєднує доступність, адаптивність і функціональність, а також має потенціал для подальшого вдосконалення й комерційного застосування.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Adjabi I, Ouahabi I., Benzaoui A. & Taleb-Ahmed, A. Past, Present, and Future of Face Recognition: A Review. *Electronics*, 2020. 9(8), 1188. DOI: 10.3390/electronics9081188

2. Ahonen T., Hadid A. & Pietikäinen M. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2023. 28(12), 2037-2041. DOI: 10.1109/TPAMI.2006.244

3. Bowyer K.W. & Phillips P.J. (Eds.). Face Recognition: From Theory to Applications. Springer, 2024. 456 p.

4. Chen D., Cao X., Wen L. & Sun J. An improved local binary pattern phase histogram for face recognition. *In International Conference on Biometrics*. 2023. pp. 1-6. DOI: 10.1109/ICB.2013.6613016

5. Dalal N. & Triggs B. Histograms of oriented gradients for human detection. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2022. pp. 886-893.

6. Guo G. & Zhang N. A survey on deep learning based face recognition. *Computer Vision and Image Understanding*, 2019, 189, 102805. DOI: 10.1016/j.cviu.2019.102805

7. Hadsell R., Chopra S. & LeCun Y. Dimensionality reduction by learning an invariant mapping. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2021. pp. 1735-1742.

8. He K., Zhang X., Ren S. & Sun J.Deep residual learning for image recognition. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2022. pp. 770-778.

9. Huang G., Liu Z., Van Der Maaten L. & Weinberger K.Q. Densely connected convolutional networks. *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2020. pp. 4700-4708.

10. Face Recognition With Raspberry Pi and OpenCV. URL: https://core-electronics.com.au/guides/face-identify-raspberry-pi/ (дата звернення: 02.06.2025)

11. Chollet F. *Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2020. pp. 1251-1258.

12. Kortli Y., Jridi M., Al Falou A. & Atri,M. Face Recognition Systems: A Survey. *Sensors*, 2020. 20(2). 342. DOI: 10.3390/s20020342

13. Li S.Z., & Jain A.K. (Eds.). 2021. Handbook of Face Recognition (2nd ed.). Springer.

14. Liu W., Wen Y., Yu Z., Li M., Raj B. & Song L. *Sphereface: Deep hypersphere embedding for face recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2024. pp. 212-220.

15. Lowe D.G. *Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision*, 60(2), 2020, 91-110. DOI: 10.1023/B:VISI.0000029664.99615.94

16. Nemavhola F. 2025. A Scoping Review of Literature on Deep Learning Techniques for Face Recognition. Human Behavior and Emerging Technologies, DOI: 10.1155/hbe2/5979728

17. Revealed: 10 Secrets Behind High-Performing Facial Recognition Systems. URL: https://blog.hidglobal.com/revealed-10-secrets-behind-high-performing-facial-recognition-systems (дата звернення: 02.06.2025)

18. Parkhi O.M., Vedaldi A. & Zisserman A. 2020. Deep Face Recognition. In Proceedings of the British Machine Vision Conference (BMVC).

19. Sensors and Sensibility: USC Researchers Develop State-of-the-Art Biometric Security Systems. URL: https://viterbischool.usc.edu/news/2020/06/sensors-and-sensibility-usc-researchers-develop-state-of-the-art-biometric-security-systems/ (дата звернення: 02.06.2025)

20. Schroff F., Kalenichenko D. & Philbin J. *FaceNet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* 2020. pp. 815-823.

21. Simonyan K. & Zisserman A. 2021. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

22. Singh S. & Prasad S.*V.A.V. Techniques and Challenges of Face Recognition: A Critical Review. Procedia Computer Science,* 143, 2020. 427-434. DOI: 10.1016/j.procs.2018.10.427

23. Benefits of Facial Recognition: Pros, Cons, & Ethical Use. URL: https://amgtime.com/blog/ethical-facial-recognition-system/ (дата звернення: 02.06.2025)

24. Szegedy C., Ioffe S., Vanhoucke V. & Alemi A.A. *Inception-v4, inception-ResNet and the impact of residual connections on learning. In Proceedings of the AAAI Conference on Artificial Intelligence* 2017. pp. 4278-4284.

25. How Facial Recognition Cameras for Home Security Work? URL: https://wuuklabs.com/blogs/home-security-camera/home-security-facial-recognition. (дата звернення: 02.06.2025)

26. Taigman Y., Yang M., Ranzato, M. & Wolf L.. DeepFace: Closing the gap to human-level performance in face verification. In Proceedings of the IEEE *Conference on Computer Vision and Pattern Recognition* 2021. pp. 1701-1708.

27. Viola P. & Jones M.J. Robust real-time face detection. *International Journal of Computer Vision*, 57(2), 2024. 137-154. DOI: 10.1023/B:VISI.0000013087.49260.fb

28. Optimizing Facial Recognition Performance in Video Surveillance: A Comprehensive Guide. ULR: https://www.lumana.ai/blog/optimizing-facial-recognition-performance-in-video-surveillance-a-comprehensive-guide (дата звернення: 02.06.2025)

29. AI Home Security: Leveraging AI for Ultimate Protection. URL: https://reolink.com/blog/ai-home-security/ (дата звернення: 02.06.2025)

30. Raspberry Pi 4. URL: https://evo.net.ua/mikrokomputer-raspberry-pi-4-model-b-4gb/ (дата звернення: 02.06.2025)

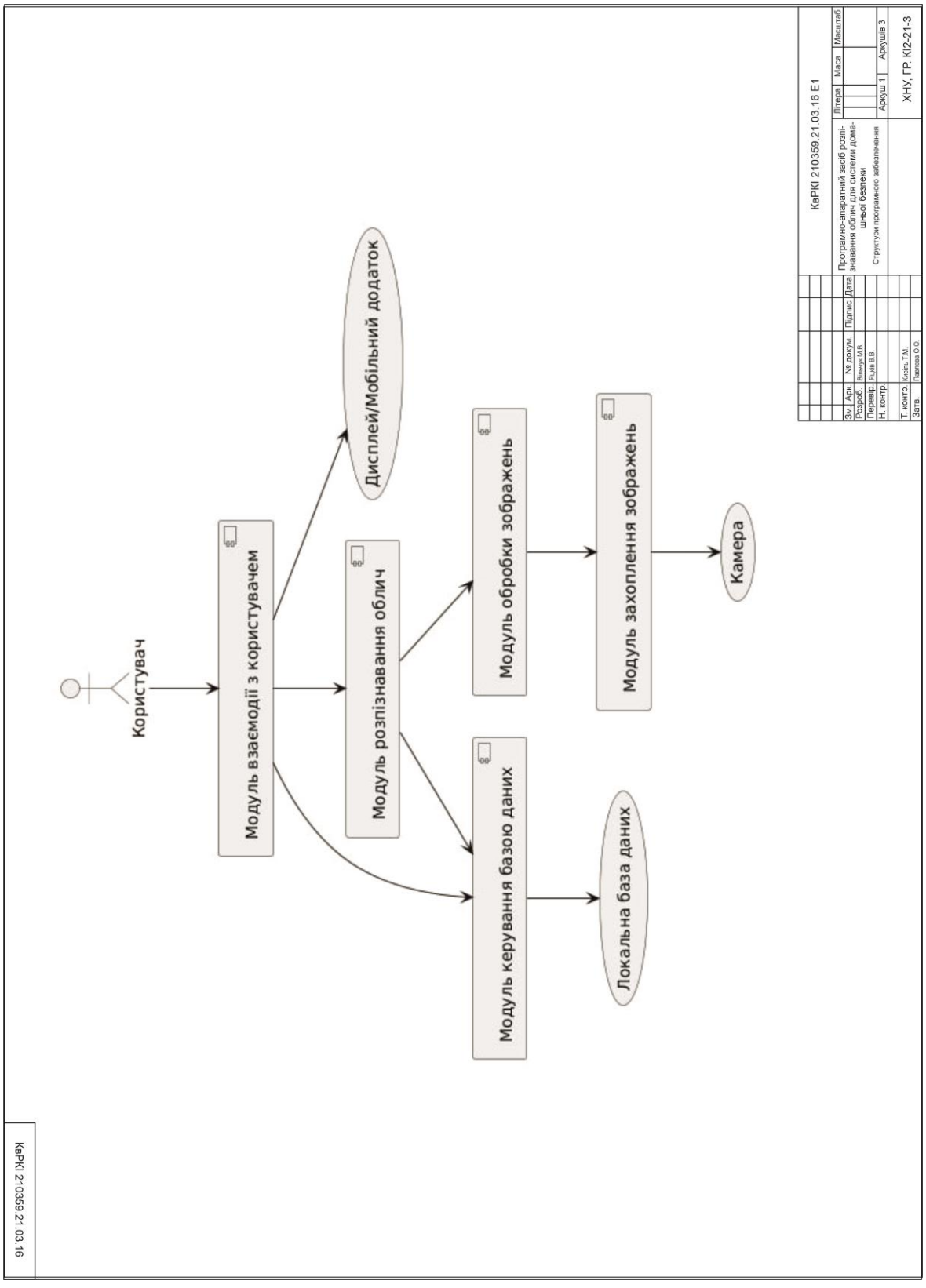
31. Мікроконтролер Esp32-CAM. URL: https://darstar.com.ua/ua/product/8450445483/ (дата звернення: 02.06.2025)

32. Raspberry Pi 4 з Raspberry Pi Camera. URL: https://evo.net.ua/raspberry-pi-camera-board/ (дата звернення: 02.06.2025)

Додаток А

(обов’язковий)

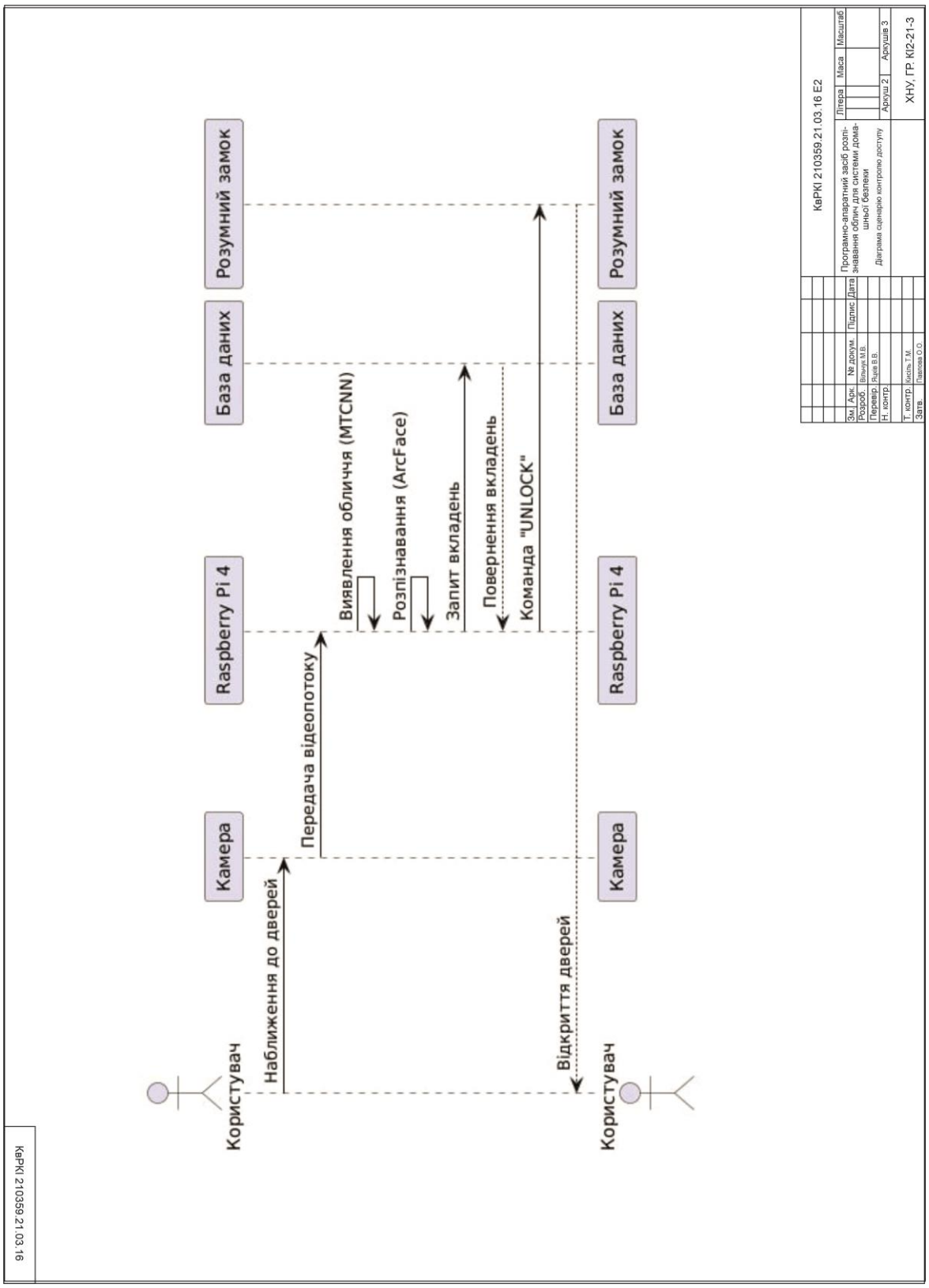
КОПІЯ КРЕСЛЕННЯ «СТРУКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»



Додаток Б

(обов’язковий)

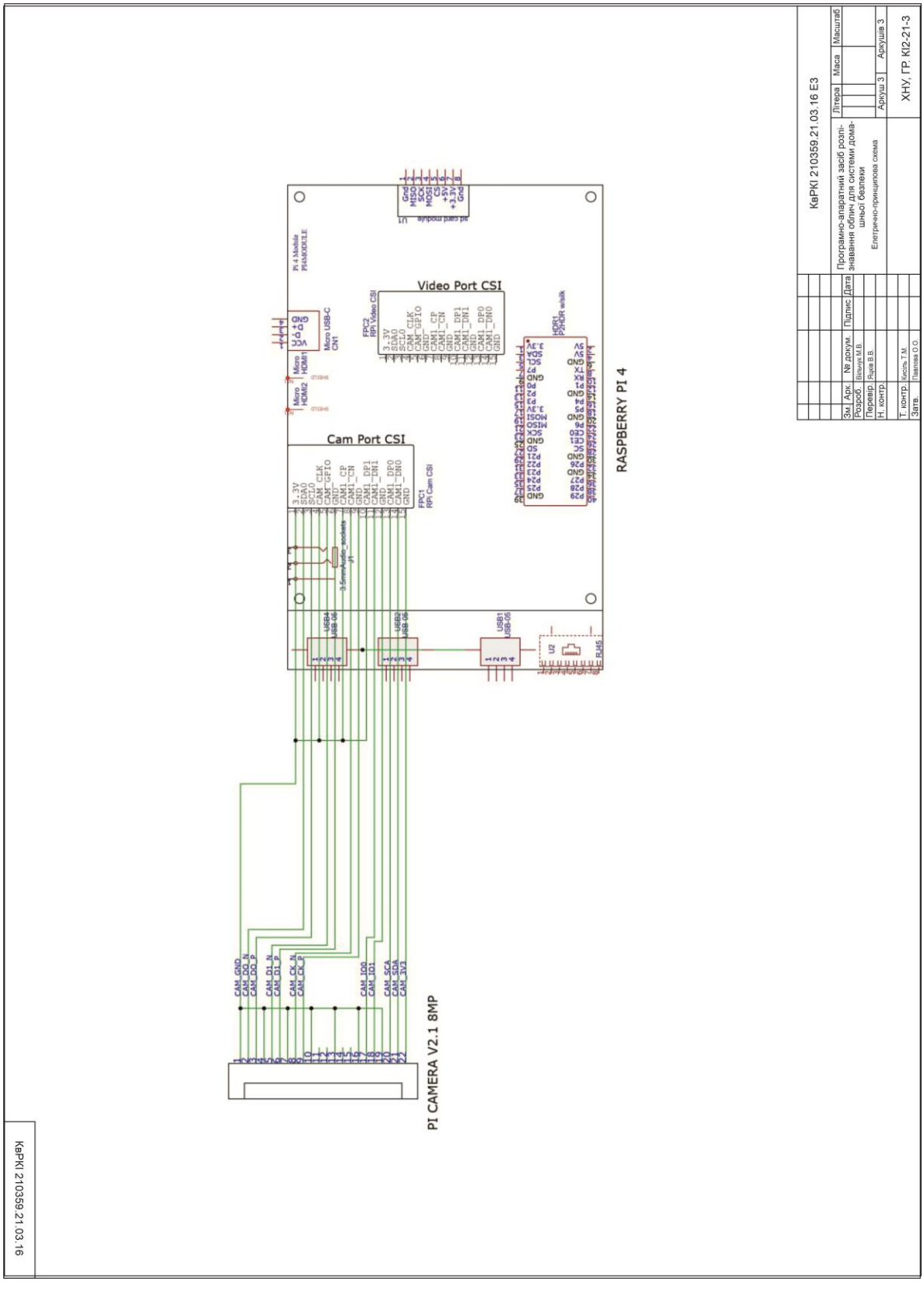
КОПІЯ КРЕСЛЕННЯ «ДІАГРАМА СЦЕНАРІЮ КОНТРОЛЮ ДОСТУПУ»



Додаток В

(обов’язковий)

КОПІЯ КРЕСЛЕННЯ «ЕЛЕТРИЧНО-ПРИНЦИПОВА СХЕМА»



Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Микита ВІЛЬЧУК

Співавтор:

Назва: Вільчук_Програмно-апаратний засіб розпізнавання облич для системи домашньої безпеки

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1:4.4%

Коефіцієнт подібності 2:1.4%

Мікропробіли: 7

Заміна букв: 0

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-06-18 14:49:40.0

Після аналізу Звіту подібності констатую наступне:

☒ **Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.**

☐ **Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.**

☐ **Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.**

Обґрунтування:

2025-06-18 Доцент Андрій Нічепорук

Дата експерт

Протокол аналізу звіту подібності експертом

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

Автор: Микита ВІЛЬЧУК

Співавтор:

Назва: Вільчук_Програмно-апаратний засіб розпізнавання облич для системи домашньої безпеки

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1:4.4%

Коефіцієнт подібності 2:1.4%

Мікропробіли: 7

Заміна букв: 0

Інтервали: 0

Білі знаки: 0

Дата створення звіту: 2025-06-18 14:49:40.0

Після аналізу Звіту подібності констатую наступне:

☒ **Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.**

☐ **Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.**

☐ **Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.**

Обгрунтування:

2025-06-18

Дата

Доцент Андрій Нічепорук

експерт

Wed Jun 18 13:32:34 EEST 2025, Медзатий Дмитро Миколайович, Хмельницький національний університет,

Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document 1.0%

Dictionaries check: en_US, ru_RU, ua_UA. **Errors in the documents: 10%**

	Document		Sum coincidence on the DB	
ID: 246704 Title: БКР Програмно-апаратний засіб розпізнавання облич для системи домашньої безпеки Added in a DB: 2025-06-18 Authors: Микита ВЛЬЧУК Heads: Василь ЯЦКІВ Consultants: Opponents:	Symbols	Lexemes	Symbols	Lexemes
	93180	1401	2764 (3%)	38 (3%)

Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Вільчук Микита Валерійович

Тема: Програмно-апаратний засіб розпізнавання облич для системи домашньої безпеки

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг кваліфікаційної роботи:

Кількість листів креслень ___3___ Кількість сторінок записки ___60___

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи є розробка Програмно-апаратного засобу розпізнавання облич для системи домашньої безпеки

2. Висновок про відповідність роботи дипломному завданню: Робота повністю відповідає поставленому завданню.

3. У ході виконання кваліфікаційної роботи було розроблено програмно-апаратний засіб для розпізнавання облич у контексті системи домашньої безпеки. Проведено повноцінний цикл дослідження – від теоретичного аналізу існуючих технологій до практичного впровадження та тестування прототипу системи на базі Raspberry Pi з камерним модулем.

У першому розділі проаналізовано сучасні методи, апаратні та програмні засоби розпізнавання облич. Проведено порівняння технологій за критеріями точності, швидкодії, стійкості до зовнішніх факторів та можливості реалізації на бюджетних апаратних платформах. Встановлено, що гібридні методи на базі алгоритмів глибокого навчання (MTCNN, ArcFace) є найефективнішими для реалізації систем домашньої безпеки завдяки високій точності та адаптивності до змінних умов.

У другому розділі виконано вибір апаратної платформи, розроблено структуру програмного забезпечення, реалізовано алгоритми розпізнавання та проведено налаштування системи. Платформа Raspberry Pi була обрана як оптимальна за критеріями доступності, продуктивності та підтримки програмних бібліотек

(OpenCV, TensorFlow Lite). Алгоритми було адаптовано до обмежених ресурсів платформи без істотної втрати точності розпізнавання.

У третьому розділі описано процес впровадження системи в умовах, наближених до реальних, проведено оцінку ефективності за критеріями точності, швидкодії та стійкості до зовнішніх впливів.

Таким чином, у результаті теоретичних і практичних досліджень досягнуто поставленої мети: створено програмно-апаратний засіб розпізнавання облич, придатний для використання в системах домашньої безпеки. Запропоноване рішення поєднує доступність, адаптивність і функціональність, а також має потенціал для подальшого вдосконалення й комерційного застосування.

4. Позитивні сторони роботи: висока практична цінність роботи.

5. Негативні сторони роботи: недостатньо глибокий аналіз фільтрації та обробки біосигналів.

6. Оцінка графічного оформлення та пояснювальної записки роботи: Пояснювальна записка оформлена коректно, згідно діючих стандартів оформлення документації.

7. Відгук про роботу в цілому: Робота виконана на належному технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: задовільно

Рецензент (прізвище, ім'я, по батькові, посада, місце роботи) _____

Мартинюк Валерій Володимирович, д.т.н., професор, зав. каф. АКІТнР

"19" травня 2025 р. _____(підпис)

Завідувачу кафедри КІІС
д-р. філософії, доц. Ользі ПАВЛОВІЙ

Микити ВІЛЬЧУКА
ПІБ здобувача вищої освіти

ФІТ, 4 курсу, групи КІ2-21-3

ЗАЯВА

З правилами чинного Положення «Про систему забезпечення академічної доброчесності у Хмельницькому національному університеті» від 01.07.2022, згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування заходів дисциплінарної та академічної відповідальності, ознайомлений (а). Про використання програмно-технічних засобів для перевірки кваліфікаційних робіт здобувачів вищої освіти на плагіат оповіщений(а) та надаю свою згоду на обробку та збереження університетом моєї роботи в інституційному репозитарії університету.

Також надаю університету право на передачу моєї роботи для обробки та збереження в базах даних програмно-технічних засобів (Strike-Plagiarism та Anti-Plagiarism) та використання роботи для виявлення плагіату в інших роботах, які перевіряються програмно-технічними засобами та користувачами, що мають доступ до цих програмно-технічних засобів, виключно в обмежених цілях для виявлення плагіату в текстах робіт.

Робота для перевірки університетом надається в друкованому та електронному варіанті. Електронна версія моєї роботи збігається (ідентична) з друкованою.

18.06 2025 року

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи Програмно-апаратний засіб розпізнавання облич для системи домашньої безпеки

Автор ВІЛЬЧУК Микита

Освітня програма Комп'ютерна інженерія та програмування

Рівень вищої освіти перший (бакалаврський) рівень

Спеціальність 123– Комп'ютерна інженерія

Науковий керівник: Василь ЯЦКІВ, д.т.н., професор.

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмним засобом(ами) комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	Відповідає
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	Не виявлено

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:

1) запозичення розміщені в розділах аналізу існуючих аналогів та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи;
2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
3) запозичення, знайдені системою Anti-Plagiarism, складають звіт з передипломної практики Вільчука Микити.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості StrikePlagiarism, складає 4.4%, та системою Anti-Plagiarism складає 1.0%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи Василь ЯЦКІВ

Гарант ОП Андрій НІЧЕПОРУК

Завідувач кафедри КІІС Ольга ПАВЛОВА