

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки

БАКАЛАВРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Визивний пристрій з розпізнаванням обличчя»

08-54.БКР.037.00.000 ПЗ

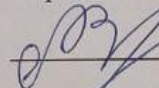
Виконав: студент 4 курсу, групи 2СП-216

спеціальності 123 — «Комп'ютерна інженерія»

освітня програма — «Системне програмування»

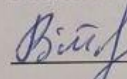
Шпикуляк А. В. 

Керівник к.т.н., доц. каф. ОТ



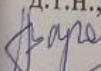
Крупельницький Л. В.

Рецензент к.т.н., доц. каф. ПЗ



Войтко В. В.

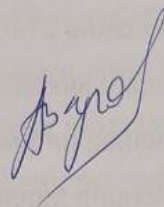
Допущено до захисту
Завідувач кафедри ОТ
д.т.н., проф. Азаров О. Д.



«1» _____ 2025 р.

Вінниця ВНТУ — 2025 рік

Вінницький національний технічний університет
Факультет інформаційних технологій та комп'ютерної інженерії
Кафедра обчислювальної техніки
Освітньо-кваліфікаційний рівень бакалавр
Галузь знань — 12 «Інформаційні технології»
Спеціальність — 123 «Комп'ютерна інженерія»
Освітня програма «Системне програмування»



ЗАТВЕРДЖУЮ
Завідувач кафедри ОТ
д.т.н., проф. Азаров О. Д.
«21» березня 2025 року

ЗАВДАННЯ НА БАКАЛАВРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

Шпикуляку Андрію Віталійовичу

1. Тема роботи: «Визивний пристрій з розпізнаванням обличчя», керівник роботи Крупельницький Леонід Віталійович, к.т.н., доцент кафедри ОТ, затверджено наказом вищого навчального закладу від «20» березня 2025 року №97.
2. Строк подання студентом роботи 13.05.2025
3. Вихідні дані до роботи: технічний опис пристрою, технології реалізації: мова програмування C++ для Raspberry Pi, бібліотека OpenCV, апаратна платформа — Raspberry Pi 4 Model B, середовище розробки — Visual Studio Code, середовище розробки схеми та її моделювання — Proteus.
4. Зміст розрахунково-пояснювальної записки: вступ; аналіз існуючих пристроїв розпізнавання обличчя; обґрунтування вибору апаратної і програмної платформи; побудова функціональної структури пристрою; розробка алгоритму роботи; реалізація програмної частини; тестування пристрою; висновки.
5. Графічні матеріали, які ілюструють технічну та логічну структуру пристрою. Зокрема, представлено електричну схему підключення основних компонентів. Наведено блок-схему алгоритму. Скріншот роботи програми.

6. Консультанти розділів роботи приведені в таблиці 1.

Таблиця 1 — Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1 – 4	к.т.н., доц. каф. ОТ Крупельницький Л.В.	21.03.2025	13.05.2025
Нормоконтроль	ас. каф. ОТ Швець С.І.	13.05.2025	30.05.2025

7. Дата видачі завдання 21.03.2025 р.

8. Календарний план виконання БКР наведений у таблиці 2.

Таблиця 2 — Календарний план

№ з/п	Назва етапів дипломної роботи	Строк виконання	Підпис
1	Постановка задачі роботи	21.03.2025	В.м.
2	Аналіз предметної області, огляд існуючих рішень та збір літературних джерел	21.03.2025— 28.03.2025	В.м.
3	Обґрунтування вибору апаратного і програмного забезпечення	29.03.2025— 4.04.2025	В.м.
4	Розробка структурної, функціональної та алгоритмічної схем	5.04.2025— 13.04.2025	В.м.
5	Розробка й тестування програмної реалізації та моделювання роботи схеми	14.04.2025— 27.04.2025	В.м.
7	Оформлення пояснювальної записки	28.04.2025— 11.05.2025	В.м.
8	Перевірка якості виконання бакалаврської кваліфікаційної роботи та усунення недоліків	13.05.2025	В.м.

Студент Андрій ШПИКУЛЯК

Керівник роботи Леонід КРУПЕЛЬНИЦЬКИЙ
к.т.н., доц.

АНОТАЦІЯ

УДК 004.9

Шпикуляк А.В. Визивний пристрій з розпізнаванням обличчя. Бакалаврська кваліфікаційна робота зі спеціальності 123 — Комп'ютерна інженерія, освітня програма — Системне програмування. Вінниця: ВНТУ, 2025. 80 с.

На укр. мові. Бібліогр.: 20 назв; рис.: 23; табл. 3

У бакалаврській кваліфікаційній роботі спроектовано визивний пристрій з розпізнаванням обличчя на базі одноплатного комп'ютера Raspberry Pi. Пристрій використовує методи комп'ютерного зору та бібліотеку OpenCV для виявлення та ідентифікації користувачів. У межах роботи здійснено аналіз апаратного і програмного забезпечення, побудовано структурну схему, реалізовано алгоритм розпізнавання та проведено базові випробування пристрою в режимі моделювання.

Ключові слова: Raspberry Pi, комп'ютерний зір, розпізнавання обличчя, визивний пристрій, OpenCV, C++.

ABSTRACT

Shpykuliak A.V. Face Recognition-Based Intercom Device. Bachelor's qualification thesis in specialty 123 – Computer Engineering, educational program – System Programming. Vinnytsia: VNTU, 2025. 80 p.

In Ukrainian. Bibliography: 20 titles; Fig.:23; table 3

The bachelor's qualification thesis presents the design of a face recognition-based intercom device built on a Raspberry Pi single-board computer. The device utilizes computer vision methods and the OpenCV library to detect and identify users. The work includes analysis of hardware and software components, development of a structural scheme, implementation of the recognition algorithm, and basic testing of the device in simulation mode.

Keywords: Raspberry Pi, computer vision, face recognition, intercom device, OpenCV, C++.

ЗМІСТ

ВСТУП	4
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	6
1.1 Огляд сфери застосування систем розпізнавання обличчя	6
1.2 Огляд та класифікація існуючих рішень	7
1.3 Аналіз аналогів та обґрунтування вибраного підходу	11
1.4 Ціль розробки та формування технічних вимог	16
1.5 Сучасні виклики та етичні аспекти використання технології розпізнавання обличчя	17
2 ТЕОРЕТИЧНЕ ОБҐРУНТУВАННЯ МЕТОДІВ РЕАЛІЗАЦІЇ ВИЗИВНОГО ПРИСТРОЮ З РОЗПІЗНАВАННЯМ ОБЛИЧЧЯ	20
2.1 Основні принципи та архітектура систем розпізнавання обличчя	20
2.2 Структурна схема, алгоритм та функціональний опис пристрою	22
2.2.1 Структурна схема та функціональний опис	22
2.2.2 Загальний алгоритм роботи пристрою	25
2.3 Розрахунок переваги обраного рішення за критерієм енергоефективності	26
2.4 Варіативний аналіз і вибір базових інструментів для технічного рішення	28
2.4.1 Аналіз і вибір апаратного забезпечення	29
2.4.2 Аналіз і вибір програмного забезпечення	34
3 РЕАЛІЗАЦІЯ ВИЗИВНОГО ПРИСТРОЮ З РОЗПІЗНАВАННЯМ ОБЛИЧЧЯ	37
3.1 Побудова схеми підключення компонентів	37
3.2 Реалізація програмної логіки пристрою	38
3.3 Можливе розширення функціональності пристрою за допомогою додаткових датчиків	41
3.4 Огляд альтернативних моделей Raspberry Pi	46
4 ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ	51
4.1 Моделювання реальних сценаріїв використання	51
4.2 Методика та умови тестування	53

4.3 Результати тестування та їх аналіз	55
4.4 Порівняння методів аутентифікації та рекомендації щодо подальших досліджень та коригувань розробки.....	58
ВИСНОВКИ	63
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	65
ДОДАТОК А Технічне завдання.....	67
ДОДАТОК Б Протокол перевірки кваліфікаційної роботи на наявність текстових запозичень	71
ДОДАТОК В Графічна частина	72
ДОДАТОК Г Ілюстративна частина	73
ДОДАТОК Д Лістинг програмного забезпечення.....	74

ВСТУП

У сучасних умовах питання забезпечення безпеки житла, офісів, підприємств і об'єктів закритого типу набувають особливої актуальності. Одним із напрямів захисту від несанкціонованого проникнення є впровадження систем біометричної ідентифікації. Серед них розпізнавання обличчя є одним із найзручніших і найменш інвазивних методів, оскільки не потребує фізичного контакту з пристроєм та може працювати в реальному часі.

Системи виклику, які використовуються в домофонах, системах доступу до приміщень або на входах до територій, часто мають обмежені функціональні можливості та застарілі технології. Розробка визивного пристрою з розпізнаванням обличчя дозволяє значно підвищити рівень безпеки, зручність користування та автоматизацію контролю доступу. Такий підхід відкриває перспективи для впровадження в розумні будинки, офісні комплекси, навчальні заклади, лікарні, склади тощо.

Проблематика полягає в необхідності розробки недорогого, компактного, енергоефективного та надійного пристрою, здатного розпізнавати обличчя, зіставляти їх з базою даних та приймати рішення про надання або відмову у доступі. Складність виконання полягає в поєднанні комп'ютерного зору, обчислювальної оптимізації та апаратної реалізації в єдину систему з можливістю автономної роботи.

Актуальність теми обумовлена зростанням запитів на локальні системи контролю доступу з мінімальним людським втручанням, особливо у військовий та післявоєнний час, коли захист об'єктів стає пріоритетом.

Об'єктом дослідження є процес розробки та організації роботи пристрою з використанням можливостей комп'ютерного зору.

Предметом дослідження – методи реалізації розпізнавання обличчя та їх застосування в пристроях контролю доступу.

Метою роботи є підвищення безпеки доступу на об'єкт використовуючи технологію розпізнавання обличчя.

Завдання бакалаврської кваліфікаційної роботи:

- провести аналіз сучасних систем біометричного контролю доступу;
- обґрунтувати вибір апаратних та програмних засобів для реалізації пристрою;
- розробити структурну та функціональну схему пристрою;
- реалізувати алгоритм розпізнавання обличчя з використанням бібліотек комп'ютерного зору;
- провести тестування працездатності пристрою.

Апробація матеріалів проєкту виконано в доповіді LIV Всеукраїнська науково-технічна конференція підрозділів Вінницького національного технічного університету (2025).

Публікація за темою проєкту [1]: Шпикуляк А.В., Крупельницький Л.В. Визивний пристрій з розпізнаванням обличчя // LIV Всеукраїнська науково-технічна конференція підрозділів Вінницького національного технічного університету (2025). – Режим доступу: <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki-2025/paper/view/24365>

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Огляд сфери застосування систем розпізнавання обличчя

Розпізнавання обличчя є однією з найважливіших і найперспективніших технологій у сфері біометричної ідентифікації. Ця технологія базується на здатності інформаційної системи виявляти та розпізнавати обличчя людини зображеного на фото чи у відеопотоці з подальшою ідентифікацією або верифікацією особи. Завдяки прогресу в галузі комп'ютерного зору, штучного інтелекту та глибинного навчання, системи розпізнавання обличчя вже не обмежуються експериментальними або вузько спеціалізованими сферами, а стали повноцінною частиною повсякденного життя, бізнесу, безпеки, медицини та освіти [2].

Історія розвитку технологій розпізнавання обличчя починається з 1960-х років, коли американський науковець Вуді Бледсо вперше почав дослідження в напрямі автоматизованої ідентифікації людей за фотографіями. Втім, реального впровадження технологія набула лише у 1990–2000-х роках, зокрема після розробки алгоритму Віюлі–Джонса (2001), який дозволив ефективно виявляти обличчя на зображеннях у реальному часі. Справжній прорив стався у 2010-х роках з розвитком глибоких нейронних мереж, які значно підвищили точність, стійкість і швидкодію розпізнавання, зробивши можливим використання технології на мобільних пристроях, хмарних сервісах, системах відеонагляду [3].

На сьогодні пристрої розпізнавання обличчя застосовуються у широкому спектрі сфер. Найпоширенішим є використання в системах безпеки для контролю доступу до будівель, приміщень або електронних пристроїв. Інфраструктура громадського простору (аеропорти, вокзали, станції метро) дедалі частіше оснащується камерами з функцією ідентифікації обличчя з метою виявлення розшукуваних осіб, попередження терористичних загроз та автоматизації проходження контролю. У споживчому секторі розпізнавання обличчя реалізується в смартфонах і ноутбуках як основний або додатковий метод біометричної аутентифікації, що забезпечує безконтактний, швидкий та доволі

зручний доступ до пристрою [4].

Інтенсивно технологія впроваджується у фінансову сферу, зокрема для дистанційного відкриття рахунків, підтвердження особи під час онлайн-транзакцій та захисту від шахрайства. У роздрібній торгівлі та маркетингу системи розпізнавання використовуються для аналізу поведінки покупців, автоматичного зчитування емоцій, ідентифікації постійних клієнтів і персоналізації рекламного контенту. Медична галузь також виявила великий потенціал у впровадженні цієї технології: тут вона дозволяє ідентифікувати пацієнтів, оптимізувати реєстрацію, слідкувати за дотриманням режиму лікування, а також допомагає в роботі з людьми з порушеннями пам'яті.

Особливу актуальність розпізнавання обличчя здобуло в умовах пандемії, коли постала потреба у безконтактній ідентифікації. Такі системи є безпечними з точки зору гігієни, що є важливою перевагою перед традиційними методами, які вимагають фізичного контакту. Крім того, у системах дистанційного навчання та онлайн-екзаменування пристрої розпізнавання обличчя застосовуються для підтвердження присутності студентів та запобігання академічній нечесності.

З огляду на вищенаведене, можна зробити висновок, що пристрої розпізнавання обличчя є технологічно зрілими рішеннями, що вже зараз відіграють ключову роль у багатьох аспектах життя суспільства. Подальший розвиток у цій галузі зумовлений потребою у ще більшій точності, швидкості, захищеності персональних даних та розширенні етичних стандартів їх використання. Актуальність наукових досліджень та інженерних рішень у сфері пристроїв розпізнавання обличчя не викликає сумнівів — ця технологія вже сьогодні формує основи майбутнього цифрового суспільства [5].

1.2 Огляд та класифікація існуючих рішень

З розвитком цифрових технологій, штучного інтелекту та інтернету речей (IoT) на ринку з'явилася велика кількість рішень для організації доступу, контролю відвідуваності, охорони об'єктів та підвищення безпеки. До найбільш поширених технічних рішень, які поєднують функції виявлення, ідентифікації та

керування доступом, належать відеодомофони, біометричні системи та інтелектуальні IoT-рішення. Кожен із цих класів систем має власні особливості реалізації, функціональні можливості, а також переваги та недоліки з точки зору вартості, безпеки, точності та масштабованості [6].

Відеодомофони є традиційним рішенням у сфері контролю доступу до житлових і офісних приміщень. Приклад пристрою зображено на рисунку 1.1.



Рисунок 1.1. — Відеодомофон

Сучасні відеодомофони забезпечують двосторонній аудіо та відеозв'язок між відвідувачем і користувачем, що дозволяє дистанційно приймати рішення про надання або відмову у доступі. Залежно від функціональності, деякі моделі підтримують зберігання відео, роботу з додатками на смартфоні, підключення до Wi-Fi або Ethernet, інтеграцію з охоронними системами. Основними перевагами відеодомофонів є простота інсталяції, зрозумілий інтерфейс для кінцевого користувача та широке поширення на ринку. Водночас такі пристрої часто не мають елементів штучного інтелекту чи біометричної аутентифікації, що обмежує їхню ефективність у ситуаціях, коли важлива точна ідентифікація особи.

Біометричні системи — це окремий клас рішень, що базуються на фізіологічних або поведінкових характеристиках людини: відбитках пальців, сітківці ока, формі обличчя, голосі або динаміці ходи. Приклад такої системи зображено на рисунку 1.2.



Рисунок 1.2 — Біометрична система контролю доступу

У межах цієї роботи основну увагу приділено саме системам розпізнавання обличчя як найзручнішому та безконтактному способу аутентифікації. Біометричні системи мають низку суттєвих переваг: висока точність, низька ймовірність підробки, можливість безперервної ідентифікації, відсутність потреби у фізичних носіях (картках, ключах). Однак їх впровадження вимагає наявності обчислювальних ресурсів для обробки зображень, спеціалізованого програмного забезпечення та гарантування конфіденційності даних. Крім того, біометричні рішення можуть викликати етичні та правові суперечки, особливо при масовому використанні в публічному просторі.

Зі стрімким розвитком технологій IoT (Internet of Things) з'явився цілий сегмент інтелектуальних пристроїв, що поєднують функції відеоспостереження, біометрії, обміну даними через мережу, аналітики та віддаленого управління. IoT-

рішення, зокрема смарт-дзвінки з розпізнаванням обличчя, інтелектуальні замки, автоматизовані шлюзи, використовують переваги хмарних обчислень, мобільних додатків і машинного навчання. Такий підхід дозволяє реалізовувати сценарії адаптивного доступу, реєстрації відвідувачів у реальному часі, надсилання сповіщень, інтеграцію з іншими «розумними» системами будинку [7]. Одним із прикладів такої системи наведено на рисунку 1.3.



Рисунок 1.3 — Розумний дверний замок для систем розумного дому

Перевагами є масштабованість, гнучкість і можливість централізованого керування. Водночас до недоліків можна віднести підвищені вимоги до захисту мережевого з'єднання, ризики кібербезпеки, залежність від стабільності інтернет-з'єднання, а також необхідність регулярного оновлення програмного забезпечення.

Огляд існуючих комерційних та open-source рішень демонструє розмаїття підходів до реалізації таких систем. Наприклад, відеодомофони Hikvision, Dahua або Commax з базовою IP-функціональністю мають обмежені можливості

персоналізації. Біометричні комплекси від компаній ZKTeco, Suprema або VisionLabs демонструють вищу точність, але потребують значних витрат на впровадження. У секторі DIY-рішень популярності набули проекти на базі Raspberry Pi, що дозволяють розробляти компактні, недорогі та гнучкі пристрої з можливістю розпізнавання обличчя завдяки бібліотекам для комп'ютерного зору. Такі рішення особливо привабливі для навчальних цілей, прототипування та малобюджетних впроваджень, оскільки дозволяють балансувати між функціональністю та вартістю.

Таким чином, існуючі технічні рішення у сфері ідентифікації осіб за обличчям можуть бути класифіковані за рівнем інтелекту, ступенем автономності, типом біометрії, способом передачі даних та архітектурою системи. Кожен підхід має свої сфери доцільного використання: від індивідуального житлового сектору до критичної інфраструктури.

1.3 Аналіз аналогів та обґрунтування вибраного підходу

Системи розпізнавання обличчя активно застосовуються у пристроях, таких як відеодомофони, системи контролю доступу та розумні IoT-рішення. Для вибору оптимального технічного рішення необхідно проаналізувати сучасні аналоги, їхні характеристики, переваги та недоліки. У цьому підрозділі розглянуто основні підходи до реалізації систем розпізнавання обличчя, проведено порівняння за ключовими параметрами та обґрунтовано вибір платформи Raspberry Pi з використанням бібліотеки OpenCV для розробки пристрою [8].

Сучасні системи розпізнавання обличчя для викличних пристроїв можна класифікувати за типом апаратної платформи, програмного забезпечення та функціональних можливостей. Основні аналоги включають комерційні рішення (наприклад, від Hikvision, ZKTeco), опенсорс-системи на базі одноплатних комп'ютерів (Raspberry Pi, NVIDIA Jetson Nano) та спеціалізовані біометричні системи. Для аналізу обрано чотири типові рішення, які відображають різноманітність підходів.

Hikvision DS-KB8113-IME1 – комерційний відеодомофон із вбудованою системою розпізнавання обличчя, що використовує власні алгоритми та хмарну обробку даних. Зображено на рисунку 1.4.



Рисунок 1.4 — Відеопанель Hikvision DS-KB8113-IME1(B) UKR IP65

ZKTeco Visible Light – біометрична система розпізнавання обличчя з функцією антиспуфінгу та підтримкою самонавчання. Зображено на рисунку 1.5.



Рисунок 1.5 — Система розпізнавання обличчя Visible Light

Raspberry Pi з OpenCV і Face Recognition Library – openсорс-рішення на базі одноплатного комп'ютера Raspberry Pi, яке використовує бібліотеки OpenCV і dlib для локальної обробки зображень. Зображено на рисунку 1.6.

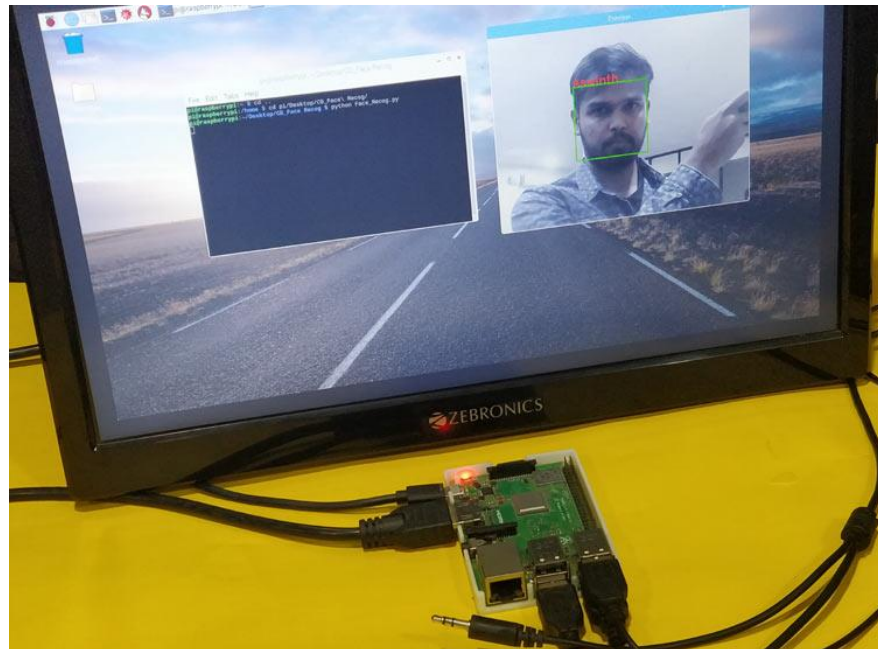


Рисунок 1.6 — Приклад типового DIY проєкту

NVIDIA Jetson Nano з TensorFlow – високопродуктивна платформа для розпізнавання обличчя з використанням нейронних мереж і бібліотеки TensorFlow. Приклад такого пристрою зображено на рисунку 1.7.

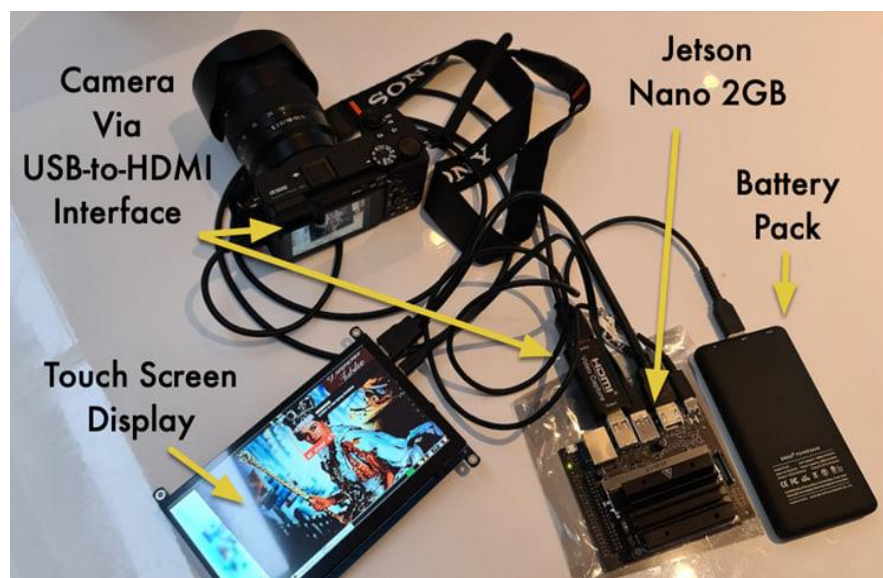


Рисунок 1.7 — Приклад реалізації пристрою

Для оцінки аналогів використано такі критерії [9]:

- продуктивність апаратної платформи: обчислювальна потужність, наявність GPU, обсяг пам'яті;
- якість програмного забезпечення: надійність, підтримка оновлень, доступність документації;
- точність розпізнавання: здатність коректно ідентифікувати обличчя за різних умов;
- швидкість обробки: час, необхідний для детекції та ідентифікації;
- вартість: доступність апаратного та програмного забезпечення;
- гнучкість: можливість модифікації, інтеграція з іншими системами;
- енергоефективність: споживана потужність.

Кожен параметр оцінено за шкалою від 1 (найгірший показник) до 5 (найкращий показник). Результати порівняння наведено в таблиці 1.1.

Таблиця 1.1 — Порівняння різних пристроїв з розпізнаванням обличчя

Параметр	Hikvision DS- KB8113- IME1	ZKTeco Visible Light	Raspberry Pi + OpenCV	NVIDIA Jetson Nano + TF
Продуктивність платформи	3	4	3	5
Якість ПЗ	4	4	4	4
Точність	3	5	4	5
Швидкість обробки	5	4	4	5
Вартість	3	1	5	2
Гнучкість	1	2	5	5
Енергоефективність	3	3	5	2
Середній бал	3.14	3.28	4.28	4.14

Аналіз результатів порівняння пристроїв:

— пристрій Hikvision DS-KB8113-IME1 отримав високі бали за швидкість обробки (5) і якість програмного забезпечення (4), але низькі за гнучкість (1) через закриту архітектуру та вартість (3) через високу ціну. Середній бал (3.14) вказує на обмежену придатність для дослідницьких проєктів;

— система ZKTeco Visible Light демонструє дуже високу точність розпізнавання (5), але низьку оцінку за вартість (1) через високу ціну та обмежену гнучкість (2). Середній бал (3.28) відображає спеціалізацію системи для комерційних біометричних застосувань;

— пристрій Raspberry Pi з OpenCV отримав найвищі бали за вартість (5), гнучкість (5) і енергоефективність (5), але нижчі за швидкість обробки (4) через обмежену обчислювальну потужність. Середній бал (4.28) підкреслює оптимальність для бюджетних і гнучких рішень в рамках дослідницьких проєктів для побутового використання [10];

— система NVIDIA Jetson Nano з TensorFlow має найвищі оцінки за продуктивність апаратної платформи (5), точність (5) і швидкість (5), але нижчі за енергоефективність (2) через високе споживання енергії. Середній бал (4.14) вказує на високу продуктивність, але меншу придатність для автономних систем.

Подальший вибір платформи Raspberry Pi з використанням бібліотеки OpenCV для розробки визивного пристрою з розпізнаванням обличчя обґрунтовано такими факторами:

— висока економічна ефективність (5 балів): Raspberry Pi є однією з найдоступніших платформ, а OpenCV – безкоштовною опенсорс-бібліотекою. Це дозволяє мінімізувати витрати на розробку порівняно з комерційними аналогами, такими як Hikvision чи ZKTeco;

— гнучкість і масштабованість (5 балів): Відкрита архітектура Raspberry Pi та OpenCV забезпечує можливість модифікації коду, додавання нових функцій (наприклад, інтеграція з IoT) і адаптації під специфічні вимоги проєкту;

— енергоефективність (5 балів): Низьке енергоспоживання Raspberry Pi робить платформу ідеальною для автономних викличних пристроїв, на відміну від

NVIDIA Jetson Nano, яка отримала нижчу оцінку за цим параметром;

— достатня продуктивність (3–4 бали): Хоча Raspberry Pi поступається NVIDIA Jetson Nano за швидкістю обробки (3 проти 5), її можливості достатні для побутових викличних пристроїв, де затримка в 0.5–1 с є прийнятною;

— локальна обробка даних: Raspberry Pi з OpenCV забезпечує обробку даних без використання хмарних сервісів, що підвищує конфіденційність і знижує залежність від інтернет-з'єднання, на відміну від Hikvision;

— підтримка спільноти: Велика спільнота розробників Raspberry Pi та OpenCV забезпечує доступ до документації, прикладів коду та технічної підтримки, що полегшує розробку та майбутнє вдосконалення проєкту.

1.4 Ціль розробки та формування технічних вимог

Розробка визивного пристрою з розпізнаванням обличчя на базі Raspberry Pi та бібліотеки OpenCV вимагає чіткого визначення цілей і завдань проєкту, а також формулювання технічних вимог до апаратного та програмного забезпечення. У цьому підпункті сформульовано основні задачі розробки, визначено функціональні та нефункціональні вимоги до системи, а також окреслено ключові аспекти, які будуть деталізовані в технічному завданні (додаток А). Це забезпечує основу для подальшого проєктування та реалізації пристрою [11].

Ціль розробки — створення енергоефективного, економічно доступного та гнучкого викличного пристрою, який здатен ідентифікувати користувачів за допомогою розпізнавання обличчя в реальному часі. Пристрій має бути придатним для використання в побутових і офісних умовах, наприклад, для контролю доступу до будинку чи офісу. Для досягнення цієї цілі визначено декілька задач.

Розробка апаратної частини:

- налаштування одноплатного комп'ютера Raspberry Pi як основної обчислювальної платформи;
- інтеграція камери для захоплення відеопотоку;

- забезпечення стабільного живлення та компактного дизайну для автономної роботи.

Розробка програмного забезпечення:

- реалізація алгоритмів розпізнавання обличчя з використанням бібліотеки OpenCV, комбінуючи методи детекції та ідентифікації;
- створення інтерфейсу для взаємодії з користувачем;
- досягнення часу розпізнавання обличчя не більше 1 секунди в умовах нормального освітлення;
- мінімізація споживання енергії для забезпечення автономної роботи пристрою;
- проведення тестування роботи пристрою в різних умовах освітлення та кутах зйомки для забезпечення високої точності розпізнавання;
- перевірка стабільності роботи пристрою при тривалому використанні.

1.5 Сучасні виклики та етичні аспекти використання технології розпізнавання обличчя

У сучасному світі технології розпізнавання обличчя все частіше впроваджуються в різноманітні сфери діяльності: від забезпечення безпеки у громадських місцях до персоналізованої реклами та банківських сервісів. Попри стрімкий розвиток, ця технологія має низку технічних, етичних та юридичних викликів, що викликають серйозні дискусії як серед науковців, так і в суспільстві загалом.

Одна з основних проблем — це нестабільність точності розпізнавання в залежності від умов зовнішнього середовища. Наприклад, недостатнє або надмірне освітлення, зміни у зовнішності людини (окуляри, борода, маска, головний убір), різний ракурс, міміка обличчя або старіння можуть суттєво знизити ефективність роботи алгоритму.

Особливу складність становлять багатонаціональні середовища, де алгоритми, навчені на обмежених наборах даних, можуть демонструвати упередженість щодо певних етнічних груп.

Одним із ключових етичних викликів є загроза порушення приватності громадян. Системи відеоспостереження з розпізнаванням облич дозволяють непомітно і постійно ідентифікувати людей у публічному просторі, не запитуючи їхньої згоди. Це створює ризик формування суспільства тотального контролю.

Відомими стали приклади Китаю, де такі системи активно використовуються для оцінки соціального рейтингу громадян. Камери з функцією розпізнавання облич ідентифікують осіб, які порушили правила дорожнього руху, і одразу публікують їхні фото на електронних щитах у містах. Хоча ці заходи обґрунтовуються безпекою, вони створюють атмосферу страху і тиску, що суперечить демократичним принципам. Приклад зображено на рисунку 1.8.

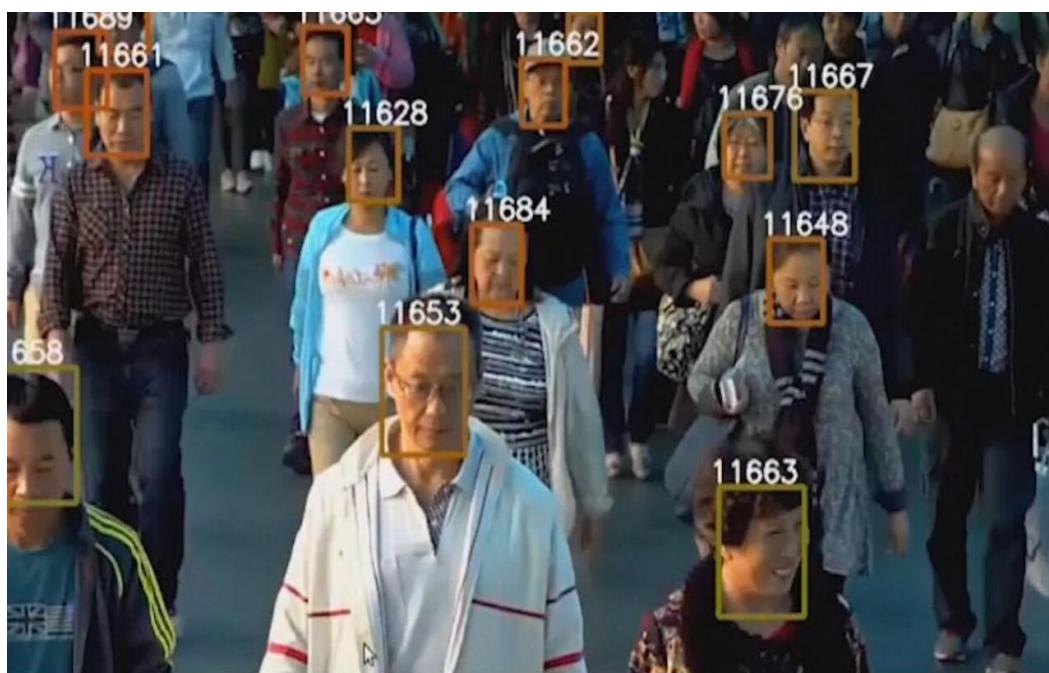


Рисунок 1.8 — Розпізнавання обличчя для соціального рейтингу

Використання технологій розпізнавання обличчя викликає багато юридичних запитань. У багатьох країнах ще не існує чіткого законодавства, що б регулювало збір, обробку та зберігання біометричних даних. Це створює правову невизначеність, яка може бути використана як державними структурами, так і приватними компаніями у власних цілях.

У Європейському Союзі діє загальний регламент захисту даних (GDPR), який вимагає інформованої згоди на обробку біометричної інформації.

Одним з опосередкованих наслідків поширення таких систем є зниження довіри громадян до держави, бізнесу та самих технологій. У громадян виникає відчуття постійного нагляду, яке може змінити їхню поведінку навіть у буденних ситуаціях. Це явище відоме як ефект Паноптикума — коли людина змінює поведінку, знаючи, що за нею можуть стежити.

Особливо чутливим є питання розпізнавання облич у навчальних закладах, лікарнях або під час протестів, де існує висока ймовірність порушення прав і свобод. Деякі активісти та правозахисні організації, зокрема ACLU (American Civil Liberties Union), виступають за повну заборону використання цієї технології в публічному просторі.

Незважаючи на високий потенціал технологій розпізнавання обличчя, який відкриває нові горизонти у сфері безпеки та автоматизації, їх широке та масове застосування супроводжується низкою складних, багатогранних проблем, які жодним чином не можна залишати поза увагою. З одного боку, ми спостерігаємо стрімкий розвиток алгоритмів машинного навчання, які забезпечують дедалі вищу точність і швидкість розпізнавання. З іншого — це породжує актуальні виклики, пов'язані з етичністю використання таких даних, дотриманням законодавства у сфері захисту персональної інформації, а також збереженням основоположних прав і свобод людини, зокрема права на приватність.

У подальшій перспективі важливо не лише продовжувати технічне вдосконалення, фокусуючись на їхній ефективності, надійності та адаптивності, але й розробляти та впроваджувати чіткі правові, етичні та інституційні рамки, які б регламентували способи та межі використання цієї технології. Потрібен підхід, який дозволить зберегти баланс між досягненнями науково-технічного прогресу та необхідністю захисту інтересів суспільства, громадян і кожної окремої людини.

2 ТЕОРЕТИЧНЕ ОБҐРУНТУВАННЯ МЕТОДІВ РЕАЛІЗАЦІЇ ВИЗИВНОГО ПРИСТРОЮ З РОЗПІЗНАВАННЯМ ОБЛИЧЧЯ

2.1 Основні принципи та архітектура систем розпізнавання обличчя

Розробка визивного пристрою з розпізнаванням обличчя на базі Raspberry Pi та бібліотеки OpenCV є прикладом інтеграції комп'ютерного зору у вбудовані системи для вирішення задач контролю доступу. Наукова задача проєкту полягає в створенні економічно доступного, енергоефективного та гнучкого рішення, яке забезпечує швидке і точне розпізнавання обличчя в побутових умовах.

Системи розпізнавання обличчя базуються на методах комп'ютерного зору та машинного навчання, які дозволяють ідентифікувати особу за унікальними рисами обличчя. Процес розпізнавання складається з кількох етапів, які формують основу функціонування викличного пристрою. Спочатку камера захоплює відеопотік або статичне зображення, яке містить обличчя. Якість зображення, зокрема роздільна здатність і рівень шуму, відіграє ключову роль у подальшій обробці. Далі виконується детекція обличчя, тобто локалізація області зображення, що містить обличчя, шляхом виявлення характерних рис, таких як очі, ніс і рот. Для цього використовуються алгоритми, такі як каскадні класифікатори Хаара (Haar Cascade), які ефективно працюють на платформах із низькою обчислювальною потужністю, або глибокі нейронні мережі (DNN), що забезпечують вищу точність [12].

Після детекції зображення нормалізується для усунення впливу змін освітлення, поворотів чи масштабування. Цей етап включає переведення зображення в градації сірого, вирівнювання контрастності та геометричну корекцію. Наступним кроком є виділення ознак, під час якого створюється цифрове представлення (дескриптор) обличчя. У сучасних системах для цього застосовуються глибокі нейронні мережі, наприклад модель ResNet-10, яка генерує 128-вимірні вектори ознак. Останній етап буде класифікація, що передбачає порівняння отриманого дескриптора з базою даних зареєстрованих користувачів за допомогою метрик схожості, таких як евклідова відстань, і

прийняття рішення про ідентифікацію [13].

Для визивного пристрою ці принципи адаптовано до обмежених ресурсів Raspberry Pi. Задача проєкту полягає в забезпеченні точності розпізнавання, часу обробки до 1 секунди та енергоспоживання не більше 7 Вт при вартості апаратного забезпечення до \$100. Вирішення цієї задачі досягається через комбінацію швидких алгоритмів детекції (Haar Cascade) і точних методів ідентифікації (DNN), реалізацію яких забезпечує бібліотека OpenCV.

Архітектура системи розпізнавання обличчя є модульною, що дозволяє адаптувати її до різних застосувань, включаючи викличні пристрої для контролю доступу. Вона складається з функціональних модулів, які взаємодіють для виконання задачі ідентифікації. Основні модулі включають захоплення даних, попередню обробку, виділення ознак, класифікацію та вивід результатів, кожен з яких виконує специфічну роль у процесі [14].

Модуль захоплення даних відповідає за отримання вхідних даних, таких як відеопотік або статичне зображення, що містить обличчя. Цей модуль є початковою точкою системи, забезпечуючи якісний вхідний матеріал для подальшої обробки. Якість даних, зокрема чіткість і рівень шуму, суттєво впливає на ефективність системи.

Модуль попередньої обробки виконує підготовку даних, включаючи детекцію обличчя та вирівнювання. На цьому етапі також можуть застосовуватися методи нормалізації, такі як корекція яскравості чи контрастності, для усунення впливу змін освітлення. Попередня обробка є критично важливою для забезпечення стабільності роботи системи в різних умовах.

Модуль виділення ознак обробляє вирівняне обличчя, створюючи унікальний дескриптор, який використовується для ідентифікації. Цей модуль є центральним у системі, оскільки від якості ознак залежить точність розпізнавання. Сучасні підходи, засновані на глибокому навчанні, дозволяють створювати дескриптори, які є стійкими до змін у виразі, позі чи частковому закритті обличчя.

Модуль класифікації порівнює вилучений дескриптор із базою даних

zareєстрованих осіб, визначаючи ідентичність або повідомляючи про невідповідність. Цей модуль може включати логіку для обробки помилок, наприклад, повторну спробу розпізнавання або запит на альтернативну ідентифікацію.

Модуль виводу забезпечує передачу результатів користувачеві або іншій системі. У контексті викличного пристрою це може бути сигнал про надання доступу, відображення імені користувача або повідомлення про невдалу ідентифікацію.

Додаткові модулі можуть включати управління базою даних для зберігання профілів користувачів, інтеграцію з зовнішніми системами (наприклад, IoT-платформами) та механізми безпеки, такі як шифрування даних. Модульна архітектура дозволяє гнучко адаптувати розробку до специфічних вимог та легко вирішувати проблеми, які можуть виникнути в ході розробки та експлуатації.

2.2 Структурна схема, алгоритм та функціональний опис

2.2.1 Структурна схема та функціональний опис

Загальна структура пристрою визначає сукупність компонентів та їх взаємозв'язки, необхідні для реалізації функції розпізнавання обличчя та керування доступом. Вона є архітектурною основою, яка забезпечує модульність, що дозволяє модифікувати або масштабувати систему без значних змін у дизайні. Структура визивного пристрою включає наступні ключові компоненти:

- камера, яка захоплює відеопотік або статичні зображення обличчя користувача, забезпечуючи вхідні дані для обробки. Камера є первинним сенсором системи, і її якість (роздільна здатність, чутливість до освітлення) критично впливає на точність розпізнавання;

- обчислювальна платформа, яка виконує обробку відеопотоку, реалізує алгоритми розпізнавання обличчя, керує базою даних і вихідними пристроями;

- база фото, що зберігає унікальні цифрові ознаки (дескриптори) обличчя zareєстрованих користувачів. Вона є локальним сховищем, що гарантує

конфіденційність даних, оскільки обробка відбувається без хмарних сервісів;

— вихідні пристрої, які забезпечують фізичну взаємодію з користувачем і середовищем. До них належать бузер для звукового сповіщення та світлодіоди для візуального сповіщення, механізм замка для фізичного відчинення дверей.

— джерело живлення, яке гарантує автономну роботу системи, що є важливим для портативних або віддалених застосувань.

Структурну схему зображено на рисунку 2.1.

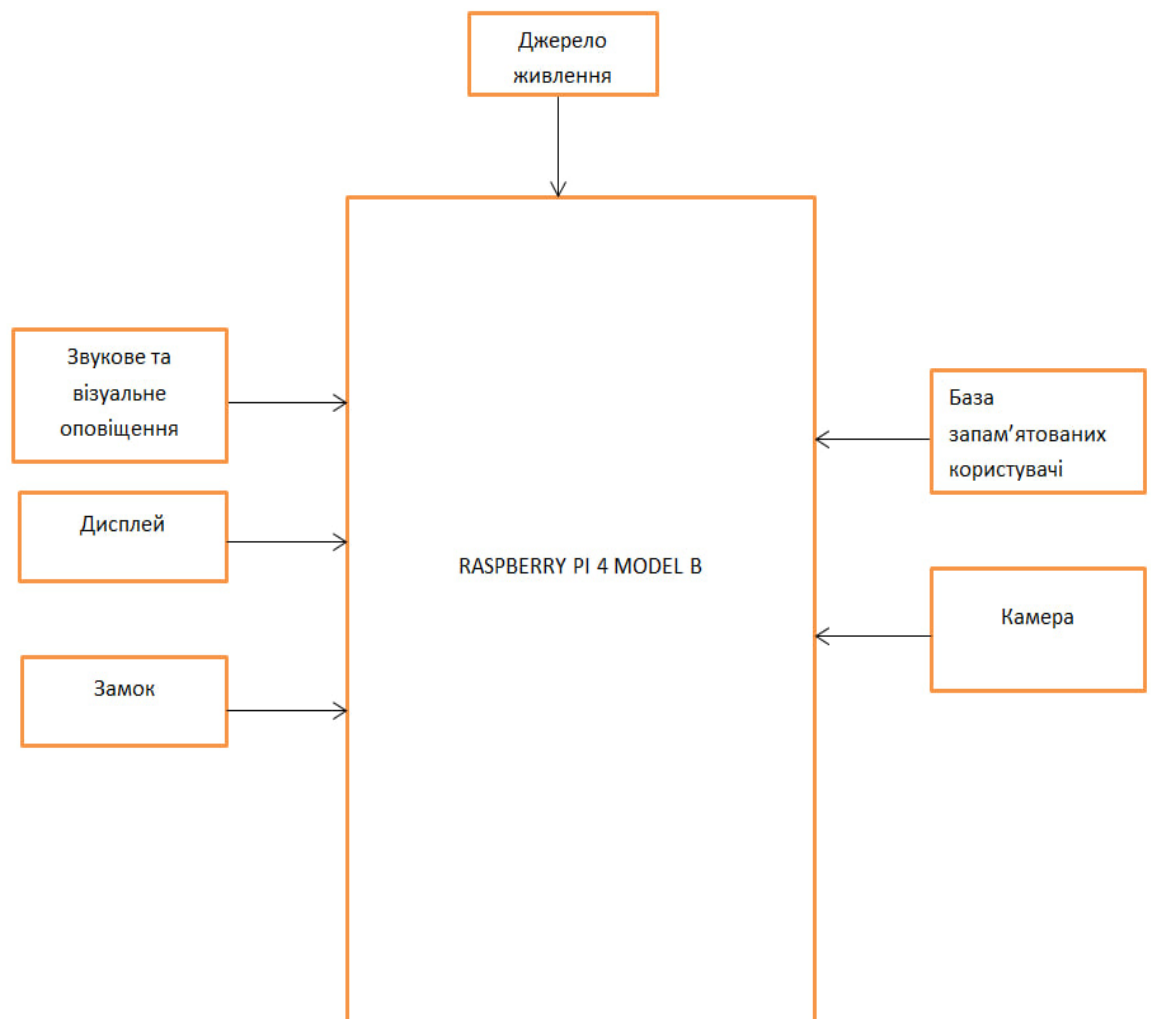


Рисунок 2.1 — Структурна схема пристрою

Компоненти взаємодіють наступним чином: камера передає відеопотік на обчислювальну платформу, яка обробляє дані, звертається до бази даних для ідентифікації користувача і, залежно від результату, активує вихідні пристрої.

Інтерфейси зв'язку забезпечують додаткову функціональність, таку як віддалене оновлення бази даних. Ця структура є модульною, що полегшує заміну компонентів (наприклад, камери з вищою роздільною здатністю) або додавання нових функцій (наприклад, інтеграція з розумним домом) [15].

Функціональна модель ілюструє, як система виконує свою основну задачу — надання доступу авторизованим користувачам шляхом розпізнавання їхнього обличчя. Вона представляє розкладання загальної функції на підфункції, що дозволяє чітко визначити ролі кожного компонента та їх взаємодію. Функціональна модель є схемою, яка показує потік даних і процесів у системі, подібно до функціональної блок-діаграми, описаної в літературі з системного проєктування (System Functional Structure Diagram) [16].

Загальна функція системи — надання доступу авторизованим користувачам на основі розпізнавання обличчя в реальному часі. Основні підфункції:

- захоплення відеопотоку для отримання зображень або відео з камери для подальшої обробки.
- виявлення обличчя: Визначення наявності обличчя у відеопотоці за допомогою алгоритмів комп'ютерного зору.
- виділення ознак обличчя та створення унікального цифрового представлення (дескриптора) обличчя на основі його характеристик.
- порівняння з базою даних отриманих ознак із збереженими в базі даних для ідентифікації користувача.
- верифікація авторизації для перевірки, чи є ідентифікований користувач авторизованим для доступу.
- керування доступом для активації механізму відчинення дверей (якщо користувач авторизований) або блокування доступу.
- надання зворотного зв'язку для інформування користувача про результат розпізнавання через звуковий сигнал або візуальне повідомлення.
- керування базою даних для додавання нових користувачів або видалення існуючих профілів.

2.2.2 Загальний алгоритм роботи пристрою

Загальний алгоритм роботи пристрою базується на послідовному виконанні етапів, починаючи з ініціалізації системи та завершуючи наданням або відмовою у доступі. Алгоритм забезпечує безперервний цикл обробки зображення користувача з подальшим прийняттям рішення про дозвіл доступу. Кроки алгоритму:

- ініціалізація системи, при подачі живлення здійснюється запуск програмного забезпечення, ініціалізація камери, завантаження моделі розпізнавання обличчя та користувацької бази даних;

- захоплення зображення, камера постійно захоплює відеопотік або серію зображень для обробки;

- виявлення обличчя, із захопленого зображення визначається наявність обличчя за допомогою алгоритму детекції;

- аналіз обличчя, якщо обличчя виявлено, із нього виділяються характерні ознаки (дескриптори), які використовуються для ідентифікації;

- порівняння з базою даних, згенерований дескриптор порівнюється з усіма записами в локальній базі зареєстрованих користувачів.

Етапи прийняття рішення:

- якщо знайдено збіг, користувач вважається авторизованим;

- якщо збіг відсутній, доступ заборонено.

Вихідна реакція системи:

- у випадку авторизації активується реле, що керує механізмом замка, подається звуковий сигнал та повідомлення на дисплеї;

- у випадку відмови пристрій автоматично переходить до спеціально передбаченої процедури інформування про відмову, цей етап реалізується шляхом подачі звукового сигналу;

- повернення до початку циклу, на цьому етапі пристрій переводиться у стан очікування наступного користувача, при цьому підтримується повна функціональна готовність до здійснення нового процесу розпізнавання.

Загальну блок схему алгоритму для забезпечення роботи пристрою зображено на рисунку 2.2.

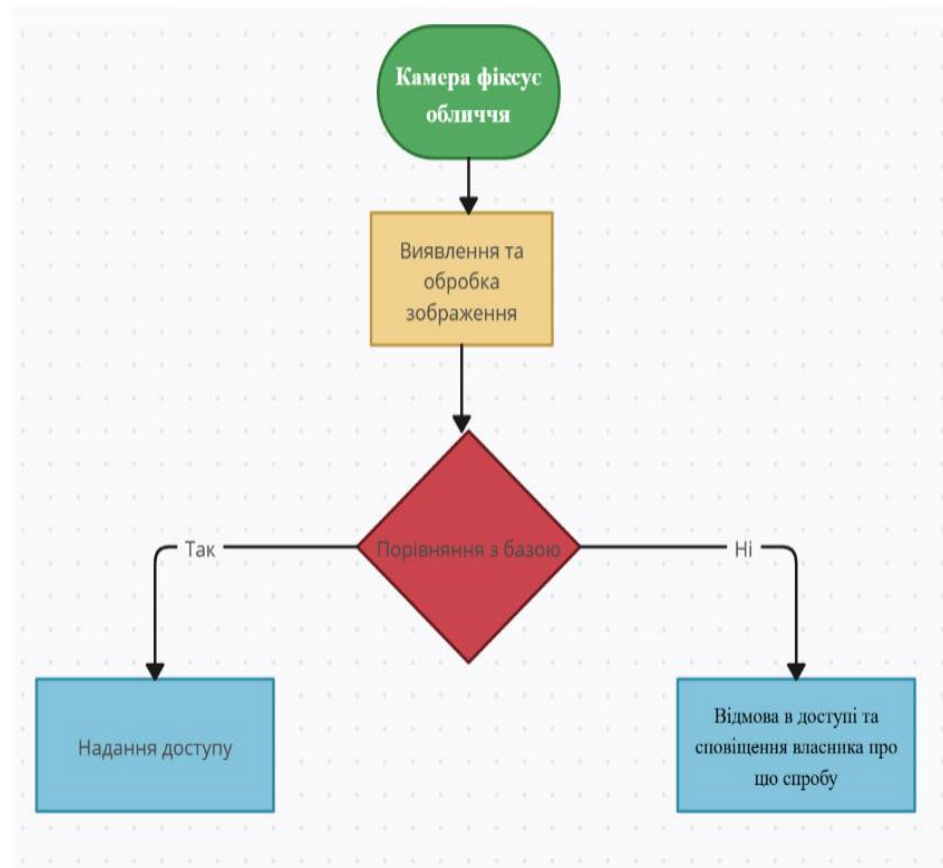


Рисунок 2.2 — Блок схема загального алгоритму пристрою

2.3 Розрахунок переваги обраного рішення за критерієм енергоефективності

Одним із надзвичайно важливих, а подекуди й визначальних критеріїв, на який звертають особливу увагу інженери та розробники при підборі апаратної складової таких пристроїв, безумовно є енергоефективність. Адже саме здатність електронного пристрою працювати з мінімальним споживанням електроенергії, зберігаючи при цьому високу продуктивність, стабільність та надійність, є тим фактором, який може суттєво вплинути як на загальні експлуатаційні характеристики пристрою. Особливо це актуально у випадках, коли пристрій може працювати автономно або в умовах нестабільного енергоживлення. Тому було проведено оцінку енерговитрат вибраної апаратної конфігурації в побутових

умовах використання пристрою, що дозволяє обґрунтувати доцільність її використання у контексті загального енергоспоживання [17].

Склад пристрою має основні енергоспоживчі елементи, які зображено в таблиці 2.1:

Таблиця 2.1 — Основні енергоспоживчі елементи для пристрою

Компонент	Середній струм,мА	Напруга живлення, В	Середня потужність, Вт
Raspberry Pi 4 Model B	600-1000	5.1	5.1
Камера (Raspberry Pi Camera V2)	250	5	1.25
Реле 5В (2 шт.)	70х2 (в активному стані)	5	0.7
Бuzzer активний 5В	30	5	0.15
LCD-дисплей	25	5	0.125
Соленоїдний замок (12В)	300	12	3.6

Ураховуючи середні значення, можна оцінити потужність у робочому стані пристрою:

$$P(\text{загальна}) \approx 5.1 + 1.25 + 0.7 + 0.15 + 0.126 + 3.6 \approx 10.925 \text{ Вт}$$

Оскільки реле, бuzzer та дисплей працюють лише протягом короткого часу після авторизації, а не постійно, для точнішої оцінки варто враховувати час активного стану. Наприклад, якщо активний цикл триває ~10 секунд при кожному спрацюванні, а за добу відбувається в середньому 50 звернень, енергоспоживання цих компонентів буде мізерним порівняно з базовим фоном Raspberry Pi.

Розрахунок постійно активних елементів (Raspberry Pi + камера):

$$E_1 = (5.1 + 1.25) \times 24 = 6.35 \times 24 = 152.4 \text{ Вт/год}$$

Розрахунок роботи інших компонентів (сумарно 4.225 Вт, активні ~500 секунд/доба):

$$E_2 = 4.225 \times 0.139 \approx 0.587 \text{ Вт/год}$$

Оцінка добового енергоспоживання пристрою:

$$E(\text{добове}) \approx 152.4 + 0.587 \approx 152.99 \text{ Вт/год}$$

З урахуванням вищенаведених розрахунків, вибір платформи Raspberry Pi 4 у поєднанні з енергоефективними периферійними модулями дозволяє досягти високої енергоефективності системи. Добове споживання становить лише близько 152.99 Вт·год, що значно менше у порівнянні з альтернативами. Це робить запропоноване рішення доцільним з точки зору витрат енергії та підходящим для умов з обмеженим живленням або автономної роботи з використанням акумуляторів чи джерел безперебійного живлення.

2.4 Варіативний аналіз і вибір базових інструментів для технічного рішення

Розробка визивного пристрою з розпізнаванням обличчя потребує ретельного підходу до вибору апаратного та програмного забезпечення, яке відповідатиме ключовим вимогам: енергоефективність, доступна вартість, достатня обчислювальна потужність для роботи в реальному часі та гнучкість для подальшого вдосконалення. У цьому розділі проведено детальний аналіз альтернативних варіантів для основних компонентів системи – обчислювальної платформи, камери, механізму керування доступом і джерела живлення – а також програмного забезпечення, включаючи операційну систему, мову програмування та бібліотеки. Особливу увагу приділено оцінці переваг та недоліків кожного варіанта, що дозволило всебічно охарактеризувати їх з точки зору практичного застосування в межах конкретного завдання. На підставі проведеного порівняльного аналізу було здійснено обґрунтований вибір оптимального набору базових апаратних та програмних інструментів, які найбільш повно відповідають

поставленим цілям і забезпечують ефективну реалізацію функціоналу проєкту з урахуванням обмежень та вимог [18].

2.4.1 Аналіз і вибір апаратного забезпечення

Обчислювальна платформа: для реалізації викличного пристрою з розпізнаванням обличчя необхідно обрати платформу, яка здатна обробляти відеопотоки та виконувати алгоритми комп'ютерного зору в реальному часі, при цьому залишаючись енергоефективною та доступною за ціною. Розглянемо три основні варіанти: Raspberry Pi 4, Arduino та NVIDIA Jetson Nano.

Raspberry Pi 4 є одноплатним комп'ютером із 4-ядерним процесором Cortex-A72, що працює на частоті 1.5 ГГц, і має до 4 ГБ оперативної пам'яті. Ця платформа підтримує Wi-Fi, Bluetooth і розширені інтерфейси GPIO, що дозволяє легко підключати камеру, дисплей та механізми керування. Енергоспоживання в робочому режимі становить близько 4.5 Вт. Завдяки широкій спільноті та оптимізованій підтримці бібліотек, таких як OpenCV, Raspberry Pi 4 є популярним вибором для вбудованих систем із завданнями комп'ютерного зору [19]. Фото одноплатного комп'ютера зображено на рисунку 2.3.

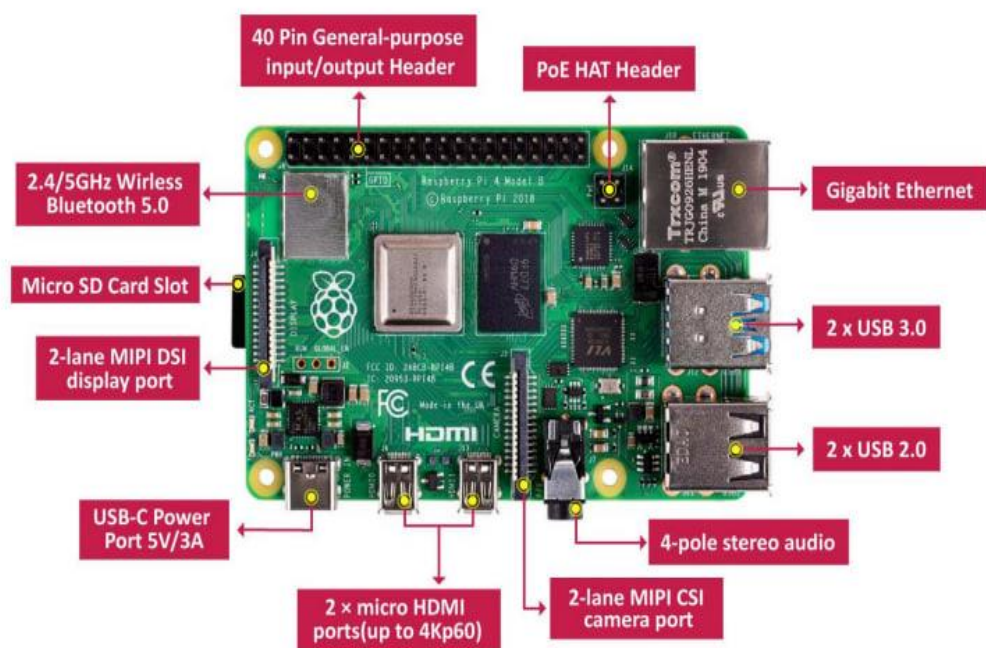


Рисунок 2.3 — Raspberry Pi 4

Arduino, зокрема модель Uno, є мікроконтролером із 8-бітним процесором ATmega328P, 32 КБ флеш-пам'яті та мінімальним енергоспоживанням – приблизно 0.05 Вт. Простота використання робить його ідеальним для невеликих проєктів із сенсорами чи простими логічними операціями. Однак обмежена обчислювальна потужність не дозволяє виконувати складні алгоритми розпізнавання обличчя, які потребують обробки зображень у реальному часі, що робить Arduino непридатним для цього проєкту. Фото мікроконтролера зображено на рисунку 2.4.

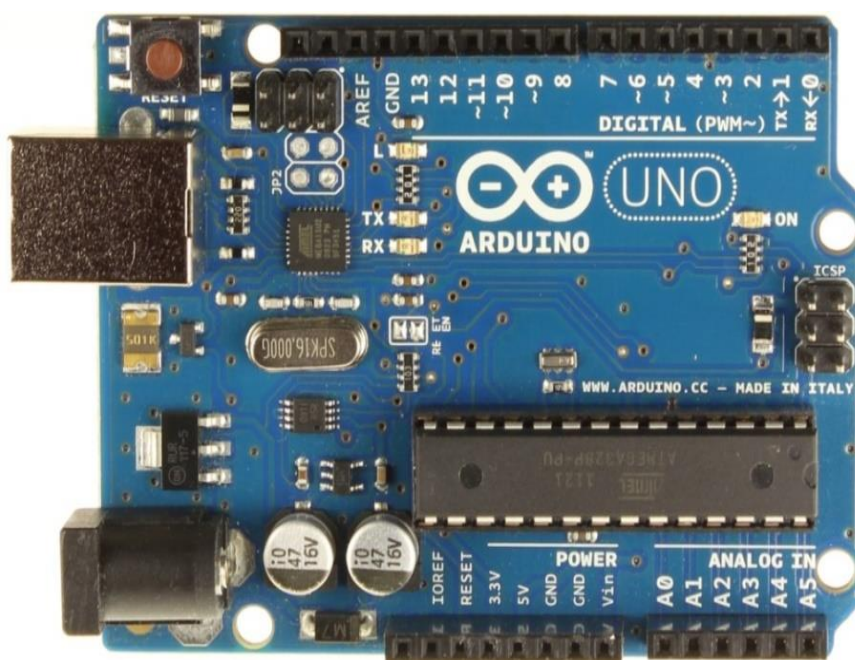


Рисунок 2.4 — Arduino Uno

NVIDIA Jetson Nano пропонує значно вищу продуктивність завдяки 4-ядерному процесору Cortex-A57 і вбудованому GPU Maxwell, що підходить для роботи з глибокими нейронними мережами. Обсяг оперативної пам'яті становить 4 ГБ, а споживання енергії в базовому режимі – близько 10 Вт, що забезпечує лише 7.5 години автономної роботи від акумулятора ємністю 5000 мА·год. Попри те, що обчислювальна платформа Jetson Nano демонструє високий рівень продуктивності та є перспективним рішенням для реалізації складних завдань, пов'язаних із машинним навчанням, зокрема для розпізнавання обличчя, обробки відеопотоку в реальному часі та інших інтелектуальних функцій, її використання у проєкті визивного пристрою викликає суттєві сумніви з огляду на низку

критичних чинників. Найбільш вагомим з них є підвищене енергоспоживання, що безпосередньо впливає на автономність системи.

Фото цього мікрокомп'ютера зображено на рисунку 2.5.



Рисунок 2.5 — NVIDIA Jetson Nano Developer Kit

Отже, вибір зупинено на Raspberry Pi 4. Ця платформа забезпечує достатню обчислювальну потужність для реалізації алгоритмів розпізнавання обличчя за допомогою OpenCV, має низьке енергоспоживання (6.05 Вт із камерою та дисплеєм, що дає 12.4 години автономності), і є економічно вигідною. Її гнучкість і підтримка екосистеми роблять її оптимальним рішенням для балансу між продуктивністю та енергоефективністю. Призначення виводів інтерфейсу GPIO обраної платформи зображено на рисунку 2.6.

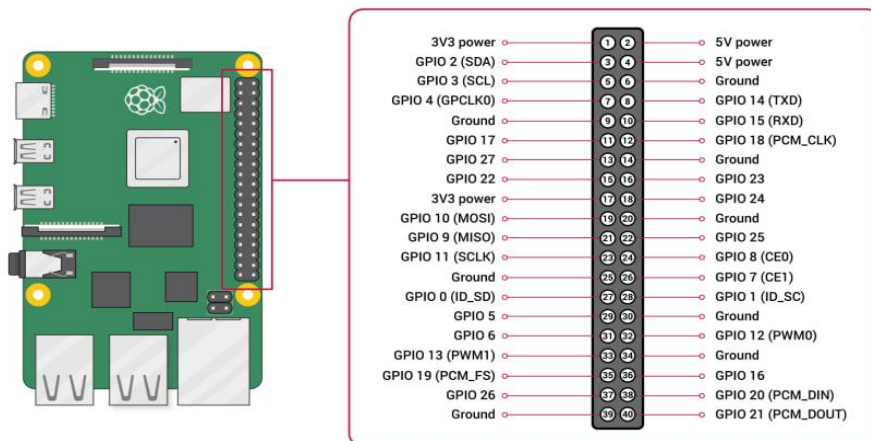


Рисунок 2.6 – Призначення виводів інтерфейсу GPIO

Камера є ключовим елементом для захоплення відеопотоку, необхідного для розпізнавання обличчя. Розглянемо три варіанти: Raspberry Pi Camera Module v2, USB-камеру (веб-камеру) і камеру ноутбука для тестування.

Raspberry Pi Camera Module v2 оснащена 8-мегапіксельним сенсором, здатним записувати відео в роздільній здатності 1080p при 30 кадрах за секунду. Її енергоспоживання становить 0.75 Вт, а підключення через CSI-інтерфейс забезпечує пряму інтеграцію з Raspberry Pi без додаткових драйверів. Ця камера оптимізована для роботи з Raspberry Pi, що мінімізує затримки в обробці даних.

USB-камера, наприклад, модель Logitech C270, пропонує роздільну здатність від 720p до 1080p залежно від моделі. Її споживання енергії варіюється від 1.5 до 2 Вт. Підключення через USB-порт є універсальним, але потребує встановлення драйверів, що може ускладнити налаштування на Raspberry Pi. Вище енергоспоживання також скорочує автономність пристрою до 7–8 годин, що є менш вигідним порівняно з CSI-варіантом.

Камера ноутбука, як вбудований модуль із роздільною здатністю 720p, споживає близько 1 Вт і доступна без додаткових витрат у тестовому середовищі. Однак її використання обмежене умовами тестування, оскільки вона не може бути інтегрована в автономний пристрій і має нижчу якість зображення, що впливає на точність розпізнавання.

Вибір зупинено на Raspberry Pi Camera Module v2 завдяки її безпосередній сумісності з платформою, низькому енергоспоживанню та стабільній роботі в реальному часі. Ця камера дозволяє уникнути складнощів із драйверами та забезпечує достатню якість зображення для ідентифікації облич, що є важливим для виконання задачі.

Механізм керування доступом відповідає за фізичне відчинення дверей після успішного розпізнавання. Розглянемо три альтернативи: електромагнітний замок із реле, серводвигун і соленоїдний замок.

Електромагнітний замок із реле працює на принципі магнітного утримання дверей і активується через реле, підключене до GPIO Raspberry Pi. Споживання енергії в активному стані становить приблизно 0.2 Вт, а час роботи – близько 10

секунд на кожне спрацьовування. Його простота підключення та низьке енергоспоживання роблять його привабливим для побутового використання, хоча надійність залежить від якості монтажу.

Серводвигун використовується для механічного повороту засувки або ручки. Його споживання енергії становить 0.5–1 Вт залежно від навантаження, а час роботи – 5–10 секунд. Однак серводвигуни потребують точного налаштування для стабільної роботи та мають вищу енергоємність, що може скоротити автономність пристрою.

Соленоїдний замок працює за принципом електромагнітного переміщення засувки і споживає 0.3–0.5 Вт під час активації, з часом роботи близько 10 секунд. Але механічний знос може знизити довговічність порівняно з магнітним замком. Цей варіант підходить для більш складних систем, але менш економічний.

Джерело живлення має забезпечувати стабільну роботу пристрою як у стаціонарних, так і в автономних умовах. Розглянемо два основні варіанти: стаціонарне живлення і акумуляторне живлення.

Стаціонарне живлення на базі блоку живлення 5V/3A забезпечує постійне постачання енергії з потужністю 15 Вт. Підключення через USB-C робить його зручним для використання в офісах чи будинках із доступом до електромережі. Однак цей варіант не підходить для автономних сценаріїв, де пристрій має працювати без підключення до розетки.

Акумуляторне живлення на основі Li-ion батареї 18650 із тримачем і ємністю 5000 мА·год при напрузі 5 В забезпечує енергію до 25 Вт·год (з ефективністю 90% – 22.5 Вт·год). Тривалість роботи при споживанні 6.05 Вт становить близько 12.4 години, що перевищує вимогу в 8 годин. Цей варіант ідеальний для портативності, але потребує періодичної зарядки або заміни.

Вирішно комбінувати обидва варіанти: блок живлення для стаціонарного режиму та Li-ion акумулятор для автономності. Такий підхід забезпечує гнучкість, дозволяючи використовувати пристрій як у постійному підключенні, так і в умовах відсутності електроенергії. Вибір обґрунтовано необхідністю адаптивності до різних умов експлуатації.

2.4.2 Аналіз і вибір програмного забезпечення

Операційна система відіграє ключову роль у забезпеченні стабільної роботи апаратного забезпечення та програмного коду. Для викличного пристрою розглянуто лише одну основну опцію, оскільки альтернативи менш підходять для Raspberry Pi.

Raspberry Pi OS, раніше відомий як Raspbian, є офіційною операційною системою для Raspberry Pi, базованою на Debian. Вона оптимізована для роботи з цією платформою, підтримує широкий спектр мов програмування, включаючи Python і C++, а також бібліотеки, такі як OpenCV. Безкоштовна і регулярно оновлювана, ця ОС забезпечує легку інтеграцію з апаратними компонентами, такими як камера та GPIO, і має розвинену документацію та спільноту. Інші варіанти, такі як Ubuntu чи Windows IoT, потребують більше ресурсів або мають обмежену підтримку бібліотек, що робить їх менш привабливими. Вибір Raspberry Pi OS обґрунтовано його стабільністю, оптимізацією та зручністю для вбудованих систем.

Для реалізації алгоритмів розпізнавання обличчя та обробки даних необхідно обрати мову програмування, яка забезпечить високу продуктивність і сумісність із обраними бібліотеками. Розглянуто три основні варіанти: Python, C++ і Java.

Python є високорівневою мовою, яка вирізняється простотою синтаксису і великою кількістю бібліотек, включаючи OpenCV. Вона широко використовується для швидкої розробки прототипів і має велику спільноту, що полегшує вирішення проблем. Однак через інтерпретовану природу Python його продуктивність нижча, і час обробки одного кадру на Raspberry Pi може перевищувати 1 секунду, що не відповідає вимозі реального часу для викличного пристрою.

C++ є компілюваною мовою, яка забезпечує високу швидкість виконання завдяки прямому доступу до апаратних ресурсів. Вона підтримує OpenCV через C++ API, що дозволяє оптимізувати алгоритми розпізнавання обличчя, такі як

Haar Cascade і LBPH. Хоча освоєння C++ вимагає більше зусиль порівняно з Python, його перевага в продуктивності є критичною для вбудованих систем із обмеженими ресурсами, такими як Raspberry Pi. Крім того, низькорівневе управління пам'яттю дозволяє ефективно використовувати 4 ГБ ОЗУ [20].

Java пропонує платформи-незалежність завдяки JVM, що може бути корисним для майбутнього перенесення коду. Однак накладні витрати на виконання JVM і менша оптимізація для вбудованих систем роблять її менш ефективною порівняно з C++. Підтримка OpenCV у Java також обмежена, що ускладнює інтеграцію з існуючими рішеннями.

Вибір зроблено на користь C++ завдяки його високій продуктивності, що є вирішальним для обробки відеопотоку в реальному часі, і сумісності з OpenCV. Ця мова дозволяє мінімізувати затримки та ефективно використовувати обчислювальні ресурси Raspberry Pi, що відповідає цілям проєкту.

Для реалізації розпізнавання обличчя необхідно обрати бібліотеки, які забезпечать необхідні алгоритми та будуть сумісні з обраною платформою та мовою програмування. Розглянуто чотири основні варіанти: OpenCV, Dlib, TensorFlow Lite і OpenFace.

OpenCV є найбільш універсальною бібліотекою для комп'ютерного зору, яка підтримує як C++, так і Python. Вона включає широкий спектр алгоритмів, таких як Haar Cascade для швидкого виявлення обличчя, Local Binary Patterns Histograms (LBPH) для ідентифікації та підтримку глибоких нейронних мереж (DNN) для покращення точності. OpenCV оптимізована для Raspberry Pi, має розвинену документацію та велику спільноту, що полегшує розробку. Її точність ($\geq 90\%$) і час обробки (до 1 секунди) відповідають вимогам проєкту, а модульність дозволяє додавати нові функції, наприклад, інтеграцію з IoT.

Dlib пропонує потужні алгоритми, такі як HOG із SVM або власні моделі розпізнавання обличчя, які базуються на глибоких нейронних мережах. Ця бібліотека забезпечує високу точність, але потребує значних обчислювальних ресурсів і щонайменше 4 ГБ ОЗУ для стабільної роботи. На Raspberry Pi це може призвести до перевантаження системи, особливо при одночасній обробці

відеопотоку, що робить Dlib менш практичним для цього проєкту.

TensorFlow Lite є легковаговою версією TensorFlow, призначеною для вбудованих систем. Вона підтримує попередньо навчені моделі для розпізнавання обличчя, але потребує додаткового етапу підготовки та оптимізації моделей, що ускладнює інтеграцію на Raspberry Pi. Хоча вона ефективна для нейронних мереж, її складність і потреба в GPU-оптимізації роблять її менш зручною порівняно з OpenCV для поточних задач.

OpenFace фокусується на розпізнаванні обличчя з використанням глибоких нейронних мереж і забезпечує високу точність. Однак ця бібліотека менш оптимізована для вбудованих систем і потребує більше ресурсів, ніж доступно на Raspberry Pi. Її інтеграція також складніша через обмежену документацію, що робить її менш привабливою для швидкої розробки.

Вибір зупинено на OpenCV як основній бібліотеці завдяки її широкій функціональності, оптимізації для Raspberry Pi та підтримці C++ API. Вона дозволяє реалізувати весь конвеєр розпізнавання – від виявлення до ідентифікації із достатньою швидкістю та точністю. Її модульна структура забезпечує гнучкість для майбутнього розширення, а велика спільнота полегшує вирішення технічних питань під час розробки.

3 РЕАЛІЗАЦІЯ ВИЗИВНОГО ПРИСТРОЮ З РОЗПІЗНАВАННЯМ ОБЛИЧЧЯ

3.1 Побудова схеми підключення компонентів

Функціонування визивного пристрою з розпізнаванням обличчя ґрунтується на інтеграції апаратних компонентів, зокрема одноплатного комп'ютера Raspberry Pi, виконавчих механізмів, сенсорів та індикаторів. Побудована схема забезпечує логічне з'єднання усіх модулів для реалізації повного циклу роботи пристрою — від захоплення зображення до фізичного відкривання замка.

На рисунку 3.1 представлено електрично-принципіву схему з'єднань пристрою. Центральним елементом є Raspberry Pi 4, який виконує функцію обробки відеопотоку, розпізнавання обличчя, прийняття рішення та активації вихідних пристроїв. З'єднання здійснено через виводи GPIO.

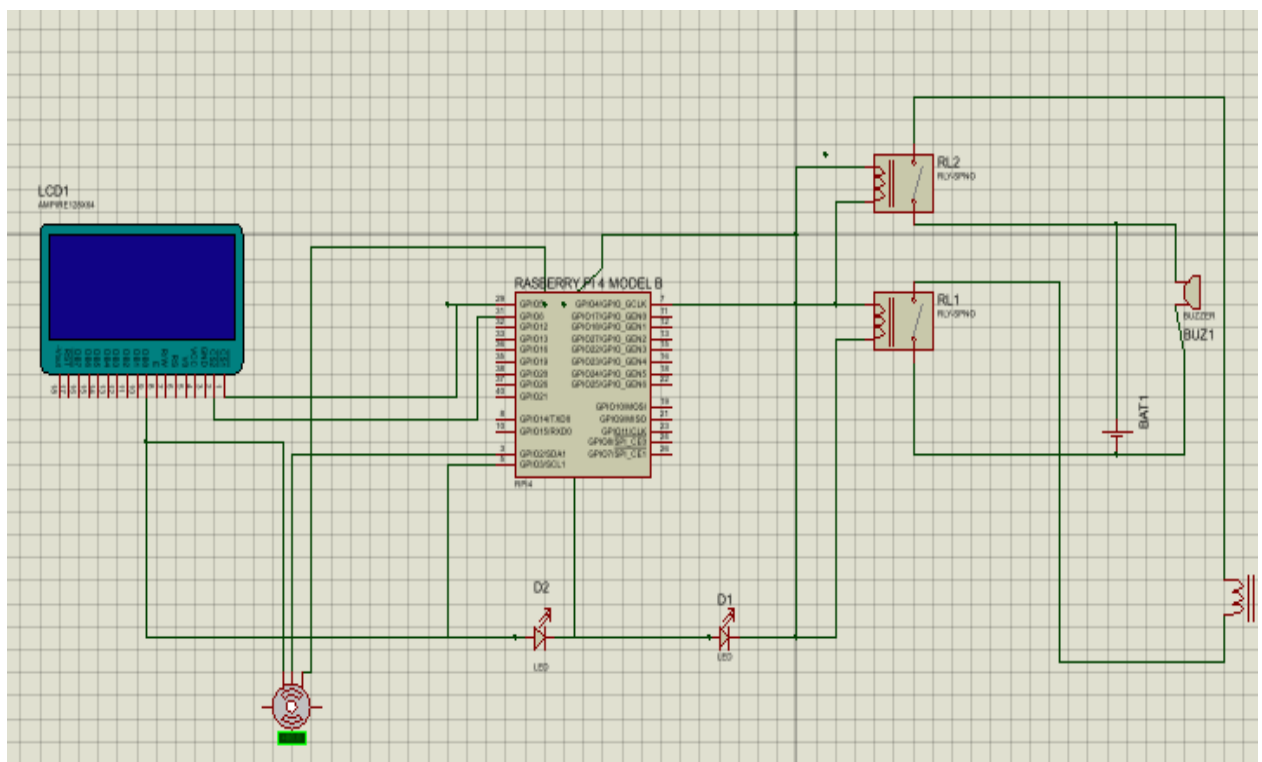


Рисунок 3.1 — Електрично-принципова схема з'єднань

До Raspberry Pi підключено наступні компоненти:

— камера (не показана на схемі, через використання вбудованої камери

ноутбука) підключається через CSI-інтерфейс або USB-порт. Вона забезпечує захоплення зображень обличчя для подальшої обробки;

- реле використовується для керування силовими компонентами — соленоїдним електрозамком та живленням інших модулів. Кожне реле отримує сигнали керування з GPIO виводів Raspberry Pi;

- соленоїдний замок підключається через реле до окремого джерела живлення. Його активація виконується при позитивному результаті розпізнавання обличчя;

- сервопривід може використовуватись як альтернатива замку для фізичної взаємодії з дверним механізмом. Його сигнальний вхід також підключений до GPIO;

- дисплей слугує інтерфейсом з користувачем — виводить повідомлення про статус авторизації: успіх або відмова;

- світлодіоди (LED) використовуються для візуальної індикації поточного стану: зелений – доступ дозволено, червоний – відмовлено;

- звуковий сигналізатор (buzzer) спрацьовує після прийняття рішення про доступ, підтверджуючи взаємодію з користувачем;

- акумуляторне живлення забезпечує автономну роботу системи. У схемі передбачено живлення як логічної частини, так і силових елементів;

Схема є модульною, що дозволяє замінювати або додавати компоненти без повного перегляду архітектури. Наприклад, LCD-дисплей або сервопривід можуть бути опційними — усе залежить від сценарію застосування пристрою (побутовий, офісний, промисловий).

3.2 Реалізація програмної логіки пристрою

Програмна реалізація визивного пристрою з розпізнаванням обличчя побудована за модульною архітектурою. Кожен модуль відповідає за окрему фазу обробки: від захоплення зображення до видачі результату у вигляді керування доступом. Нижче описано основні програмні блоки для роботи пристрою.

Модуль захоплення даних відповідає за ініціалізацію камери та отримання

відеопотоку в реальному часі. Забезпечує подачу якісного зображення для подальшої обробки.

Лістинг 3.1 — відповідає за реалізацію захоплення зображень з камери.

```
VideoCapture cap;
cap.open(0);
if (!cap.isOpened()) {
    cerr << "Помилка доступу до камери!" << endl;
    return -1;
}
```

Модуль попередньої обробки виконує перетворення зображення у відтінки сірого та виявлення облич за допомогою алгоритму каскадів Хаара.

Лістинг 3.2 — відповідає за детекцію обличчя та підготовку зображення.

```
cvtColor(frame, gray, COLOR_BGR2GRAY);
CascadeClassifier face_cascade;
face_cascade.load("haarcascade_frontalface_default.xml");
vector<Rect> faces;
face_cascade.detectMultiScale(gray, faces, 1.3, 5);
```

Модуль виділення ознак обробляє область обличчя, створюючи дескриптор на основі алгоритму LBPH для того, щоб провести ідентифікацію особи.

Лістинг 3.3 – код, що реалізує побудову дескриптора обличчя

```
Mat faceROI = gray(faces[0]);
int label;
double confidence;
Ptr<LBPHFaceRecognizer> model = LBPHFaceRecognizer::create();
model->read("trainer.yml");
model->predict(faceROI, label, confidence);
```

Модуль класифікації виконує ключовий етап – прийняття рішення щодо ідентифікації особи на основі попередньо отриманих та оброблених даних. Його основна задача полягає у порівнянні згенерованого дескриптора із заздалегідь підготовленою та збереженою в базі даних колекцією дескрипторів осіб, яка раніше сформувалась шляхом проходження процедури реєстрації в системі.

Лістинг 3.4 – код, що відповідає за класифікацію користувача за допомогою тренованої моделі.

```
if (confidence < 60) {
    name = (label == 1) ? "Андрій" : "Інший користувач";
    access_granted = true;
} else {
    name = "Невідомо";
    access_granted = false;
}
```

Модуль виводу результатів забезпечує інформування користувача про результат розпізнавання: у разі успіху активується реле, включається зелений світлодіод і бузер; у разі невдачі – червоний індикатор.

Лістинг 3.5 – код, що реалізує реакцію системи на результат розпізнавання.

```
if (access_granted) {
    digitalWrite(RELAY_PIN, HIGH);
    digitalWrite(GREEN_LED, HIGH);
    digitalWrite(BUZZER_PIN, HIGH);
    delay(3000);
    digitalWrite(RELAY_PIN, LOW);
    digitalWrite(GREEN_LED, LOW);
    digitalWrite(BUZZER_PIN, LOW);
    digitalWrite(RED_LED, HIGH);
    delay(2000);
    digitalWrite(RED_LED, LOW);
}
```

Модуль збереження фото неавторизованого користувача випадку, коли користувач не ідентифікований, система автоматично зберігає зображення його обличчя у локальну директорію для подальшого аналізу або додавання до бази даних.

Лістинг 3.6 – код, що зберігає обличчя у файл при невдалому розпізнаванні

```
if (!access_granted) {
    string filename = "unknown_faces/face_" + to_string(time(0)) + ".jpg";
    imwrite(filename, faceROI);
}
```

Модуль запису журналу доступу, система веде текстовий лог, де фіксує

дату, час, ім'я користувача (або повідомлення "Невідомо") та результат доступу. Це дозволяє адміністратору перевірити історію викликів.

Функція `currentDateTime()` отримує та записує поточні дані дати і часу у форматі `YYYY-MM-DD HH:MM:SS`.

Лістинг 3.7 – код, що записує лог події у файл

```
ofstream logfile("access_log.txt", ios::app);
logfile << "[" << currentDateTime() << "] Користувач: " << name
        << " | Результат: " << (access_granted ? "Доступ дозволено" :
"Відмовлено") << endl;
logfile.close();
```

Модуль індикації на екрані, пристрій оснащений дисплеєм, який виводить повідомлення "Доступ дозволено" або "Доступ заборонено" залежно від результату розпізнавання.

Лістинг 3.8 – код, що виводить текст на дисплей

```
display.clearDisplay();
display.setTextSize(1);
display.setTextColor(WHITE);
display.setCursor(0, 0);
display.println(access_granted ? "Доступ дозволено" : "Доступ заборонено");
display.display();
```

3.3 Можливе розширення функціональності пристрою за допомогою додаткових датчиків

Сучасні технології дедалі більше орієнтуються на автоматизацію та інтелектуалізацію систем, що дозволяє пристроям не лише виконувати базові функції, а й адаптуватися до змін у навколишньому середовищі, підвищуючи свою ефективність і зручність використання. Визивний пристрій із розпізнаванням обличчя на базі Raspberry Pi, який використовується для контролю доступу, не є винятком із цього правила. У своїй базовій версії такий пристрій зазвичай складається з камери для захоплення відеопотоку, дисплея для відображення результатів і основного модуля обробки даних — Raspberry Pi. Проте потенціал подібної системи значно зростає, якщо інтегрувати в її архітектуру додаткові сенсори, які дозволяють зробити роботу пристрою більш

інтелектуальною, енергоефективною та надійною. Додавання таких сенсорів не вимагає кардинальних змін у програмному чи апаратному забезпеченні, але водночас суттєво розширює функціональні можливості системи, роблячи її більш адаптивною до реальних умов експлуатації. Розглянемо детально, які саме сенсори можна додати до пристрою, як вони працюють і який вплив вони мають на загальну функціональність.

Першим сенсором, який варто розглянути для інтеграції, є пасивний інфрачервоний датчик руху, відомий як PIR (Passive Infrared Sensor). Цей сенсор є одним із найпростіших і водночас найефективніших рішень для оптимізації роботи пристрою. Він працює на основі виявлення інфрачервоного випромінювання, яке виділяє людське тіло у вигляді тепла, коли людина рухається в зоні дії сенсора. Основна перевага PIR-сенсора полягає в його низькому енергоспоживанні і простоті підключення до Raspberry Pi через GPIO. Його використання дозволяє реалізувати концепцію інтелектуального очікування: камера і система розпізнавання обличчя активуються лише тоді, коли датчик фіксує рух людини перед пристроєм. Це означає, що у ті моменти, коли перед дверима нікого немає, камера не працює, а Raspberry Pi не витрачає ресурси на обробку порожніх кадрів. Такий підхід не лише зменшує навантаження на процесор, що може бути критичним для Raspberry Pi 4 Model B при безперервній роботі, але й значно економить електроенергію, що особливо важливо для систем із автономним живленням. Наприклад, якщо людина підходить до дверей, PIR-сенсор вловлює її рух і надсилає сигнал на Raspberry Pi, який у відповідь активує камеру, запускає алгоритми виявлення та розпізнавання обличчя, а після завершення процесу знову вимикає камеру, повертаючись у режим очікування. Це не лише підвищує енергоефективність, але й подовжує термін служби камери та інших компонентів, зменшуючи їх зношування через постійну роботу.

Технічні характеристики PIR-сенсора роблять його ідеальним для використання у визивному пристрої. Він працює від напруги 5 В, яку легко забезпечити через піни Raspberry Pi, має кут виявлення до 120 градусів, що дозволяє охопити широку зону перед дверима, і дальність дії до 7 метрів, що

достатньо для більшості сценаріїв використання. Час реакції сенсора становить від 0.3 до 1 секунди, що забезпечує швидке спрацювання системи, а вихідний сигнал у вигляді логічного рівня HIGH при виявленні руху легко обробляється через GPIO Raspberry Pi. Таким чином, інтеграція PIR-сенсора дозволяє зробити систему більш "розумною", економною і чутливою до присутності людини, уникаючи зайвих витрат ресурсів. Датчик зображено на рисунку 3.2.



Рисунок 3.2 — Інфрачервоний датчик руху PIR

Наступним сенсором, який може суттєво покращити функціональність пристрою, є ультразвуковий далекомір HC-SR04. Цей сенсор працює за принципом ехолокації: він випромінює ультразвукові імпульси на частоті 40 кГц, які відбиваються від об'єктів і повертаються до сенсора, дозволяючи обчислити відстань до об'єкта на основі часу, потрібного для проходження сигналу. HC-SR04 є ідеальним доповненням до PIR-сенсора, оскільки дозволяє точно визначити, чи знаходиться людина на оптимальній відстані для розпізнавання обличчя, незалежно від температури тіла чи умов освітлення, які можуть впливати на роботу PIR. Наприклад, система може бути налаштована так, щоб активувати камеру лише тоді, коли людина підходить на відстань меншу за 1 метр від пристрою. Це зменшує кількість помилкових спрацювань, які можуть

виникати через випадкові джерела тепла (наприклад, тварини чи нагріті предмети) у зоні дії PIR-сенсора, і робить роботу пристрою більш точною.

Уявімо сценарій використання: людина підходить до дверей, і ультразвуковий датчик фіксує, що вона перебуває на відстані 80 см від пристрою. HC-SR04 надсилає сигнал на Raspberry Pi, який активує камеру і запускає алгоритм розпізнавання обличчя. Якщо обличчя розпізнане, система відкриває замок або надсилає повідомлення власнику; якщо ні — видає відповідне сповіщення. Технічні характеристики HC-SR04 роблять його зручним для такого використання: він працює від 5 В, має дальність виявлення від 2 см до 400 см із точністю 3 мм, що дозволяє точно визначати відстань. Його інтеграція в систему не потребує складного програмного забезпечення — достатньо кількох рядків коду для обчислення відстані на основі часу відгуку імпульсу. Таким чином, HC-SR04 додає додатковий рівень перевірки, підвищуючи точність активації системи та зменшуючи помилкові спрацьовування. Його зображено на рисунку 3.3.



Рисунок 3.3 — Ультразвуковий далекомір HC-SR04

Ще одним важливим доповненням до системи є IR-датчик, який дозволяє забезпечити ефективну роботу пристрою в умовах недостатньої освітленості. Визивний пристрій часто використовується в зовнішніх умовах, де рівень освітлення може змінюватися від яскравого денного світла до повної темряви в

нічний час. У таких умовах камера може не забезпечити якісне зображення, що ускладнить роботу алгоритмів розпізнавання обличчя. IR-сенсор, який зазвичай складається з фотодіода або фототранзистора, здатен визначати рівень освітленості в зоні дії та надсилати сигнал на Raspberry Pi, який може активувати інфрачервоне підсвічування для камери. Це підсвічування дозволяє отримати чітке чорно-біле зображення обличчя навіть у повній темряві, що достатньо для роботи алгоритму LBPH, який використовує відтінки сірого. Прикладом такого датчика є GY-2561, який зображено на рисунку 3.4.

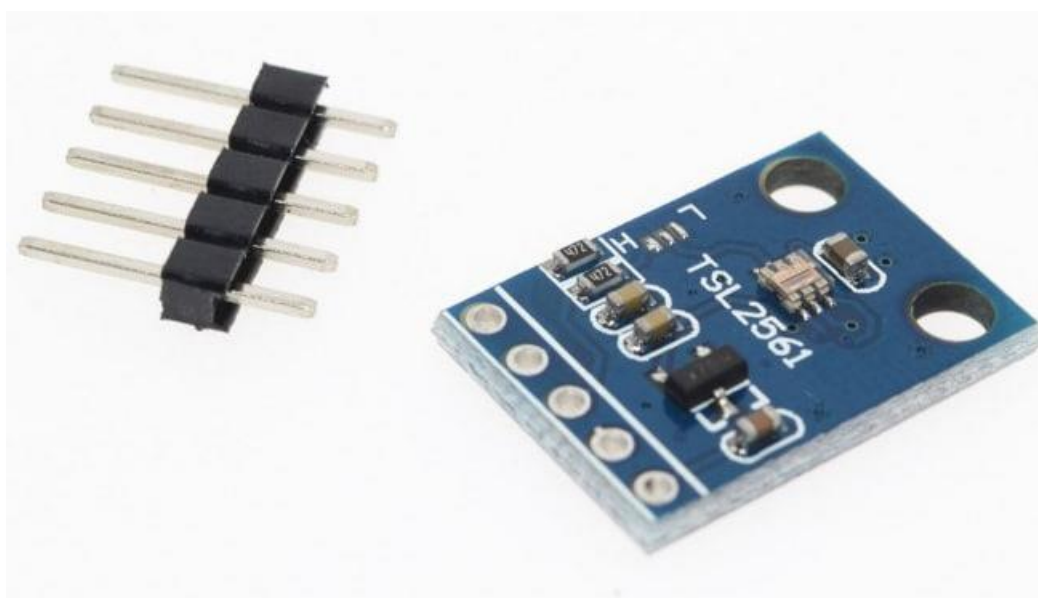


Рисунок 3.4 — Цифровий датчик рівня освітленості GY-2561

Уявімо ситуацію: після заходу сонця рівень освітленості падає нижче порогового значення, і IR-сенсор фіксує це, надсилаючи сигнал на Raspberry Pi. Система автоматично вмикає інфрачервоне підсвічування, що дозволяє камері захопити якісне зображення обличчя відвідувача. Це особливо важливо для безпеки, адже нічні відвідування можуть бути пов'язані з потенційними загрозами, такими як спроби проникнення чи хуліганські дії. IR-сенсор має дальність дії до 30 см і може видавати як цифровий, так і аналоговий сигнал, що дозволяє точно налаштувати поріг спрацьовування залежно від умов. Його інтеграція не потребує значних змін у коді — достатньо додати перевірку рівня освітленості та команду для активації підсвічування.

Найбільш ефективним підходом до розширення функціональності є комбіноване використання всіх трьох сенсорів, що дозволяє створити багаторівневу систему активації. Наприклад, спочатку PIR-сенсор фіксує рух людини в зоні дії, сигналізуючи про її присутність. Далі ультразвуковий датчик HC-SR04 перевіряє, чи людина підійшла на оптимальну відстань. IR-сенсор аналізує рівень освітленості і, за потреби, активує інфрачервоне підсвічування. Лише після виконання всіх цих умов система вмикає камеру і запускає алгоритм розпізнавання обличчя. Такий підхід робить пристрій значно "розумнішим", адже він не реагує на випадкові зміни в середовищі, а також економить ресурси, активуючи камеру лише тоді, коли це дійсно потрібно. Це не лише підвищує енергоефективність, але й зменшує ймовірність помилкових спрацьовувань, що робить систему більш надійною та зручною для користувача.

3.4 Огляд альтернативних моделей Raspberry Pi

Вибір Raspberry Pi 4 Model B як основи для визивного пристрою з розпізнаванням обличчя був обґрунтованим рішенням, адже ця модель забезпечує оптимальний баланс між обчислювальною потужністю, кількістю інтерфейсів і енергоефективністю, що необхідно для реалізації системи з обробкою відеопотоку в реальному часі. Проте в контексті розробки подібних пристроїв важливо не обмежуватися однією моделлю, а розглянути й інші варіанти одноплатних комп'ютерів Raspberry Pi, які можуть бути застосовані залежно від конкретних вимог проєкту. Такий аналіз є не лише академічно доцільним, але й має практичну цінність, оскільки різні моделі можуть краще відповідати потребам у разі обмеженого бюджету, вимог до компактності чи підготовки до серійного виробництва. Розглянемо кілька альтернативних моделей Raspberry Pi, їхні технічні характеристики, переваги та недоліки, щоб зрозуміти, як вони могли б вплинути на архітектуру та функціональність системи.

Першою альтернативою, яку варто розглянути, є Raspberry Pi Zero 2 W. Ця модель є однією з найкомпактніших у лінійці Raspberry Pi, що робить її привабливою для використання в невеликих пристроях, де розмір має вирішальне

значення. Її габарити дозволяють легко інтегрувати її в обмежений простір, наприклад, у корпус встановлений на дверях. Це робить її ідеальним вибором для бюджетних проєктів або для створення прототипів, де потрібно мінімізувати витрати. Модель оснащена 4-ядерним процесором із частотою 1 ГГц і 512 МБ оперативної пам'яті, а також має вбудований Wi-Fi модуль, що дозволяє передавати дані на віддалений сервер без додаткових модулів. Однак ці характеристики також є її основним недоліком: обмежена обчислювальна потужність і невеликий об'єм пам'яті не дозволяють ефективно виконувати складні обчислення, такі як розпізнавання обличчя за допомогою алгоритму LBPH, безпосередньо на пристрої. Наприклад, обробка одного кадру на Raspberry Pi Zero 2 W може займати 2–3 секунди, що не відповідає вимогам реального часу. Крім того, 512 МБ ОЗУ швидко вичерпуються при роботі з OpenCV, що може призводити до зависань або збоїв. Дану модель зображено на рисунку 3.5.



Рисунок 3.5 — Raspberry Pi Zero 2 W

У контексті визивного пристрою Raspberry Pi Zero 2 W могла б бути використана у спрощеній версії системи, де основна обробка даних виконується на віддаленому сервері. Наприклад, пристрій із Zero 2 W може лише захоплювати відеопотік за допомогою камери, передавати його через Wi-Fi на сервер, де відбуватиметься розпізнавання обличчя, а потім отримувати результат для активації замка. Такий підхід зменшує навантаження на сам пристрій, дозволяючи використовувати його обмежені ресурси лише для базових функцій, таких як

активація сенсорів (PIR, HC-SR04) і керування апаратними компонентами (реле, бузер). Однак це потребує стабільного інтернет-з'єднання, що може бути недоліком у віддалених місцях або в умовах нестабільної мережі. Таким чином, Raspberry Pi Zero 2 W є хорошим вибором для пасивних сенсорних систем із віддаленою обробкою, але не підходить для повноцінного локального розпізнавання обличчя, як у поточному проєкті.

Наступною альтернативою є Raspberry Pi 3 Model B+, яка є попереднім поколінням у лінійці Raspberry Pi і досі залишається популярною завдяки своїй доступності та достатній продуктивності для багатьох задач. Ця модель оснащена 4-ядерним процесором із частотою 1.4 ГГц і 1 ГБ оперативної пам'яті, що дозволяє їй виконувати базові завдання розпізнавання обличчя, хоча й із меншою швидкістю порівняно з Raspberry Pi 4. Наприклад, час обробки кадру на Raspberry Pi 3 B+ із алгоритмами Haar Cascade і LBPH становить приблизно 1.5 секунди, що трохи перевищує вимогу реального часу, але все ще може бути прийнятним для систем із невеликою кількістю користувачів. Модель має вбудований Wi-Fi, 4 порти USB 2.0 і 40 GPIO-пінів, що забезпечує достатню кількість інтерфейсів для підключення камери, дисплея, реле та сенсорів. Дану модель зображено на рисунку 3.6.



Рисунок 3.6 — Raspberry Pi 3 Model B+

Raspberry Pi 3 Model B+ є хорошим компромісом між ціною та функціональністю, адже її вартість нижча, ніж у Raspberry Pi 4, але все ще здатна

виконувати основні завдання. Вона могла б бути використана у спрощених системах, де кількість користувачів обмежена (до 10–15 осіб), або у проєктах із помірними вимогами до продуктивності. Наприклад, для невеликого офісу чи приватного будинку, де система не потребує обробки великої кількості кадрів чи складних алгоритмів, Raspberry Pi 3 B+ могла б забезпечити достатню швидкість і стабільність. Однак її стара архітектура (Cortex-A53) і менший об'єм пам'яті обмежують можливості використання сучасних бібліотек, таких як TensorFlow Lite, для глибшого навчання, що могло б покращити точність розпізнавання. Таким чином, Raspberry Pi 3 B+ є хорошим вибором для бюджетних систем із базовою логікою, але поступається Raspberry Pi 4 у продуктивності та гнучкості.

Останньою альтернативою, яку варто розглянути, є Raspberry Pi Compute Module 4. Ця модель була розроблена спеціально для вбудованих рішень і серійного виробництва, що робить її ідеальним варіантом для масштабування визивного пристрою у промислових масштабах. Compute Module 4 має технічні характеристики, аналогічні Raspberry Pi 4 Model B: 4-ядерний процесор Cortex-A72 із частотою до 1.8 ГГц, до 8 ГБ оперативної пам'яті, вбудований Wi-Fi і підтримку кастомних інтерфейсів. Однак, на відміну від стандартної моделі, Compute Module 4 постачається у форматі модуля без периферійних роз'ємів, таких як USB, HDMI чи Ethernet, що дозволяє розробникам створювати власні плати розширення, адаптовані під конкретні потреби проєкту. Це дає змогу оптимізувати систему, залишивши лише ті порти, які дійсно потрібні. Цю модель зображено на рисунку 3.7.



Рисунок 3.7 — Raspberry Pi Compute Module 4

У контексті пристрою Compute Module 4 могла б стати основою для серійного виробництва, адже її компактність і гнучкість дозволяють зменшити розміри кінцевого продукту, оптимізувати схему живлення та знизити собівартість при масовому виробництві. Наприклад, розробивши власну плату розширення, можна було б інтегрувати лише необхідні компоненти, такі як роз'єм для камери, кілька GPIO для підключення сенсорів і реле, а також стабілізатор живлення для роботи від батареї чи адаптера. Це дозволило б створити більш компактний і енергоефективний пристрій, який би відповідав вимогам промислового використання, наприклад, для встановлення в багатоквартирних будинках чи офісних комплексах. Однак використання Compute Module 4 потребує додаткових зусиль на етапі розробки, адже необхідно спроектувати і виготовити власну плату розширення, що може бути складним і дорогим на початковому етапі. Для прототипу, як у поточному проєкті, Raspberry Pi 4 Model B є кращим вибором через готовність до використання "з коробки", але Compute Module 4 відкриває можливості для подальшого масштабування.

Таким чином, вибір моделі залежить від конкретних цілей і умов роботи. Raspberry Pi 4 Model B, обрана для цієї роботи, є оптимальним рішенням для прототипу завдяки своїй продуктивності, великій кількості інтерфейсів і підтримці сучасного програмного забезпечення, що дозволяє реалізувати локальне розпізнавання обличчя.

4 ТЕСТУВАННЯ ТА ЕКСПЕРИМЕНТАЛЬНІ ДОСЛІДЖЕННЯ

4.1 Моделювання реальних сценаріїв використання

Для оцінки роботи визивного пристрою з розпізнаванням обличчя в умовах, наближених до реальних, потрібно провести моделювання типових сценаріїв використання. Це дозволить перевірити, як він справляється з різноманітними викликами, які можуть виникнути в процесі експлуатації, і виявити сильні та слабкі сторони. Тестування проводилося на ноутбучі з вбудованою камерою, що імітувала роботу камери Raspberry Pi, а програмна частина була розгорнута в віртуалізованому середовищі для апробації алгоритмів без залучення фізичних зовнішніх пристроїв на цьому етапі. Усі сценарії були ретельно сплановані, щоб охопити як побутові, так і офісні умови, а також врахувати вплив зовнішніх факторів, таких як освітлення, шум і присутність сторонніх осіб.

Першим сценарієм, який може бути змодельований, буде використання пристрою в домашніх умовах. Уявімо типову ситуацію: власник квартири повертається додому після роботи, підходить до дверей, де встановлено визивний пристрій. Освітлення в під'їзді є середнім (близько 100 люкс), що відповідає стандартним умовам у багатоквартирному будинку вдень. Камера захоплює обличчя користувача з фронтального кута на відстані 50–70 см. У цьому сценарії система має швидко виявити обличчя, розпізнати його за допомогою попередньо тренованої моделі LVRN і надати доступ. Для ускладнення тесту можна додати змінні: користувач змінює міміку (усмішка, нейтральний вираз), а також тест можна провести з різними кутами повороту голови (30° вліво чи вправо). Це дозволить оцінити, наскільки система здатна адаптуватися до природних змін у зовнішності людини під час повсякденного використання.

Другим сценарієм може бути використання пристрою в офісному середовищі. У цьому випадку пристрій імітує роботу системи контролю доступу до офісного приміщення, де кілька співробітників щодня проходять аутентифікацію. Тестування має проводитись в умовах штучного освітлення

(приблизно 300 люкс, що є типовим для офісів), але з додаванням шуму в кадрі. Наприклад, у кадрі могли з'являтися сторонні об'єкти (рука, що випадково закриває частину обличчя, або папка, яку тримає користувач), а також кілька осіб одночасно намагалися пройти автентифікацію, створюючи "переповнення" кадру. Це дозволить перевірити, як система справляється з виявленням і розпізнаванням обличчя в умовах, коли в кадрі є кілька людей, і як вона реагує на часткове перекривання обличчя сторонніми об'єктами.

Третій сценарій має бути орієнтований на роботу пристрою в умовах нічного освітлення. Уявімо, що користувач підходить до дверей у темний час доби, коли рівень освітлення падає до 10 люкс (слабке вуличне світло або світло від ліхтаря). У таких умовах камера без ІЧ-підсвічування не може забезпечити якісне зображення, тому тестування має проводитись з увімкненим підсвічуванням, яке активується за допомогою сенсора освітленості. Камера має захопити чорно-біле зображення, яке обробляється алгоритмом LBRN. У цьому сценарії також має перевірятись стійкість системи до засвічень (наприклад, від ліхтаря, що світить у камеру) і тіней, які могли б утворюватися на обличчі через нерівномірне освітлення.

Четвертим аспектом моделювання має стати перевірка роботи системи при різних кутах зору та відстанях. Користувач підходить до камери під різними кутами (45° в профіль, 30° зверху чи знизу), а також змінює відстань від 30 см до 1.5 м. Це дозволить оцінити, наскільки система здатна виявляти та розпізнавати обличчя, коли людина не стоїть строго фронтально перед камерою, що є частим випадком у реальному житті. Наприклад, уявімо, що людина нахиляє голову, дивлячись на дверний замок, або відступає назад, чекаючи на доступ. Такі сценарії допомогли виявити межі роботи алгоритму Haar Cascade, який використовується для первинного виявлення обличчя.

Останнім сценарієм може стати тестування системи на стійкість до сторонніх облич і шумів. У цьому випадку до камери мають підходити люди, які не були зареєстровані в базі даних (імітація спроби несанкціонованого доступу). Також у кадрі додаються шуми, такі як рухливі об'єкти на фоні (наприклад,

людина, що проходить повз), або статичні елементи (декоративні рослини чи меблі), які могли б бути помилково інтерпретовані як обличчя. Це дозволить перевірити, чи система допускає хибнопозитивні спрацьовування, коли доступ надається неавторизованій особі, і як вона реагує на складний фон.

Моделювання таких сценаріїв у майбутньому дозволить отримати комплексне уявлення про поведінку системи в реальних умовах, враховуючи побутові та офісні ситуації, різні рівні освітлення, кути зору, відстані та зовнішні перешкоди. Результати цих тестів можуть стати основою для подальшого аналізу та оцінки роботи пристрою.

4.2 Методика та умови тестування

Після завершення етапу розробки програмної логіки виникла потреба в оцінці його функціональних можливостей, щоб переконатися в коректності роботи. Зокрема, необхідно було перевірити кілька ключових аспектів: загальну працездатність системи, її точність у розпізнаванні осіб, швидкодію при обробці даних, а також стабільність функціонування в різноманітних умовах експлуатації, які можуть виникати в реальному житті. Однак, через обмеження фізичного середовища та відсутність можливості використовувати повноцінний набір зовнішніх пристроїв на цьому етапі, тестування було організовано у віртуалізованому середовищі. Для цього використано ноутбук із вбудованою камерою, що дозволило зосередитися на апробації програмної частини.

Основною метою тестування стало підтвердження того, що система відповідає заявленим вимогам і здатна ефективно виконувати свої функції в умовах, максимально наближених до реальних. Зокрема, досліджувалися чотири основні аспекти роботи. По-перше, перевірялася здатність виявляти обличчя в реальному часі, тобто забезпечувати оперативне визначення наявності особи в кадрі без значних затримок. По-друге, аналізувалася можливість ідентифікації особи на основі попередньо тренованої моделі, що є ключовим елементом для забезпечення безпеки доступу. По-третє, важливим завданням було переконатися у відсутності хибнопозитивних спрацьовувань, коли система могла б помилково

ідентифікувати сторонню особу як зареєстрованого користувача, що могло б призвести до серйозних ризиків у контексті безпеки. Нарешті, досліджувалася стабільність роботи алгоритмів у змінних умовах зйомки, таких як різне освітлення, кути повороту обличчя чи наявність шумів у кадрі, щоб переконатися, що система залишається надійною навіть за неідеальних обставин.

Для проведення дослідження було підготовлено спеціальний набір даних. Ці набори даних використовувалися для тренування моделі розпізнавання обличчя, для чого було застосовано алгоритм LBPН, реалізований у бібліотеці OpenCV. Цей алгоритм обрано через його ефективність у роботі з відтінками сірого та здатність враховувати локальні особливості обличчя, що робить його придатним для використання в системах із обмеженими обчислювальними ресурсами, таких як Raspberry Pi.

Для відеозахоплення під час тестування використовувалася вбудована камера ноутбука 720. Ця роздільна здатність була обрана як компроміс між якістю зображення та обчислювальним навантаженням, адже вона дозволяє отримати достатньо деталей для розпізнавання обличчя, але водночас не перевантажує систему обробки. Тестування проводилося в стандартному житловому приміщенні. Такий підхід дав змогу імітувати різні умови експлуатації, з якими пристрій може зіткнутися в побуті, наприклад, у коридорі багатоквартирного будинку чи на вході до приватного житла, де освітлення може варіюватися залежно від часу доби чи погодних умов.

У рамках тестування було розроблено два основні типи сценаріїв, які відображають типові ситуації використання системи. Перший сценарій, так званий позитивний тест, передбачав ситуацію, коли до камери підходила особа, чії дані вже були зареєстровані в базі даних. У цьому випадку система мала коректно ідентифікувати користувача та надати доступ, що імітує успішну автентифікацію. Другий сценарій, негативний тест, моделював спробу несанкціонованого доступу: до камери підходила особа, якої немає в базі даних, і система мала відмовити в доступі, не допускаючи хибнопозитивного результату. Таке розмежування сценаріїв дозволило перевірити як точність розпізнавання, так

і безпеку в умовах, коли можливі спроби проникнення.

Для кожної тестової спроби фіксувалися кілька ключових параметрів, які дозволили провести детальний аналіз роботи системи. По-перше, відзначався сам факт розпізнавання: чи вдалося системі виявити обличчя в кадрі. По-друге, оцінювалася правильність результату, тобто чи коректно було ідентифіковано особу (для позитивного тесту) або чи правильно система відмовила в доступі (для негативного тесту). По-третє, вимірювався час обробки кадру, який включав період від моменту захоплення зображення камерою до видачі остаточного рішення про доступ — цей параметр є важливим для оцінки швидкодії системи в реальному часі. Нарешті, у разі виникнення помилок (наприклад, ігнорування обличчя чи хибне розпізнавання) фіксувався їх тип, що допомогло виявити слабкі місця алгоритмів і сформулювати напрямки для подальшого вдосконалення. Такий комплексний підхід до збору даних забезпечив можливість глибокого аналізу всіх аспектів роботи системи.

4.3 Результати тестування та їх аналіз

Усього було проведено 10 тестових спроб, з яких 8 — за участі авторизованих користувачів (тобто осіб, обличчя яких наявні у базі тренування), а 2 — за участі сторонніх (особи, які не були зареєстровані). На рисунку 4.1 зображено успішну спробу аутентифікації на моєму зображенні яке є в базі. Тест проведено при денному освітленні.

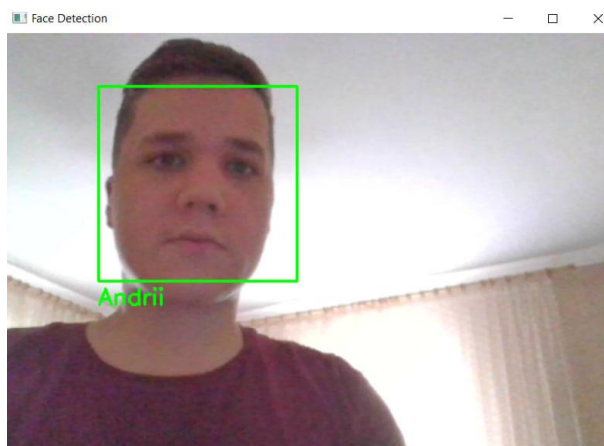


Рисунок 4.1 — Успішна спроба аутентифікації

На рисунку 4.2 зображено невдалу спробу, так як зображення цієї людини немає в базі користувачів.

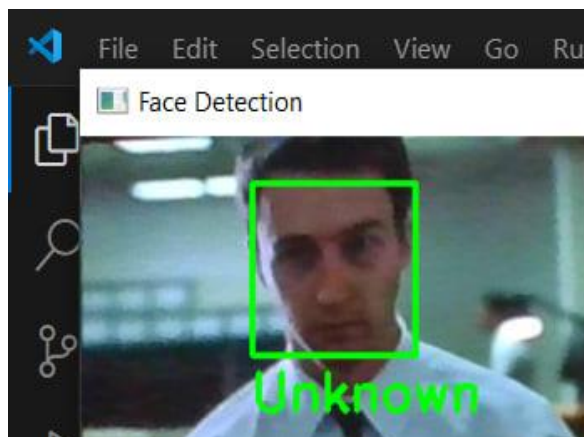


Рисунок 4.2 — Невдала спроба аутентифікації

Проведене тестування розпізнавання обличчя дало змогу сформувати ґрунтовну оцінку його загальної ефективності та рівня надійності в умовах, наближених до реальних. У ході цих досліджень було отримано значний обсяг даних, які дозволили підтвердити низку ключових характеристик, що свідчать про стабільність функціонування алгоритмів у більшості типових ситуацій, з якими пристрій може стикатися в повсякденному використанні. Зокрема, система продемонструвала результати, які підкреслюють її потенціал як надійного інструменту для забезпечення безпеки та зручного доступу.

Серед основних позитивних аспектів, що були виявлені під час тестування, варто відзначити високу точність роботи при розпізнаванні обличч зареєстрованих користувачів. Алгоритм LVRN, який лежить в основі, показав здатність точно ідентифікувати осіб, чиї дані були попередньо введені до бази, навіть за умови незначних змін у позі чи виразі обличчя. Ще однією важливою перевагою стало те, що не допускались неавторизовані користувачі, не зафіксувавши жодного випадку хибнопозитивного спрацьовування протягом усього циклу тестування. Така якість є критичною для забезпечення безпеки, адже вона мінімізує ризик несанкціонованого проникнення. Окрім того, пристрій виявився достатньо швидкодієним, демонструючи середній час реакції, який не перевищував 1 секунди, залежно від конкретних умов навколишнього середовища, таких як

освітлення чи складність кадру.

Разом із тим, у процесі тестування розпізнавання обличчя було виявлено цілу низку типових ситуацій, у яких спостерігалось нестабільне або некоректне функціонування алгоритмів детекції, що вказує на наявність певних обмежень у її поточній реалізації. Ці проблеми стали предметом аналізу, оскільки їх усунення може суттєво підвищити загальну ефективність пристрою. Зокрема, серед найбільш поширених і повторюваних випадків, що викликали труднощі, можна виокремити кілька ключових сценаріїв.

Перш за все, значні проблеми виникали в ситуаціях, коли певна частина обличчя користувача виявлялася частково закритою сторонніми об'єктами, такими як маска, рука, головний убір. У таких випадках система демонструвала зниження точності розпізнавання, що пояснюється фундаментальною вимогою алгоритму до наявності повного доступу до характерних контурів обличчя для коректного спрацювання. Якщо ці контури були відсутні або лише частково видимі через перешкоди, алгоритм або повністю ігнорував обличчя, не вдаючись до ідентифікації, або, що ще гірше, видавав хибні результати, розпізнаючи особу неправильно. Це вказує на необхідність розробки додаткових механізмів, які могли б компенсувати подібні перешкоди, наприклад, шляхом інтеграції альтернативних методів аналізу часткових даних.

Другою значною проблемою стало погіршення ефективності алгоритму при наявності тіней, засвічень або загального дефіциту світла в кадрі. У таких умовах каскад Хаара, який використовується для первинного виявлення обличчя, демонстрував нестабільну поведінку: він або повністю не розпізнавав обличчя, залишаючи кадр без обробки, або помилково інтерпретував кілька об'єктів, таких як тіні чи відбиття, як окремі обличчя. Ця вразливість особливо помітна в умовах нічного освітлення або при різкому контрасті світла, що підкреслює важливість інтеграції сенсорів освітленості та ІЧ-підсвічування для стабілізації роботи системи в різноманітних середовищах.

Третім фактором, який викликав труднощі, було наближення користувача до камери на відстань менше 30 см. У таких випадках через надмірне збільшення

масштабу зображення обличчя користувача перевищувало межі кадру, і детектор не міг зібрати повну інформацію, необхідну для коректного розпізнавання. Це призводило до того, що система не здійснювала ідентифікацію, залишаючи спробу без результату. Така ситуація може виникати, наприклад, коли користувач випадково підходить занадто близько до пристрою, що є досить ймовірним у реальних умовах. Це свідчить про потребу в налаштуванні системи для роботи з динамічними відстанями, можливо, через інтеграцію датчиків відстані чи оптимізацію алгоритму для обробки частково обрізаних зображень.

4.4 Порівняння методів аутентифікації та рекомендації щодо подальших досліджень та коригувань розробки

Для аргументації щодо доцільності обраного методу аутентифікації наведено таблицю 4.1 для порівняння трьох методів.

Таблиця 4.1 — Порівняння методів аутентифікації

Критерій	Набір коду	Відбиток пальця	Розпізнавання обличчя
Зручність	3	4	5
Безконтактність	1	2	5
Швидкість доступу	3	4	5
Безпека	3	4	5
Вартість впровадження	5	3	3
Надійність	4	4	4
Унікальність	1	4	5
Сприйняття користувачем	4	5	5
Середній бал	3	3.75	4.62

Оцінка трьох методів аутентифікації за ключовими критеріями на основі 5-бальної шкали показала, що найвищий середній бал отримав метод розпізнавання обличчя — 4.62. Це свідчить про його переваги в безпеці, безконтактності, зручності та користувацькому сприйнятті. Метод виявився ефективнішим порівняно з кодовим замком (3) та біометрією за відбитком пальця (3.75).

Результати тестування визивного пристрою з розпізнаванням обличчя підтверджують його працездатність і демонструють значну практичну цінність у контексті забезпечення контролю доступу в реальних умовах. Незважаючи на успішну роботу системи в базових сценаріях, виявлені обмеження, такі як зниження точності при слабкому освітленні чи частковому закритті обличчя, а також значний потенціал для розширення функціональності, підкреслюють необхідність подальших досліджень і вдосконалень. Ці аспекти відкривають можливості для глибшого аналізу та практичного розвитку системи, спрямовані на підвищення її надійності, зручності та адаптивності до різноманітних умов експлуатації. Нижче наведено деталізовані рекомендації, які враховують як поточні недоліки, так і перспективні напрямки покращення пристрою.

Перш за все, одним із ключових напрямків розвитку є розширення бази даних користувачів, що дозволить адаптувати систему до більш широкого кола користувачів. Наразі модель LVRH тренується на обмеженій кількості осіб, що може призвести до зниження точності при збільшенні кількості зареєстрованих осіб через перевантаження алгоритму. Для вирішення цього питання пропонується включити до бази даних до 50-100 користувачів, що відповідає типовим потребам офісних приміщень чи багатоквартирних будинків. Важливим аспектом є автоматизація процесу додавання нових користувачів, яка могла б відбуватися через інтуїтивно зрозумілий інтерфейс, наприклад, веб-панель чи мобільний додаток. Такий підхід дозволить користувачу чи адміністратору завантажувати кілька зображень нової особи (наприклад, із різних кутів і при різному освітленні) без необхідності перевчання всієї моделі з нуля. Це можна реалізувати за допомогою інкрементального навчання, де нова інформація інтегрується в існуючу модель, зберігаючи її ефективність і уникаючи значних

обчислювальних витрат.

Ще одним важливим напрямком є покращення якості зображення, що безпосередньо впливає на точність розпізнавання. Поточна система виявляє труднощі в умовах слабого освітлення чи при засвітленнях, що пов'язано з обмеженнями камери. Для подолання цього пропонується інтегрувати камеру з інфрачервоним фільтром, яка здатна працювати в широкому діапазоні освітлення, включаючи повну темряву, за умови використання інфрачервоного підсвічування. Таке підсвічування, наприклад, на основі ІЧ-світлодіодів, могло б активуватися автоматично за сигналом від сенсора освітленості TSL2561, коли рівень освітлення падає нижче 50 люкс. Крім того, додавання поляризаційних фільтрів на камеру допоможе зменшити відблиски від скляних поверхонь чи мокрих ділянок, що часто трапляються в зовнішніх умовах, таких як дощові ночі. Ці зміни не лише підвищать якість зображення, а й зроблять систему більш надійною в екстремальних погодних умовах.

Перехід до глибоких нейронних мереж є перспективним кроком для підвищення точності розпізнавання в складних сценаріях. Поточний алгоритм LBRN демонструє хороші результати в контрольованих умовах, але поступається сучасним методам, таким як FaceNet чи Dlib CNN, у ситуаціях із частковим закриттям обличчя, різними кутами зору чи шумом у кадрі. Альтернативно, Dlib CNN, який є менш вимогливим до ресурсів, може бути адаптований для локальної обробки, що дозволить підвищити точність до 95–98% у порівнянні з 87.5% для LBRN, особливо при роботі з великими базами даних чи в умовах тіней і засвічень.

Створення інтерфейсу адміністратора відкриває нові можливості для зручного керування системою. Замість ручного оновлення бази даних через код можна розробити веб-інтерфейс із використанням технологій Flask або Django, який дозволить адміністратору віддалено переглядати логи доступу, аналізувати спроби аутентифікації (успішні чи невдалі) і додавати чи видаляти користувачів. Наприклад, інтерфейс може відображати графік активності за останні 24 години або деталізований звіт про кожну спробу доступу з позначенням часу, імені

користувача та результату. Мобільний додаток, розроблений на базі React Native, міг би доповнити веб-інтерфейс, надаючи сповіщення в реальному часі на смартфон адміністратора, наприклад, про спроби несанкціонованого доступу. Такий підхід не лише спростить адміністрування, а й підвищить рівень безпеки через миттєву реакцію на потенційні загрози.

Мережева інтеграція є логічним продовженням розвитку системи. Підключення пристрою до локальної мережі через Wi-Fi модуль Raspberry Pi дозволить передавати дані на центральний сервер, де зберігатимуться логи доступу та резервні копії бази даних. У разі підключення до хмарного сервера система зможе надсилати сповіщення на електронну пошту чи месенджер у разі виявлення сторонньої особи. Наприклад, якщо алгоритм фіксує п'ять невдалих спроб доступу за 10 хвилин, сервер може автоматично повідомити власника з фотографіями з камери. Така інтеграція також відкриває можливості для віддаленого оновлення програмного забезпечення, що зробить пристрій більш гнучким і захищеним від уразливостей.

Додаткові методи аутентифікації можуть суттєво підвищити рівень безпеки. Поєднання біометричного розпізнавання обличчя з PIN-кодом, введеним через цифрову клавіатуру, або з NFC-картою створить двофакторну автентифікацію. Уявімо сценарій: користувач підносить NFC-карту до зчитувача, а потім система перевіряє його обличчя. Якщо обидва фактори збігаються, доступ надається. Це особливо актуально для офісів чи об'єктів із підвищеним рівнем безпеки, де необхідна додаткова перевірка. Реалізація такого підходу вимагатиме інтеграції модуля NFC і розробки логіки для комбінованої перевірки, але модульна архітектура пристрою дозволяє легко додати ці компоненти.

Крім того, варто розглянути можливість інтеграції датчиків середовища для адаптації до зовнішніх умов. Наприклад, додавання барометра (BMP180) допоможе відстежувати атмосферний тиск і попереджати про різкі зміни погоди (дощ, сніг), які можуть вплинути на камеру, а датчик температури (DHT22) дасть змогу моніторити умови роботи Raspberry Pi, уникаючи перегріву. Ці дані можуть відображатися в інтерфейсі адміністратора, надаючи повну картину поточного

стану пристрою. Крім того, варто приділити особливу увагу розробці надійної системи резервного живлення, яка здатна автоматично активуватися у разі раптового відключення основного джерела електроенергії. Така інновація відіграє ключову роль у забезпеченні безперервної роботи пристрою навіть у найнесподіваніших ситуаціях, таких як перебої в електропостачанні. Наприклад, можна розглянути використання портативного джерела живлення, такого як потужний акумулятор, обладнаного системою автоматичного переключення, яка миттєво реагує на втрату основного живлення, гарантуючи стабільність функціонування в критичні моменти.

Завдяки модульному підходу до розробки програмного забезпечення, яке необхідне для забезпечення функціонування пристрою, розробка залишається гнучкою та відкритою для майбутніх вдосконалень. Кожен компонент можна оновлювати чи замінювати окремо, що дозволяє поступово додавати нові функції без порушення їх цілісності.

ВИСНОВКИ

У рамках бакалаврської кваліфікаційної роботи було досліджено визивний пристрій з розпізнаванням обличчя. Такий пристрій є сучасною альтернативою традиційним домофонам і охоронним рішенням, оскільки дозволяє здійснювати біометричну ідентифікацію користувача без фізичного контакту та з високим ступенем зручності.

Актуальність даного пристрою була зумовлена потребами безпечного доступу до житлових і промислових приміщень, а також загальним розвитком технологій розумного дому та автономних систем контролю доступу. У роботі було враховано сучасні виклики — потребу в енергоефективності, автономності, невисокій вартості та можливості модульного розширення.

Відповідно до поставленої мети та завдань, у процесі виконання роботи були досягнуті наступні результати:

- проведено аналіз сучасних систем біометричного контролю доступу;
- обґрунтовано вибір апаратних та програмних засобів для реалізації пристрою;
- розроблено структурну та функціональну схему пристрою;
- реалізовано алгоритм розпізнавання обличчя з використанням бібліотек комп'ютерного зору;
- проведено тестування працездатності пристрою.

Отже всі задачі виконані, а мета роботи яка полягає в підвищенні безпеки доступу на об'єкт використовуючи технологію розпізнавання обличчя — досягнута.

Серед ключових переваг розробленої системи:

- відсутність необхідності в постійному з'єднанні з сервером (вся обробка здійснюється локально);
- гнучка та модульна архітектура, що дозволяє змінювати або додавати компоненти без суттєвих змін у коді;
- потенціал до впровадження у розумні будинки, навчальні заклади,

складські приміщення, військові об'єкти тощо.

Перспективи подальшої роботи включають:

- перехід на алгоритми глибокого навчання для підвищення точності;
- розробку мобільного застосунку або веб-інтерфейсу керування доступом;
- впровадження хмарної синхронізації та сповіщень у разі спроб несанкціонованого доступу.

У цілому, пристрій є ефективним, доступним та перспективним рішенням у сфері локального біометричного контролю доступу. Його програмна частина повністю готова до практичного впровадження, а модульна структура відкриває широкі можливості для подальшого вдосконалення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Шпикуляк А.В., Крупельницький Л.В. Визивний пристрій з розпізнаванням обличчя LIV Всеукраїнська науково-технічна конференція Вінницького національного технічного університету (2025). – Режим доступу <https://conferences.vntu.edu.ua/index.php/all-fitki/all-fitki2025/paper/view/24365>
2. Elgendy M. Deep Learning for Vision Systems. – Shelter Island: Manning Publications, 2020. – 480 p.
3. Jain A.K., Hong L., Pankanti S. Biometric identification // Communications of the ACM. – 2000. – Vol. 43, No. 2. – P. 91–98.
4. Царьов Р.Ю., Лемеха Т.М. Біометричні технології: навч. посіб. для вищих навчальних закладів. – Одеса: ОНАЗ ім. О.С. Попова, 2016. – 140 с.
5. Klette R. Concise Computer Vision: An Introduction into Theory and Algorithms. – London: Springer, 2014. – 447 p.
6. Szeliski R. Computer Vision: Algorithms and Applications. – 2nd ed. – Cham: Springer, 2021. – 1232 p.
7. Методи обробки зображень та компютерний зір: навч. посіб. / С.М. Вовк, В.В. Гнатушенко, М.В Бондаренко.-Д.: ЛІРА, 2016. -148с.
8. Gollapudi S. Learn Computer Vision Using OpenCV: With Deep Learning CNNs and RNNs. – 2019. – 151 p.
9. Нікольський Ю.В. Системи штучного інтелекту. Навчальний посібник / Ю.В. Нікольський, В.В. Пасічник, Ю.М. Щербина – Львів, видавництво «Магнолія – 2006», 2010. - 267с.
10. Sonka M. Image Processing, Analysis, and Machine Vision. / M. Sonka, V. Hlavac, Boyler - Stamford: Cengage Learning, 2014.
11. Catsoulis J. Designing Embedded Hardware: Create New Computers and Devices. – Sebastopol, CA: O'Reilly Media, 2002. – 298 p.
12. Petrou M. Image Processing: The Fundamentals, Second Edition / M. Petrou, C. Petrou. - Hoboken, NJ: John Wiley & Sons, 2010. – 832 p.

13. Robert Laganier, Open CV 2 Computer Vision Application Programming Cookbook, Paperback, 2011. – 298 p.
14. Martin R.C. Clean Architecture: A Craftsman's Guide to Software Structure and Design. – Boston: Pearson Education, 2017. – 352 p.
15. Howse J., Minichino J. Learning OpenCV 4 Computer Vision with Python 3. – 3rd ed. – Birmingham: Packt Publishing, 2020. – 372 p.
16. Blokdyk G. Functional Flow Block Diagram Standard Requirements. – 2019. – 227 p.
17. Bradski G., Kaehler A. Learning OpenCV: Computer Vision with the OpenCV Library. – O'Reilly Media, 2008. – 555 p.
18. Lacamera D. Embedded Systems Architecture: Design and Write Software for Embedded Devices to Build Safe and Connected Systems. – 2nd ed. – Packt Publishing, 2023. – 342 p.
19. Проєкти на Raspberry Pi – знайомство з міні-комп'ютером [Електронний ресурс] – Режим доступу: <https://www.sim-networks.com/ukr/kb/best-raspberry-pi-projects>
20. Barr M. Programming Embedded Systems in C and C++. – O'Reilly Media, 1999. – 191 p.

ДОДАТОК А

Технічне завдання

Міністерство освіти і науки України

Вінницький національний технічний університет

Факультет інформаційних технологій та комп'ютерної інженерії

Кафедра обчислювальної техніки

ЗАТВЕРДЖУЮ

Завідувач кафедри ОТ

д.т.н., проф. Азаров О. Д.

_____ 2025 року

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання бакалаврської кваліфікаційної роботи

«Визивний пристрій з розпізнаванням обличчя»

08-54.БДР.037.00.000.ПЗ

Науковий керівник: доцент к.т.н.

_____Крупельницький Л.В.

Студент групи 2СП-216

_____Шпикуляк А.В.

1 Підстави для виконання бакалаврської кваліфікаційної роботи (БКР)

1.1 Актуальність теми обумовлена зростанням запитів на локальні системи контролю доступу з мінімальним людським втручанням, особливо у військовий та післявоєнний час, коли захист об'єктів стає пріоритетом.

1.2 Тема бакалаврської кваліфікаційної роботи затверджена наказом ректора університету №97 від 20.03.2025 року.

2. Мета БКР і призначення розробки

2.1 Метою роботи є розробка та дослідження визивного пристрою з функцією розпізнавання обличчя, здатного працювати автономно та забезпечувати безпечний доступ до об'єкта.

2.2 Розробка пристрою з функцією розпізнавання обличчя, який забезпечує автоматичну ідентифікацію користувача та приймає рішення щодо надання доступу.

3 Вихідні дані для виконання БКР

3.1 Проведення огляду сучасних комерційних, opensource і DIY-систем розпізнавання обличчя.

3.2 Побудова архітектури пристрою та алгоритму взаємодії між компонентами.

3.3 Розрахунок енергоспоживання, вибір оптимальних компонентів, моделювання роботи пристрою.

3.4 Обґрунтування вибору недорогих і доступних апаратних засобів.

3.5 Розгляд державних санітарних правил і норм, що мають дотримуватися при проєктуванні та експлуатації системи.

4 Вимоги до виконання БКР

Проєктування та моделювання пристрою та його програмного забезпечення, які виконують розпізнавання обличчя.

5 Етапи БКР та очікувані результати

Етапи роботи та очікувані результати наведено в таблиці А.1.

Таблиця А.1. — Етапи БКР

№ з/п	Назва етапів дипломної роботи	Строк виконання	Очікувані результати
1	Постановка задачі роботи	21.03.2025	Сформована тема
2	Аналіз предметної області, огляд існуючих рішень та збір літературних джерел	21.03.2025— 28.03.2025	Розділ 1, літературні джерела
3	Обґрунтування вибору апаратного і програмного забезпечення	29.03.2025— 4.04.2025	Розділ 2
4	Розробка структурної, функціональної та алгоритмічної схем системи	5.04.2025— 13.04.2025	Практичний матеріал для розробки пристрою
5	Розробка й тестування програмної реалізації та моделювання роботи схеми	14.04.2025— 27.04.2025	Програма та схема
7	Оформлення пояснювальної записки та ілюстративного матеріалу	28.04.2025— 11.05.2025	Готова пояснювальна записка
8	Перевірка якості виконання бакалаврської кваліфікаційної роботи та усунення недоліків	13.05.2025	Рекомендації щодо поліпшення матеріалу

6 Матеріали, що подаються до захисту БКР

До захисту подаються: пояснювальна записка БКР, графічні та ілюстративні матеріали, протокол попереднього захисту на кафедрі, відгук наукового керівника та рецензія рецензента, анотації до БКР українською та іноземною мовами.

7 Порядок контролю виконання та захисту БКР

Виконання етапів документації БКР контролюється науковим керівником згідно зі встановленими термінами. Захист БКР відбувається на засіданні Екзаменаційної комісії, затвердженої наказом ректора.

8 Вимоги до оформлювання та порядок виконання БКР

8.1 При оформлюванні БКР використовуються:

— ДСТУ 3008: 2015 «Звіти в сфері науки і техніки. Структура та правила оформлювання»;

— ДСТУ 8302: 2015 «Бібліографічні посилання. Загальні положення та правила складання»;

— міждержавний ГОСТ 2.104-2006 «Єдина система конструкторської документації. Основні написи»;

— методичні вказівки до виконання бакалаврських кваліфікаційних робіт зі спеціальності 123 – «Комп'ютерна інженерія» (освітня програма «Системне програмування»). Кафедра обчислювальної техніки ВНТУ 2022.

8.2 Порядок виконання БКР викладено в «Положення про кваліфікаційні роботи на першому (бакалаврському) рівні вищої освіти СУЯ ВНТУ-03.02.02-П.001.01:21».

ДОДАТОК Б

ПРОТОКОЛ ПЕРЕВІРКИ КВАЛІФІКАЦІЙНОЇ РОБОТИ НА НАЯВНІСТЬ ТЕКСТОВИХ ЗАПОЗИЧЕНЬ

Назва роботи:

Визивний пристрій з розпізнаванням обличчя

Тип роботи: Бакалаврська кваліфікаційна робота
(БДР, МКР)

Підрозділ _____ кафедра обчислювальної техніки
(кафедра, факультет)

Показники звіту подібності StrikePlagiarism

Оригінальність	99%	Схожість	1%
----------------	-----	----------	----

Аналіз звіту подібності (відмітити потрібне):

- ✓ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її виконання автором. Роботу направити на розгляд експертної комісії кафедри.
- Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Особа, відповідальна за перевірку _____ Захарченко С.М.
(підпис) (прізвище, ініціали)

Ознайомлені з повним звітом подібності, який був згенерований системою Unichesk щодо роботи.

Автор роботи

(підпис)

Шпикуляк А.В.

(прізвище, ініціали)

Керівник роботи _____ Крупельницький Л.В
(підпис) (прізвище, ініціали)

ДОДАТОК В

Графічна частина

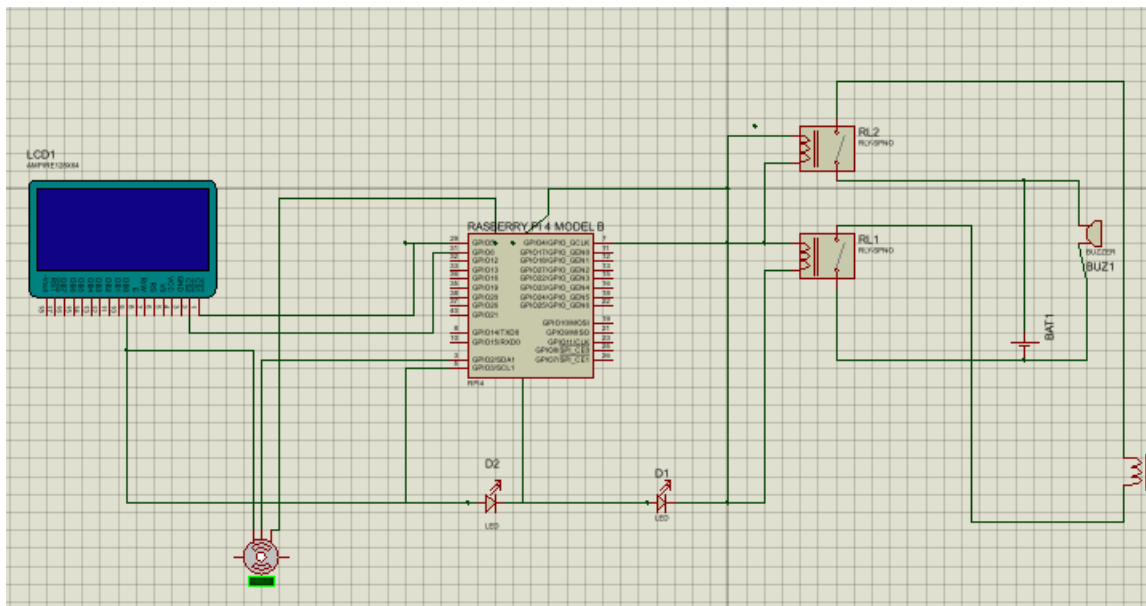


Рисунок В.1 — схема електрично-принципова

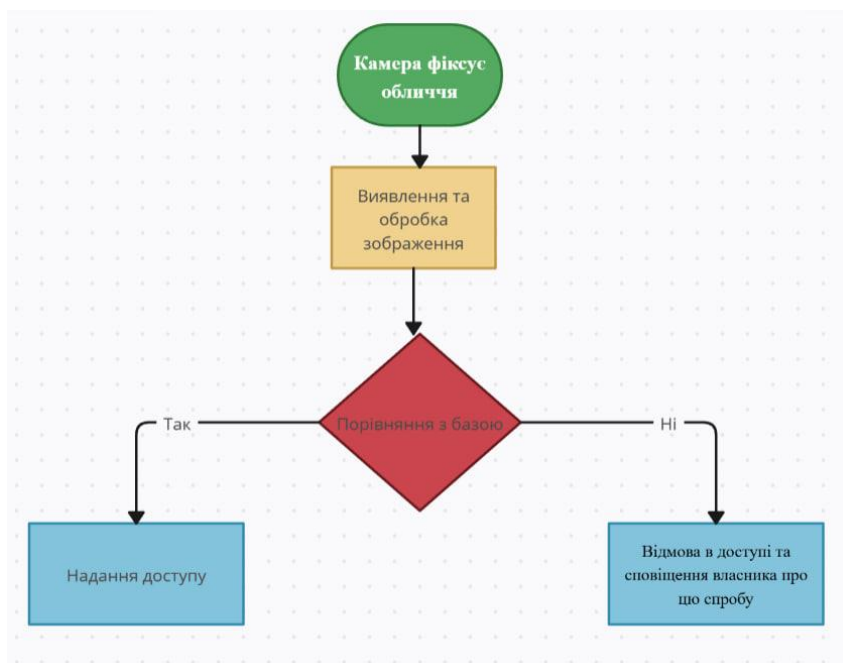


Рисунок В.2 — Блок-схема логіки роботи пристрою

ДОДАТОК Г

Ілюстративна частина

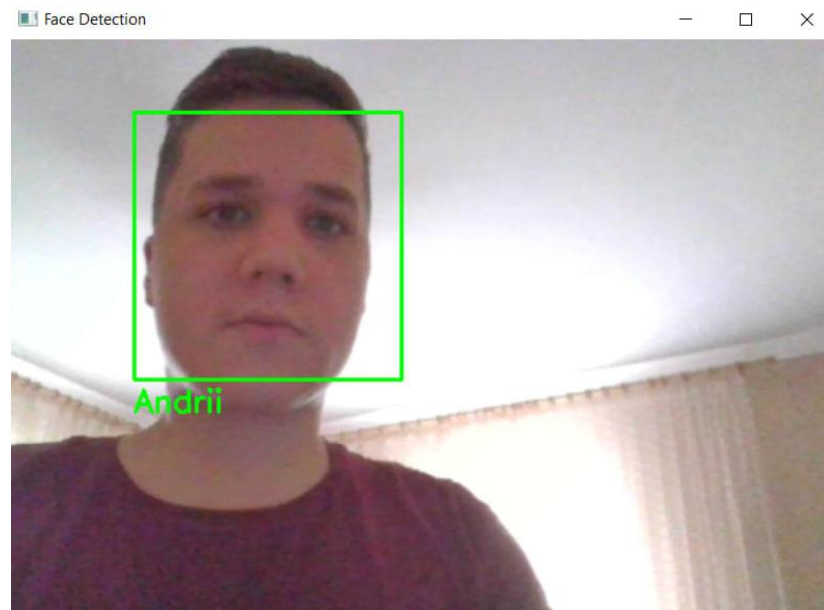


Рисунок Г.1 — Успішна ідентифікація

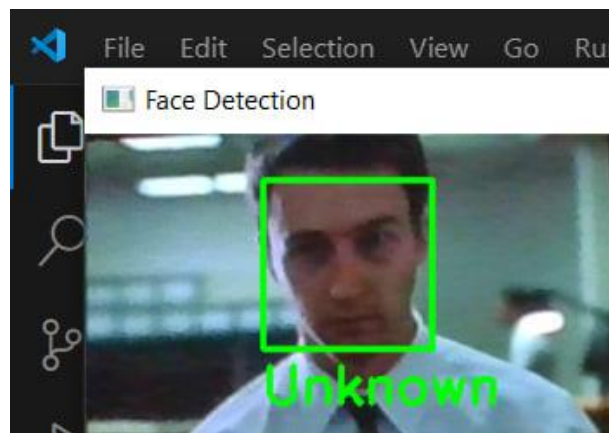


Рисунок Г.2 — Невідома особа

ДОДАТОК Д

Лістинг програмного забезпечення

Лістинг Д.1 — програмний код для тренування моделі на фото користувачів

```
#include <opencv2/opencv.hpp>
#include <opencv2/face.hpp>
#include <filesystem>
#include <iostream>

using namespace std;
using namespace cv;
using namespace cv::face;
namespace fs = std::filesystem;

int main() {
    vector<Mat> images;
    vector<int> labels;

    if (!fs::exists("faces") || fs::is_empty("faces")) {
        cerr << "Папка faces порожня або не існує! Помістіть зображення в папку 'faces'." << endl;
        return -1;
    }

    for (const auto& entry : fs::directory_iterator("faces")) {
        Mat img = imread(entry.path().string(), IMREAD_GRAYSCALE);
        if (img.empty()) {
            cerr << "Не вдалося завантажити зображення: " << entry.path() << endl;
            continue;
        }
    }
}
```

```

resize(img, img, Size(100, 100));
images.push_back(img);

string filename = entry.path().filename().string();
int label = filename[4] - '0';
if (label < 1 || label > 9) {
    cerr << "Некоректна мітка для файлу: " << entry.path() << endl;
    continue;
}
labels.push_back(label);
}

if (images.empty()) {
    cerr << "Немає зображення" << endl;
    return -1;
}

Ptr<LBPHFaceRecognizer> model = LBPHFaceRecognizer::create();
model->train(images, labels);
model->save("trainer.yml");

cout << "Тренування завершено" << endl;
return 0;
}

```

Лістинг Д.2 — Програмний код для роботи пристрою

```

#include <opencv2/opencv.hpp>
#include <opencv2/face.hpp>
#include <wiringPi.h>
#include <iostream>
#include <vector>

```

```

#include <fstream>
#include <unistd.h>
#include <ctime>

#define RELAY_PIN 0
#define BUZZER_PIN 1
#define GREEN_LED 2
#define RED_LED 3

using namespace cv;
using namespace cv::face;
using namespace std;

void logEvent(const string& event) {
    ofstream logFile("log.txt", ios::app);
    if (logFile.is_open()) {
        time_t now = time(0);
        logFile << "Time: " << ctime(&now) << ", Event: " << event << endl;
        logFile.close();
    }
}

bool initCamera(VideoCapture& cap) {
    cap.open(0);
    if (!cap.isOpened()) {
        cerr << "Помилка доступу до камери!" << endl;
        return false;
    }
    cap.set(CAP_PROP_FRAME_WIDTH, 640);
    cap.set(CAP_PROP_FRAME_HEIGHT, 480);
}

```

```

    return true;
}

bool detectFace(Mat& gray, CascadeClassifier& cascade, Rect& faceRect) {
    vector<Rect> faces;
    cascade.detectMultiScale(gray, faces, 1.3, 5);
    if (faces.empty()) return false;
    faceRect = faces[0];
    return true;
}

int extractFeaturesAndPredict(Mat& faceROI, Ptr<LBPHFaceRecognizer>& model,
double& confidence) {
    resize(faceROI, faceROI, Size(100, 100));
    int label = -1;
    model->predict(faceROI, label, confidence);
    return label;
}

bool isUserRecognized(int label, double confidence, string& nameOut) {
    if (confidence < 60) {
        if (label == 1) nameOut = "Андрій";
        else if (label == 2) nameOut = "Інший";
        else nameOut = "Користувач " + to_string(label);
        return true;
    }
    nameOut = "Невідомо";
    return false;
}

```

```

void outputResult(bool recognized, string name, Mat& frame) {
    if (recognized) {
        putText(frame, "Access Granted: " + name, Point(10, 30),
            FONT_HERSHEY_SIMPLEX, 1, Scalar(0, 255, 0), 2);
        digitalWrite(RELAY_PIN, HIGH);
        digitalWrite(GREEN_LED, HIGH);
        digitalWrite(BUZZER_PIN, HIGH);
        delay(3000);
        digitalWrite(RELAY_PIN, LOW);
        digitalWrite(GREEN_LED, LOW);
        digitalWrite(BUZZER_PIN, LOW);
        logEvent("Access Granted for " + name);
    } else {

        putText(frame, "Access Denied", Point(10, 30),
            FONT_HERSHEY_SIMPLEX, 1, Scalar(0, 0, 255), 2);
        digitalWrite(RED_LED, HIGH);
        delay(2000);
        digitalWrite(RED_LED, LOW);
        logEvent("Access Denied");
    }
}

int main() {
    if (wiringPiSetup() == -1) {
        cerr << "Помилка ініціалізації wiringPi!" << endl;
        return -1;
    }
    pinMode(RELAY_PIN, OUTPUT);
    pinMode(BUZZER_PIN, OUTPUT);
}

```

```
pinMode(GREEN_LED, OUTPUT);
pinMode(RED_LED, OUTPUT);
```

```
digitalWrite(RELAY_PIN, LOW);
digitalWrite(BUZZER_PIN, LOW);
digitalWrite(GREEN_LED, LOW);
digitalWrite(RED_LED, LOW);
```

```
VideoCapture cap;
if (!initCamera(cap)) {
    return -1;
}
```

```
CascadeClassifier face_cascade;
if (!face_cascade.load("haarcascade_frontalface_default.xml")) {
    cerr << "Не вдалося завантажити каскад Хаара!" << endl;
    return -1;
}
```

```
Ptr<LBPHFaceRecognizer> model = LBPHFaceRecognizer::create();
try {
    model->read("trainer.yml");
} catch (const Exception& e) {
    cerr << "Помилка завантаження моделі: " << e.what() << endl;
    return -1;
}
```

```
Mat frame, gray;
Rect faceRect;
string userName;
```



```

double confidence;
while (true) {
    cap >> frame;
    if (frame.empty()) {
        cerr << "Не вдалося захопити кадр!" << endl;
        break;
    }
    cvtColor(frame, gray, COLOR_BGR2GRAY);
    if (detectFace(gray, face_cascade, faceRect)) {
        Mat faceROI = gray(faceRect);
        int label = extractFeaturesAndPredict(faceROI, model, confidence);
        bool recognized = isUserRecognized(label, confidence, userName);
        outputResult(recognized, userName, frame);
    }
    imshow("Face Access System", frame);
    if (waitKey(30) == 27) break;
}
cap.release();
destroyAllWindows();
return 0;
}

```