

МІНІСТЕРСТВО ВНУТРІШНІХ СПРАВ УКРАЇНИ
ЛЬВІВСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ ВНУТРІШНІХ СПРАВ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ УПРАВЛІННЯ, ПСИХОЛОГІЇ
ТА БЕЗПЕКИ

Кафедра інформаційних технологій

ІНФОРМАЦІЙНА СИСТЕМА ВІЗУАЛЬНОЇ ІДЕНТИФІКАЦІЇ
ОСОБИ ЗА ЗОБРАЖЕННЯМ ОБЛИЧЧЯ

кваліфікаційна робота
здобувача вищої освіти
4 курсу денної форми навчання
Олега РУДЯ

Науковий керівник:
доцент, кандидат технічних наук
Олег ЗАЧЕК

Рецензент:

вчене звання, науковий ступінь

(Ім'я ПРІЗВИЩЕ рецензента)

Кваліфікаційна робота допущена до захисту

«___» _____ 2025 р., протокол № _____

Завідувач кафедри інформаційних технологій

Ігор ФЕДЧАК

(підпис)

Львів
2025

АНОТАЦІЯ

Рудь О., Зачек О. (керівник). Інформаційна система візуальної ідентифікації особи за зображенням обличчя. Бакалаврська кваліфікаційна робота. Львівський державний університет внутрішніх справ. Львів. 2025.

У кваліфікаційній роботі розроблено інформаційну систему візуальної ідентифікації особи за зображенням обличчя. Дана система орієнтована на використання у сфері безпеки, управління доступом до приміщень, автоматизації процесів обслуговування клієнтів та верифікації користувачів у різних сервісах. Основна увага приділена забезпеченню високої точності розпізнавання облич у реальному часі за умов варіативного освітлення, різних ракурсів обличчя та присутності шуму у зображеннях.

У рамках дослідження виконано глибокий аналіз сучасних методів розпізнавання облич, таких як Eigenfaces, Fisherfaces, Local Binary Patterns (LBP), і нейронних мереж (CNN). На основі аналізу обґрунтовано вибір згорткових нейронних мереж, які забезпечують високу ефективність у складних сценаріях. Для попередньої обробки зображень було застосовано бібліотеку OpenCV, що дозволило зменшити рівень шуму, покращити контрастність і нормалізувати розмір облич.

Інформаційна система складається з декількох модулів: модуль завантаження зображень, модуль попередньої обробки, основний модуль розпізнавання та модуль виводу результатів. Основна обчислювальна частина системи реалізована з використанням Python і фреймворків TensorFlow та PyTorch для навчання та інференсу нейронної мережі. Розроблена система має широкий спектр застосувань, включаючи інтеграцію у системи відеоспостереження, контроль доступу до закритих приміщень, автоматизацію реєстрації користувачів та інші галузі.

Ключові слова: розпізнавання облич, ідентифікація особи, нейронна мережа, OpenCV, TensorFlow, PyTorch, UML-діаграми.

ABSTRACT

Rud' O., Zachek O. (supervisor). The information system for visual identification of a person by face image. Bachelor's thesis. Lviv State University of Internal Affairs. Lviv. 2025.

The qualification work presents the development of an information system for visual identification of a person by face image. This system is designed for use in the fields of security, access management to premises, automation of customer service processes, and user verification across various services. Particular attention is paid to ensuring high accuracy of face recognition in real-time under variable lighting conditions, different face angles, and the presence of noise in the images.

As part of the research, an in-depth analysis of modern face recognition methods was conducted, including Eigenfaces, Fisherfaces, Local Binary Patterns (LBP), and Convolutional Neural Networks (CNN). Based on this analysis, convolutional neural networks were chosen for their high efficiency in complex scenarios. For image preprocessing, the OpenCV library was used, allowing for noise reduction, contrast enhancement, and face size normalization.

The information system consists of several modules: an image loading module, a preprocessing module, a core recognition module, and a results output module. The main computational component of the system is implemented using Python and the TensorFlow and PyTorch frameworks for neural network training and inference. The developed system has a wide range of applications, including integration into video surveillance systems, access control for restricted areas, user registration automation, and other fields.

Keywords: face recognition, person identification, neural network, OpenCV, TensorFlow, PyTorch, UML diagrams.

ЗМІСТ

ВСТУП	6
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	8
1.1. Основні засади дослідження	8
1.2. Аналіз відомих засобів розпізнавання	11
1.3. Аналіз відомих алгоритмів розпізнавання	16
Висновки до першого розділу	20
РОЗДІЛ 2. СТВОРЕННЯ ПРОЄКТУ СИСТЕМИ	21
2.1. Постановка та обґрунтування задачі	21
2.2. Визначення вимог та побудова діаграми варіантів використання	23
2.3. Проєктування системи на логічному рівні	25
Висновки до другого розділу	30
РОЗДІЛ 3. ВИБІР ЗАСОБІВ РОЗРОБЛЕННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ	31
3.1. Вибір методів розв’язання задачі	31
3.2. Вибір та обґрунтування засобів розробки	39
3.3. Аналіз отриманих результатів	43
Висновки до третього розділу	49
ВИСНОВКИ	50
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	52
ДОДАТКИ	55

ВСТУП

Актуальність теми дослідження. У сучасному світі технологія розпізнавання та ідентифікації обличчя викликає значні дискусії серед прихильників захисту персональних даних. Можна стверджувати, що людство перебуває на порозі нової ери тотального стеження, де особа не може бути впевненою у збереженні своєї приватності через повсюдне розповсюдження камер спостереження. Для забезпечення громадського контролю та запобігання неправомірного використання технології ідентифікації важливо розуміти принципи її функціонування.

Аналіз останніх досліджень. Розпізнавання обличчя є практичним напрямком теорії розпізнавання образів, що передбачає автоматичну локалізацію та ідентифікацію осіб за допомогою баз даних. Правоохоронні органи різних країн та соціальні мережі активно впроваджують цю технологію. Провідні технологічні компанії, такі як Google, Apple, Microsoft, Amazon та Facebook, вже запатентували власні розробки у цій сфері [1].

Раніше технологія використовувалась переважно правоохоронними органами для пошуку конкретних осіб у натовпі. Нині вона стала більш доступною та економічно вигідною, що дозволило інтегрувати її в різноманітні комерційні простори: готелі, магазини, банки, вокзали та аеропорти. Серед біометричних методів автентифікації існують технології ідентифікації за голосом, відбитком пальця, ДНК, райдужною оболонкою ока, долонею та рисами обличчя [2, 3]. Остання викликає особливий інтерес через низку переваг порівняно з іншими методами.

Мета та завдання дослідження. Метою дослідження є створення інформаційної системи візуальної ідентифікації особи за зображенням обличчя. Для досягнення мети передбачено виконання таких **завдань**:

- провести аналіз сучасних технологій проектування систем розпізнавання та ідентифікації людей;

- здійснити порівняльний аналіз технічних реалізацій та алгоритмів;
- здійснити проектування системи;
- проаналізувати та вибрати методи для реалізації системи;
- проаналізувати та вибрати засоби для реалізації системи;
- здійснити розробку системи візуальної ідентифікації особи за зображенням обличчя;
- здійснити аналіз контрольного прикладу.

Об'єкт дослідження - процес ідентифікації людини за рисами обличчя.

Предмет дослідження - методи та засоби проектування й реалізації системи розпізнавання та ідентифікації осіб.

Методи дослідження. Під час виконання роботи були використані методи: системного аналізу, що дозволило розбити проєкт на більш дрібні, конкретні завдання та підзавдання, забезпечуючи чіткий план дій. Методи об'єктно-орієнтованого проєктування, які полягали в створенні UML-діаграм що надали чітке відображення взаємодії між користувачем і системою. Це в свою чергу забезпечило високий рівень надійності та масштабованості проєктованої системи для розпізнавання та ідентифікації осіб.

Практична цінність роботи. Полягає у розробленні інформаційної системи візуальної ідентифікації особи за зображенням обличчя. Результатом дослідження є система, яка може бути інтегрована з будь-якою іншою системою або використовуватись як самостійний продукт. Її актуальність підтверджується широким функціоналом та потенціалом практичного застосування.

Структура роботи. Робота містить вступ, три розділи, список використаних джерел, додатки. Повний обсяг основного тексту складає 46 сторінок, кількість ілюстрацій - 30, таблиць - 3, додатків - 2, кількість літературних джерел – 27. Загальний обсяг роботи – 61 сторінка.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Основні засади дослідження

Протягом тисячоліть люди використовували тілесні характеристики, такі як обличчя, голос, хода для взаємного розпізнавання. Альфонс Бертілло, начальник кримінального відділу поліції Парижа, розробив практичний метод використання тілесних вимірювань для ідентифікації злочинців у середині 19-го століття. Його відомість значно зросла після більш вагомого відкриття – розпізнавання людських відбитків пальців наприкінці 19-го століття [4].

Біометрія, яка раніше була переважно інструментом правоохоронної системи для виявлення злочинців, нелегалів, перевірки благонадійності співробітників, нині дедалі частіше застосовується в цивільних сферах для встановлення особи. Біометричні вимірювання мають відповідати низці фундаментальних вимог [5]:

- загальність - характеристика має бути притаманною кожній особі;
- відмінність - різні особи повинні мати помітні відмінності;
- сталість - характеристика залишається незмінною впродовж певного періоду часу;
- вимірюваність - можливість кількісного виміру;
- прийнятність - готовність людей використовувати метод у повсякденному житті;
- захищеність - стійкість до шахрайських маніпуляцій.

Практична біометрична система повинна забезпечувати високу точність розпізнавання, бути безпечною для користувачів та стійкою до різноманітних атак. Розпізнавання обличчя є одним з провідних методів біометричної ідентифікації. У сучасному середовищі мережевого

мультимедійного доступу ця технологія привертає значну увагу. Контроль доступу через розпізнавання обличчя не лише ускладнює незаконне проникнення, а й підвищує зручність взаємодії людини з комп'ютерними системами [6].

Хоча розпізнавання обличчя здавна було метою комп'ютерного зору, лише останнім часом воно стало реальною метою біометричних досліджень, перетворившись на дієвий інструмент ідентифікації особи. У межах дослідження планується комплексний аналіз методів розпізнавання обличчя. Передбачається детальне вивчення існуючих технологічних підходів, їх переваг та обмежень, а також огляд сучасних та перспективних сфер застосування біометричної ідентифікації.

Розпізнавання обличчя є природним процесом для людини, який здійснюється майже без усвідомлених зусиль. Водночас для комп'ютерних систем це надзвичайно складне завдання. Попри двадцятирічні дослідження, науковці лише починають отримувати дійсно корисні технологічні рішення. Як біометрична технологія, автоматизоване розпізнавання обличчя включає низку специфічних властивостей, що потребують глибокого вивчення та практичної апробації. Серед біометричних методів ідентифікації технологія розпізнавання обличчя вирізняється високою точністю та неінвазивністю. З початку 1990-х років вона привертала увагу дослідників з різних галузей: від інформаційної безпеки та психології до комп'ютерного зору та обробки зображень. За цей час було запропоновано численні алгоритми розпізнавання [7].

Мережева безпека та контроль доступу є пріоритетними напрямками впровадження технології розпізнавання обличчя. Її вважають одним з найбільш надійних та результативних методів ідентифікації особи. Проблема розпізнавання обличчя науковці формують як процес ідентифікації індивіда за зображенням його обличчя, що передбачає широкий спектр варіативних підходів.

Особливістю технології є її міждисциплінарний характер. Дослідження охоплюють різноманітні наукові галузі: комп'ютерне розпізнавання образів, біометрію, інформаційну безпеку, мультимедійну обробку, психологію та нейробіологію [8].

Для людини розпізнавання обличчя є настільки буденним процесом, що особа навіть не усвідомлює кількість таких ідентифікацій протягом дня. Хоча дослідження в галузі автоматизованого розпізнавання проводяться з 60-х років, лише нещодавно вони привернули увагу наукового співтовариства. Протягом останнього десятиліття методи аналізу та моделювання обличчя зазнали суттєвих змін. Проте проблема надійності схем розпізнавання особи досі залишається актуальною.

Підробка документів, несанкціонований доступ до фізичних та віртуальних просторів шляхом злому паролів постійно висвітлюються у медіа. Ці виклики сучасного суспільства зумовили нагальну потребу в створенні надійних, зручних та широко прийнятних механізмів контролю ідентифікації особистості. На початкових етапах розвитку технології розпізнавання обличчя дослідники зосереджувалися на автоматичному виявленні окремих рис обличчя. Перевагами геометричних методів була нечутливість до освітлення та інтуїтивне розуміння виокремлених ознак.

Однак навіть сучасні методи виявлення та вимірювання рис обличчя не забезпечують абсолютної надійності. Геометричне розпізнавання елементів обличчя виявилось недостатнім для гарантування високоякісного результату. Через означені обмеження дослідники поступово відмовилися від геометричних підходів на користь цілісних методик. Інноваційні техніки, зокрема Eigenfaces та Fisherfaces, запропонували статистичні та комплексні підходи до розпізнавання [9].

Техніка Eigenfaces базується на статистичному навчанні і становить основу багатьох сучасних схем розпізнавання обличчя. Метод Fisherfaces інтегрує підходи Eigenfaces з лінійним дискримінантним аналізом для оптимізації розпізнавання.

Таким чином, система візуальної ідентифікації особи за зображенням обличчя перебуває в стані активного наукового пошуку та технологічного розвитку, демонструючи значний потенціал для практичного впровадження в різноманітних сферах життєдіяльності сучасного суспільства.

1.2. Аналіз відомих засобів розпізнавання

Система візуальної ідентифікації особи за зображенням обличчя являє собою комплексне технічне рішення, що складається з апаратного та програмного забезпечення. Технологія передбачає введення даних через цифрові зображення або відеокадри, де система здійснює виявлення та локалізацію обличчя для подальшої ідентифікації. Процес розпізнавання базується на аналізі близько 80 характерних параметрів обличчя, серед яких ширина носа, відстань між очима, конфігурація щелепи та інші унікальні біометричні характеристики. Специфікація обличчя формується шляхом внесення цих параметрів до спеціалізованої бази даних як унікального біометричного профілю конкретної особи [10].

Механізм роботи системи передбачає порівняння вхідного зображення з наявною базою фотографій, зроблених під різними кутами. На первинному етапі система визначає положення обличчя, а потім здійснює порівняльний аналіз з еталонними зображеннями в аналогічних позиціях. Розпізнавання обличчя є природним процесом для людини - експериментально доведено, що навіть немовлята здатні розрізняти знайомі обличчя. Протягом багатьох років дослідники працюють над створенням технологій автоматизованої ідентифікації особи, і сучасний ринок пропонує широкий спектр інноваційних рішень.

На сьогоднішній день існує декілька провідних технологій розпізнавання обличчя, серед яких можна виділити чотири найбільш відомі:

NEC NeoFace® Унікальність технології полягає у високій толерантності до зображень низької якості. Система здатна розпізнавати

обличчя навіть з надзвичайно низьким розділенням - до 24 пікселів між точками. Це принципово відрізняє її від інших систем розпізнавання [11].

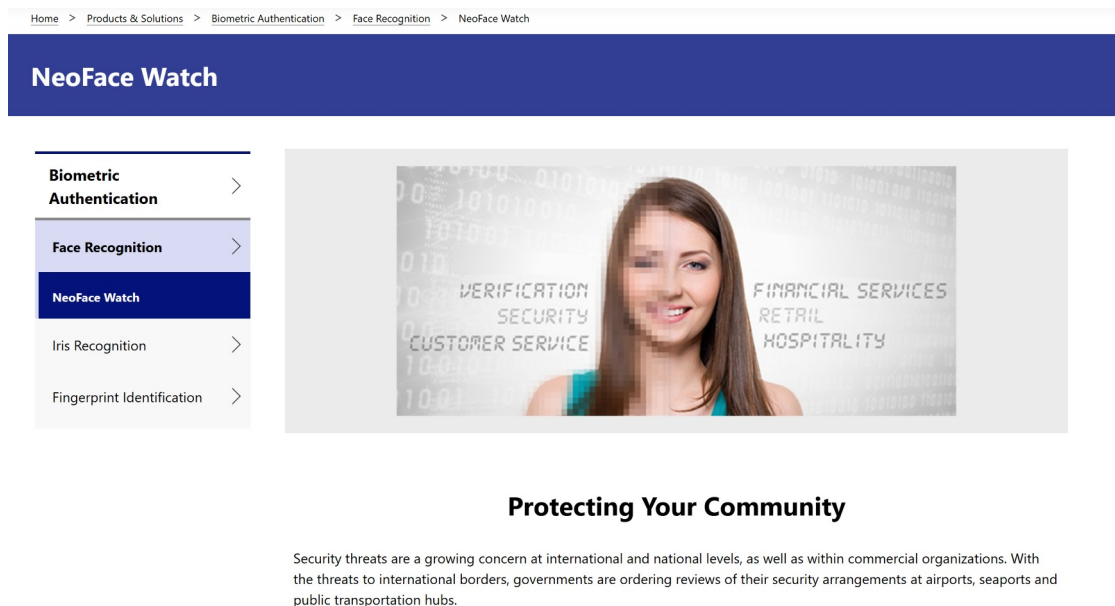


Рис. 1.1 Головне вікно ресурсу NEC NeoFace

Cognitec Systems. Компанія спеціалізується на біометричних технологіях персонального розпізнавання за рисами обличчя. Їх підхід базується на глибоких наукових розробках інженерів та постійному вдосконаленні продуктивності наявних рішень. Додатковою перевагою є надання консультаційних послуг та технічної підтримки [12].

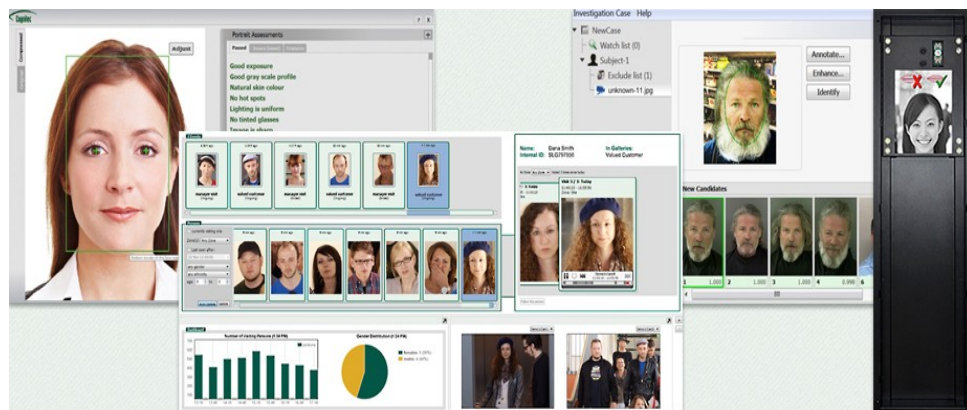


Рис. 1.2. Приклад роботи застосунку Cognitec Systems

FRS SDK. Являє собою комплексний набір бібліотек, що включає: функції обробки зображення (пошук маркерів), інструменти роботи з базою даних, функції порівняння (1:1 та пошук у масиві зображень) та допоміжні інструменти.

Детальний опис функціональних інтерфейсів міститься в інструкції, що поставляється в складі FRS SDK [13].



Рис. 1.3 Приклад роботи застосунку FRS SDK

VeriLook SDK. Технологія орієнтована на розробників біометричних систем та інтеграторів. Має наступні ключові особливості: висока продуктивність розпізнавання в реальному часі, здатність розпізнавати декілька осіб одночасно, підтримка режимів "один-до-одного" та "один-до-багатьох" та сумісність з різними операційними системами [14].

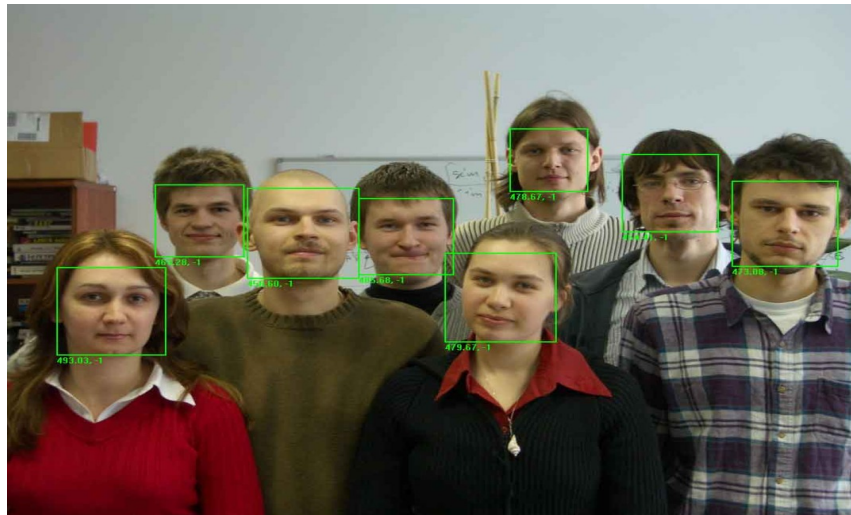


Рис. 1.4. Приклад роботи застосунку VeriLook SDK

На основі проведеного аналізу було сформовано такі критерії порівняння:

- Розпізнавання зображень низької якості полягає у здатності системи ефективно працювати з нечіткими, розмитими або малорозмірними

зображеннями. Даний критерій є критично важливим для реальних сценаріїв використання (відеонагляд, архівні записи).

- Підтримка різних операційних систем полягає у сумісності з різними операційними систмами (Windows, Linux, macOS, мобільними платформами Android та iOS). Забезпечує гнучкість впровадження та інтеграції системи.
- Виявлення гендеру та віку, є додатковою функцією біометричного аналізу, що є корисною для маркетингу, безпеки, соціологічних досліджень.
- Розпізнавання емоцій, визначає здатність визначати психоемоційний стан людини. Зазвичай застосовується в психології, маркетингу, безпеці.
- Робота в режимі реального часу визначає швидкість обробки та розпізнавання зображень. Даний параметр є критично важливим для систем безпеки та контролю доступу.
- Багатопотокова обробка відео передбачає здатність паралельно аналізувати кілька відеопотоків.
- Міжнародна сертифікація передбачає підтвердження відповідності світовим стандартам та визначає гарантії якості та безпеки технології.
- Точність розпізнавання визначає відсоток правильно ідентифікованих осіб. При цьому, вище 95% вважається високоточною системою.
- Захист від підробок передбачає застосування технологій протидії використанню фотографій або масок, що значно підвищує безпеку біометричної системи.

Проведений аналіз систем згідно з визначеними критеріями наведений в таблиці 1.1. Ці критерії дозволяють комплексно оцінити технологічні можливості систем розпізнавання обличчя, враховуючи різні аспекти їхньої функціональності, безпеки та зручності використання.

Таблиця 1.1

Аналіз відомих систем розпізнавання

Критерій	NEC NeoFace®	Cognitec Systems	FRS SDK	VeriLook SDK
Розпізнавання зображень низької якості	+	-	-	+
Підтримка різних операційних систем	+	+	+	+
Виявлення гендеру та віку	+	-	-	+
Розпізнавання емоцій	-	-	-	+
Робота в режимі реального часу	+	+	+	+
Багатопотокова обробка відео	+	+	-	+
Технічна підтримка	+	+	+	+
Міжнародна сертифікація	+	+	-	+
Точність розпізнавання (>95%)	+	+	-	+
Захист від підробок	+	+	-	+

Джерело: складено автором за матеріалами [11-14]

1.3. Аналіз відомих алгоритмів розпізнавання

Система визначення особи за зображенням обличчя включає процес верифікації (автентифікації) на основі якого визначаються категорії легітимних працівників та потенційних зловмисників.

Верифікація (автентифікація). Система перевіряє достовірність особи, яка намагається отримати доступ. Вона аналізує, чи дійсно людина є тією, за кого себе видає. Технологія здійснює пошук "один до одного", порівнюючи пред'явлене зображення з еталонним. При цьому система розділяє суб'єктів на дві групи:

- Легітимні користувачі, які намагаються отримати доступ за власною ідентичністю.
- Потенційні зловмисники, котрі використовують чужу або підроблену особу.

Ефективність верифікації оцінюється двома ключовими показниками: FAR (False Acceptance Rate) - відсоток неправомірно наданого доступу та FRR (False Rejection Rate) - відсоток необґрунтованої відмови в доступі.

Ідентифікація (розпізнавання). Система намагається встановити конкретну особу серед відомих їй суб'єктів. Це класичний сценарій "пошук один до багатьох", притаманний системам відеоспостереження. Технологія працює з припущенням, що всі обличчя в кадрі попередньо внесені до бази даних. Результативність ідентифікації вимірюється такими параметрами: CIR (Correct Identification Rate) - частка правильних ідентифікацій та FIR (False Identification Rate) - частка помилкових ототожнень [15].

Список спостереження - цей режим є складнішою версією ідентифікації, що включає невідомі раніше обличчя. Система не лише шукає збіги в базі, але й оцінює ступінь подібності з раніше не реєстрованими особами.

Зазначені процеси реалізуються з допомогою множини алгоритмів, серед найбільш поширених є наступні.

Алгоритм Eigenfaces. Цей алгоритм ґрунтується на використанні середніх значень, коваріаційних матриць і методу головних компонент. Його особливістю є простота реалізації, яка зазвичай не перевищує кількох десятків рядків коду, водночас забезпечуючи високий рівень ефективності для певних завдань. На етапі навчання використовується вибірка зображень облич, які система має розпізнавати. Ці зображення проектуються в новий простір із більшою кількістю розмірностей, де кожен вимір визначає внесок у загальну структуру даних. Основні компоненти нового простору (власні вектори або eigenfaces) відображають найбільш важливу інформацію, тоді як менш значущі компоненти відсіюються.

Після навчання система може порівнювати нові зображення із проєкціями навчальної вибірки. Якщо зображення знаходиться на значній відстані від усіх еталонних прикладів, воно вважається таким, що не належить до бази даних.

Алгоритм Fisherfaces. Використовує дискримінантний аналіз (LDA) для покращення розпізнавання облич. Головною метою є побудова підпростору, який забезпечує максимальне розділення між класами (різними особами) і мінімізує відмінності всередині одного класу. Цей підхід підходить для класифікаційних завдань, де важливо забезпечити чітке відділення різних груп даних. Fisherfaces ефективний у випадках, коли необхідно враховувати такі варіації, як освітлення або зміни виразів обличчя.

Geometric feature-based recognition (розпізнавання на основі геометричних ознак) – це технологія розпізнавання облич, яка використовує геометричні ознаки, такі як форма та розмір різних частин обличчя, для ідентифікації особи [16]. Для розпізнавання облич на основі геометричних ознак, системи використовують особливості обличчя, такі як відстань між очима, ширина носа, форма брів та губ, а також інші параметри. Ці ознаки використовуються для створення унікального "геометричного шаблону" кожної людини, який може бути використаний для ідентифікації цієї людини.

Image-based recognition (розпізнавання на основі зображень) – це процес розпізнавання об'єктів на зображеннях, який використовується у сфері комп'ютерного зору. Це може бути використано для розпізнавання облич, предметів, логотипів та інших об'єктів на фотографіях, відео або в реальному часі. Методи image-based recognition можуть використовувати різні характеристики зображення, такі як особливості, текстури, контури та інші ознаки для розпізнавання об'єктів. Такі методи можуть використовуватися як у статичних зображеннях, так і в відео [17].

Contour-based recognition (розпізнавання на основі контурів) – це метод розпізнавання об'єктів на зображеннях, який використовує форму та розташування контуру об'єкта для його ідентифікації. Контур може бути визначений як зовнішня межа об'єкта на зображенні, яка відділяє його від фону [18]. В розпізнаванні облич, контур може бути використаний для визначення форми та розташування обличчя на зображенні. Це може включати в себе визначення місцезнаходження очей, носа, рота та інших особливостей обличчя. За допомогою цього методу можна виявити та розпізнати обличчя навіть на фотографіях зі зміненим розміщенням об'єкта або на зображеннях низької якості. Contour-based recognition може бути використаний і для розпізнавання інших об'єктів на зображеннях, таких як автомобілі, будівлі, тварини тощо. Цей метод може бути особливо корисним у випадках, коли об'єкти на зображенні мають спільні риси або форму. Загалом, контур-базовий метод розпізнавання є ефективним методом розпізнавання об'єктів на зображеннях, який може бути використаний в поєднанні з іншими методами

Головних компонент (PCA). PCA є основним інструментом для зменшення розмірності даних і виявлення ключових закономірностей у наборі зображень. Його суть полягає в знаходженні кореляцій між змінними та визначенні нових напрямків (основних компонент), які максимізують дисперсію даних. Основні етапи PCA [19]:

- стандартизувати дані;

- отримання власних векторів і власних значень з коваріаційної матриці або кореляційної матриці, або виконати декомпозицію;
- сортування власних значень в порядку спадання і вибрати k власні вектори, які відповідають k найбільш власним значенням, де k є число вимірювань нового підпростору ($k \leq d$);
- побудувати матрицю проєкції W з обраних k власних векторів;
- перетворення вихідного масиву даних X за допомогою W , щоб отримати k -мірний підпростір функцію Y .

РСА використовується в багатьох областях, включаючи обробку зображень, розпізнавання облич, машинне навчання та аналіз даних. У розпізнаванні облич, RSA може бути використано для витягування характеристик облич із зображення та зменшення розмірності цих характеристик для подальшого використання в алгоритмах класифікації. RSA працює шляхом знаходження головних компонентів набору даних, які відповідають за найбільшу частину дисперсії даних. Головні компоненти визначаються таким чином, щоб вони були ортогональними одна одній і відображали максимальну кількість варіації в даних.

Після знаходження головних компонент, можна зменшити розмірність даних шляхом відкидання менш важливих компонент. Наприклад, у розпізнаванні облич можуть бути використані лише перші декілька головних компонент, що дозволяє зменшити розмірність даних та спростити алгоритми класифікації. Загалом, RSA є потужним інструментом для зменшення розмірності даних та знаходження складних взаємозв'язків між ознаками даних. В розпізнаванні облич, він може бути використаний для покращення точності класифікації та зменшення обсягу даних для обробки.

Сучасні системи дедалі частіше інтегрують алгоритми Eigenfaces, Fisherfaces та RSA. Такі комбіновані підходи дозволяють значно підвищити точність розпізнавання, особливо в умовах обмежених даних або складних сценаріїв, як-от різноманітні кути зйомки, часткове перекриття обличчя або зміни в освітленні. Завдяки цьому системи візуальної ідентифікації стають

більш адаптивними та універсальними для застосування в різних галузях: від безпеки до автоматизації бізнес-процесів.

Висновки до першого розділу

В даному розділі проведений аналіз методологічного підґрунтя дослідження в галузі захисту інформації, зроблено наголос на біометричному методі автентифікації. Розглянуто сучасні системи ідентифікації осіб, перераховано їх сильні та слабкі сторони. Врахувавши отримані результати проведеного аналізу систем, актуальною задачею є розробка інформаційної системи ідентифікації особи за зображенням обличчя, що усувала б недоліки існуючих систем, а також могла бути використана не лише при автентифікації користувачів, але і як продукт, що можна було б інтегрувати в системи безпеки міста.

РОЗДІЛ 2

СТВОРЕННЯ ПРОЄКТУ СИСТЕМИ

2.1. Постановка та обґрунтування задачі

Результатом проведеного дослідження є розробка системи візуальної ідентифікації особи за зображенням обличчя. Ця система забезпечує можливість використання біометричної ідентифікації як способу автентифікації. Крім того, вона може бути ефективно застосована для ідентифікації певної групи осіб. Система повинна демонструвати стійкість до різних зовнішніх факторів, таких як зміни кута освітлення, нахил або поворот голови, недостатня освітленість зображення, наявність макіяжу, неголеність, носіння окулярів тощо. Функціональність системи передбачає точне розпізнавання осіб, якщо вони є у базі даних, або надсилання сповіщень у випадку, якщо ідентифікація виявляє незнайомого користувача.

Розроблена система може бути інтегрована як додатковий модуль для швидкої автентифікації користувачів. Проте важливо врахувати, що у такому випадку доцільніше обмежити доступ до даних лише в режимі читання, оскільки біометричні дані є матеріалом, що може бути сфальсифікованим і практично не підлягає зміні.

Крім того, система може бути використана для ідентифікації певної групи осіб, які становлять інтерес для користувачів системи.

2.1.1. Місце застосування системи.

Система візуальної ідентифікації осіб за зображенням обличчя має широкий спектр застосувань у різних сферах діяльності. Зокрема, її можна використовувати як інтеграційний модуль для автентифікації користувачів і надання доступу до даних. Сучасна технологія ідентифікації за рисами обличчя є однією з найбільш популярних і перспективних, що гарантує її подальший розвиток і поширення в майбутньому.

Цей метод ідентифікації особливо ефективний у закладах, де важливо відстежувати постійних клієнтів для покращення якості обслуговування та підвищення репутації. Наприклад, система може пропонувати гостям ресторанів страви, які відповідають їхнім уподобанням, або інформувати клієнтів готелів про доступність улюблених номерів і послуг.

Крім того, технологія має перспективи у забезпеченні безпеки в місцях масового скупчення людей, таких як банки, казино, аеропорти, вокзали та парки розваг. Вона здатна допомогти в ідентифікації потенційно небезпечних осіб, що значно полегшить роботу служб безпеки, зокрема правоохоронців.

2.1.2. Обґрунтування розроблення та впровадження системи.

Google, Facebook, Apple та інші провідні компанії у сфері ІТ активно інвестують у проекти, пов'язані з технологією розпізнавання облич. Це свідчить про величезний потенціал цієї технології в майбутньому. Однак, через високу комерційну цінність, успішні системи розпізнавання облич давно стали дорогими продуктами, доступними лише для обмеженого кола користувачів. У рамках цієї кваліфікаційної роботи пропонується створити систему, яка не тільки не поступалася б існуючим аналогам, а й перевершувала б їх у деяких аспектах. Передбачається, що система забезпечить швидкий доступ користувачів до інформації, значно спростивши процес автентифікації. Вона позбавить потреби запам'ятовувати складні паролі чи зберігати електронні ключі, що підвищить зручність і ефективність роботи з нею.

Крім того, розроблюване програмне забезпечення буде масштабованим і придатним для використання у різних сферах. Воно може сприяти захисту інформації, забезпеченню безпеки в місті, та виконувати інші важливі функції. Наприклад, система може бути використана в банках для швидкого виявлення шахраїв або інтегрована у "розумні" будинки, полегшуючи повсякденне життя мешканців і прискорюючи виконання рутинних завдань.

2.1.3. Очікувані ефекти від впровадження системи.

Впровадження системи передбачає досягнення таких позитивних результатів:

- Економічний ефект – забезпечення продуктивного та ефективного оброблення даних, посилений захист інформації, а також відповідність очікуванням користувачів, які зацікавлені в інтеграції цього продукту.
- Зростання продуктивності праці – інтеграція розробленої системи у технології, як-от "розумний дім", сприятиме значному підвищенню ефективності роботи й спрощенню виконання щоденних завдань. Загалом система оптимізує та прискорює роботу правоохоронців.
- Організаційний ефект – покращення якості та швидкості обробки інформації завдяки масштабованій архітектурі системи. Правильна конфігурація дозволяє обробляти більший обсяг даних за той самий проміжок часу.
- Технологічний ефект – підвищення рівня автоматизації процесів через застосування передових інноваційних технологій під час розробки та експлуатації системи.

2.2. Визначення вимог та побудова діаграми варіантів використання

Для успішного впровадження та ефективної роботи системи, її вимоги можна розділити на чотири ключові категорії: користувацькі, бізнесові, функціональні та нефункціональні.

1) Користувацькі вимоги включають:

- Простота використання. Інтерфейс системи має бути інтуїтивно зрозумілим, дозволяючи користувачам будь-якого рівня технічної підготовки швидко опанувати її функціонал.
- Швидкість та точність. Система повинна миттєво реагувати на запити, забезпечуючи розпізнавання та ідентифікацію облич із максимально можливою точністю.

2) Бізнесові вимоги включають:

- Безпека та контроль доступу. Забезпечення надійного контролю доступу до об'єктів або ресурсів шляхом ідентифікації осіб із високою достовірністю.
- Інтеграція з існуючими системами. Система має легко інтегруватися з уже наявними компонентами бізнес-інфраструктури, включно з базами даних, системами безпеки та іншими додатками.

3) Функціональні вимоги включають:

- Розпізнавання облич та ідентифікація осіб. Система повинна ідентифікувати обличчя на основі закладених алгоритмів розпізнавання та наявних баз даних.
- Аналіз характеристик облич. Система має виділяти ключові риси обличчя, такі як контури, форма очей, носа, рота тощо.
- Класифікація та контекстуалізація. Система повинна класифікувати об'єкти (наприклад, розпізнавати працівників, клієнтів, відвідувачів) і створювати відповідні контекстуальні описи.

4) Нефункціональні вимоги включають:

- Швидкодія. Система має забезпечувати мінімальні затримки, відповідати вимогам реального часу.
- Масштабованість. Система повинна підтримувати роботу з великим обсягом даних і бути здатною обслуговувати велику кількість користувачів одночасно.
- Надійність. Забезпечення стабільної роботи, мінімізація помилок та швидке відновлення після збоїв.

Додаткові аспекти розробки системи:

- Модульність. Система повинна дозволяти поступове розширення функціоналу завдяки модульній архітектурі.
- Конфіденційність даних. Важливо передбачити механізми захисту біометричних даних відповідно до сучасних стандартів інформаційної безпеки.

На основі моделювання цих вимог була створена UML-діаграма варіантів використання (Use Case Diagram). Вона є візуальним відображенням основних функціональних можливостей системи та взаємодії між користувачами і системою. Діаграма ілюструє ключові сценарії використання, включаючи автентифікацію користувачів, розпізнавання нових осіб, інтеграцію із зовнішніми системами та підтримку реального часу. Це дозволяє зрозуміти, як саме будуть виконуватися завдання та як система відповідає на запити користувачів.

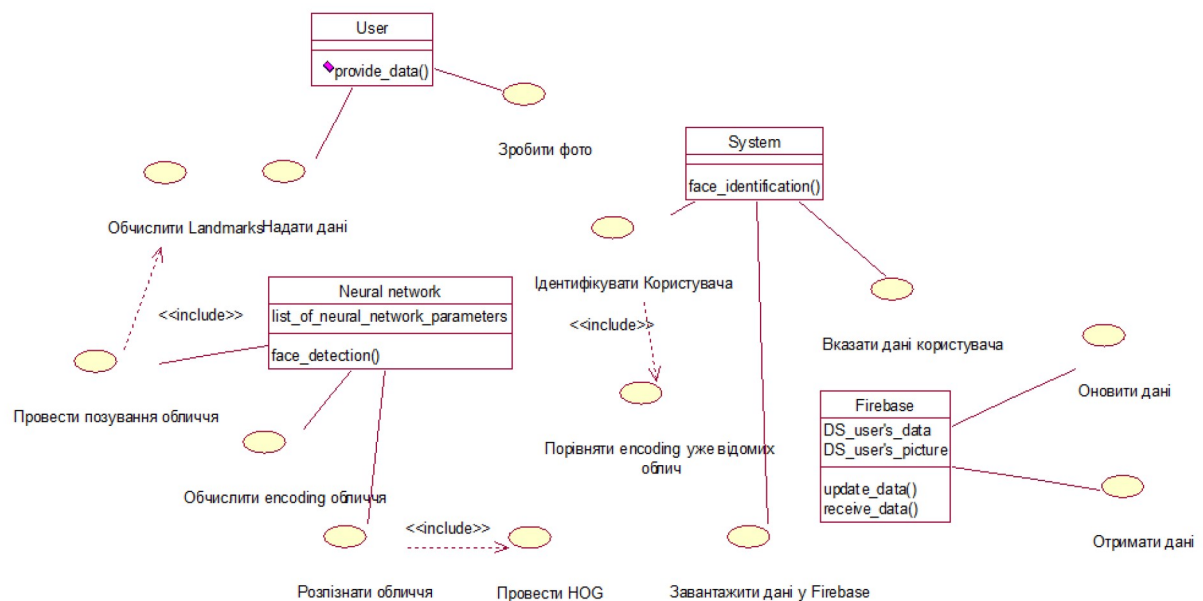


Рис. 2.1. Діаграма варіантів використання

Діаграма містить чотири актори та тринадцять прецедентів, які змодельовані на основі функціональних та користувацьких вимог.

2.3. Проектування системи на логічному рівні

2.3.1 Діаграма класів.

Діаграма класів UML є найбільш поширеним типом діаграм для програмної документації. Оскільки більшість програм, що створюються на сьогоднішній день, все ще засновані на парадигмі об'єктно-орієнтованого програмування, діаграми класів для документування виявляються програмним рішенням. Це відбувається тому, що об'єктно-орієнтоване програмування базується на класах та взаємовідносинах між ними.

Метою діаграми класу є моделювання статичного вигляду програми. Діаграми класів є єдиними діаграмами, які можуть бути безпосередньо пов'язані з об'єктно-орієнтованими мовами, і тому широко використовуються на момент побудови. Діаграма класів в основному є графічним представленням статичного вигляду системи і являє собою різні аспекти застосування. Колекція діаграм класів являє собою всю систему.

У контексті аналізу даних в процесі ідентифікації обличчя в режимі реального часу, можливі об'єкти системи можуть включати:

- Система – основний клас, з якого починається робота програми. Має методи: завантажити дані, завантажити фото, вказати дані користувача.
- Нейронна мережа містить список параметрів розпізнавання обличчя та методи: розпізнати обличчя та ідентифікувати користувача.
- Firebase – сховище даних з атрибутами: БД дані користувачів та БД фото користувачів. Методи: оновити дані та отримати дані.
- Користувацький інтерфейс – інтерфейс системи. Має атрибути: “Ідентифікувати користувача” та “Завантажити дані”, а також метод Зробити фото.
- Користувач – об'єкт, який починає роботу проекту. Містить два методи, а саме: ідентифікувати користувача та надати дані.

Таблиця 2.1.

Опис об'єктів класу

Класи об'єктів		Атрибути класу		Методи класу	
Назва класу	Призначення класу	Назва атрибута	Зміст атрибута	Назва методу	Призначення методу
Система	Основний клас, з якого починається робота програми			Завантажити дані	Завантажує дані у сховище
				Завантажити фото	Завантажує фото у сховище
				Вказати дані користувача	Вказує скільки разів було ідентифіковано користувача

Продовж. табл. 2.1.

Нейронна мережа		Список параметрів нейромережі	Параметри для налаштування нейронної мережі	Розпізнати обличчя	Повертає зашифровані ознаки обличчя
				Ідентифікувати користувача	Порівнює з уже розпізнаними обличчями у Firebase
Firebase	Хмарне зберігання та синхронізація даних в режимі реального часу	дані користувачів	І'мя, прізвище, спеціалізація користувачів та інше	Отримати дані	
		БД фото користувачів	Сховище фото обличчя користувачів	Оновити дані	Оновлюється час останнього ідентифікування користувача
Користувачий інтерфейс	Інтерфейс системи	Кнопка "Ідентифікувати користувача"		Зробити фото	Реалізація механізму фільмування
		Кнопка "Завантажити дані"			
Користувач				Надати дані	Надає персональні дані та фото обличчя
				Ідентифікувати користувача	

Джерело: складено автором за аналізом вимог

Опис відношень між класами представлено в таблиці 2.2, в якій описано класи між якими визначено відношення, вид та розмірність відношення.

Таблиця 2.2.

Опис відношень між класами

Назва відношення	Класи між якими визначено відношення		Вид відношення	Розмірність відношення
Has	System	Neural network	Агрегація	1 – 1
Has	System	User Interface	Агрегація	1 – 1
Contain	System	API Firebase	Агрегація	1 – 1
Upload	System administrator	System	Асоціація	1 – 1
Has	Firebase	API Firebase	Агрегація	1 – 1
Provide	User	System administrator	Асоціація	1 – 1

Джерело: складено автором за аналізом вимог

На основі таблиць 2.1 та 2.2 побудовано діаграму класів (Рис. 2.2).

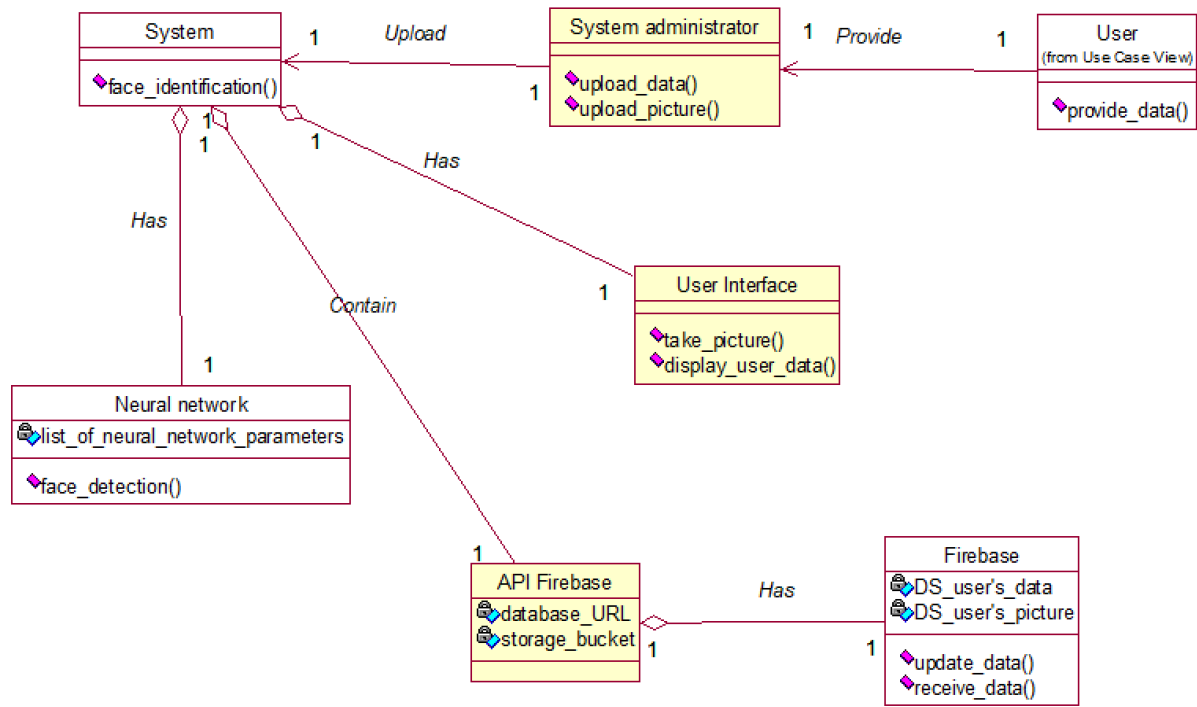


Рис. 2.2 Діаграма класів проєктованої системи

2.3.2. Діаграма діяльності.

Діаграма діяльності демонструє, як процеси виконання певних завдань, які здійснюються системою або зовнішніми сутностями, координуються між собою. Виконання цих завдань може залежати від певних передумов, які повинні бути виконані. Перехід до наступної дії відбувається після завершення попередньої. Як правило, дія отримує на вході певну інформацію у вигляді множини сигналів, які необхідно опрацювати та перетворити на множину вихідних сигналів. На рис. 2.3 представлено діаграму діяльності системи візуальної ідентифікації особи за зображенням обличчя.

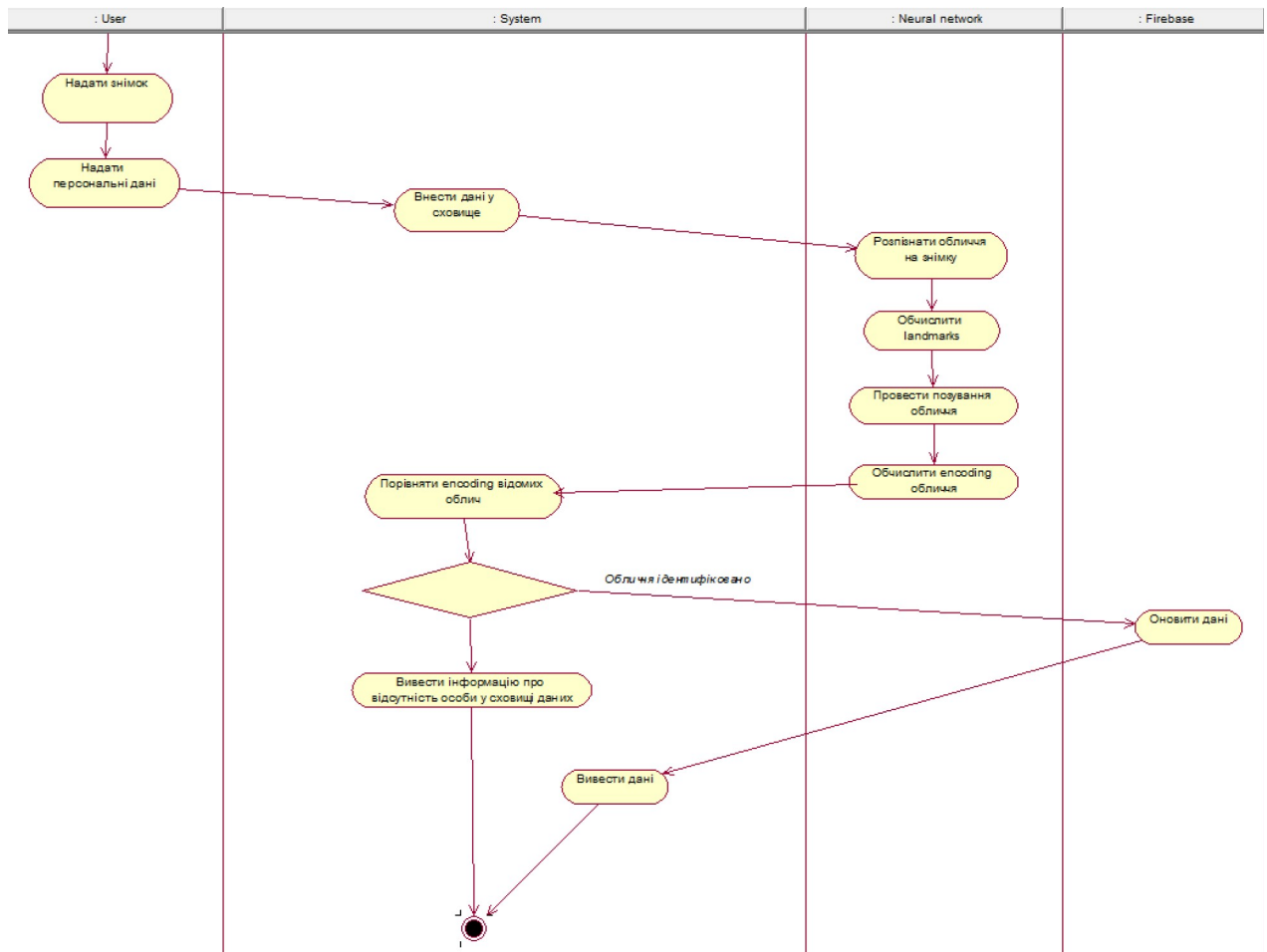


Рис.2.3. Діаграма діяльності

2.3.2 Діаграма розгортання

Діаграма розгортання це діаграма на якій відображаються вузли під час роботи програми. Містять компоненти та об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Компоненти, що не мають представлення під час роботи

програми на таких діаграмах не відображаються. Діаграма розгортання складається з таких елементів як робоча станція та веб-сервер (Рис.2.4.)

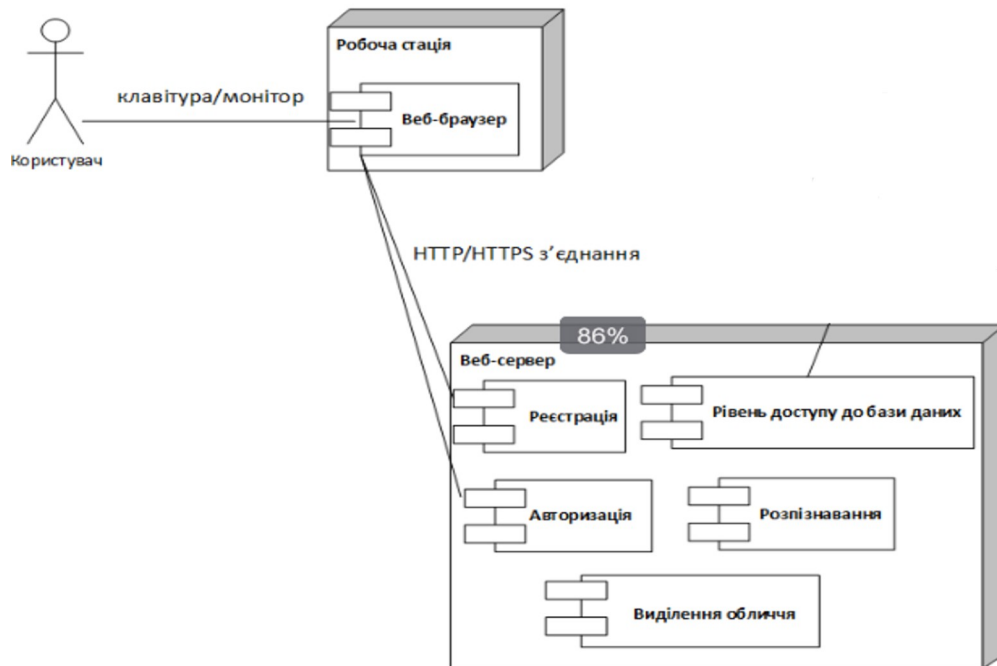


Рис.2.4. Діаграма розгортання

Робоча станція містить такий компонент як веб-браузер, через який буде здійснюватися робота з системою. На веб-сервері знаходяться такі компоненти як: підсистема авторизації, підсистема реєстрації, підсистема виділення та розпізнавання обличчя.

Висновки до другого розділу

У даному розділі було визначено особливості предметної області для побудови системи візуальної ідентифікації особи за зображенням обличчя. Для вищезгаданої предметної області було проведено концептуальний аналіз та проєктування. Аналіз системи допоміг визначити головні компоненти та, методом декомпозиції, спростити в розумінні та виконанні розроблювану систему. Також було побудовано концептуальну модель предметної області, що включала в себе опис та побудову UML діаграм.

РОЗДІЛ 3

ВИБІР ЗАСОБІВ РОЗРОБЛЕННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ

3.1. Вибір методів розв’язання задачі

Загальна схема процесу ідентифікації користувача за зображенням обличчя наведено на рис.3.1.

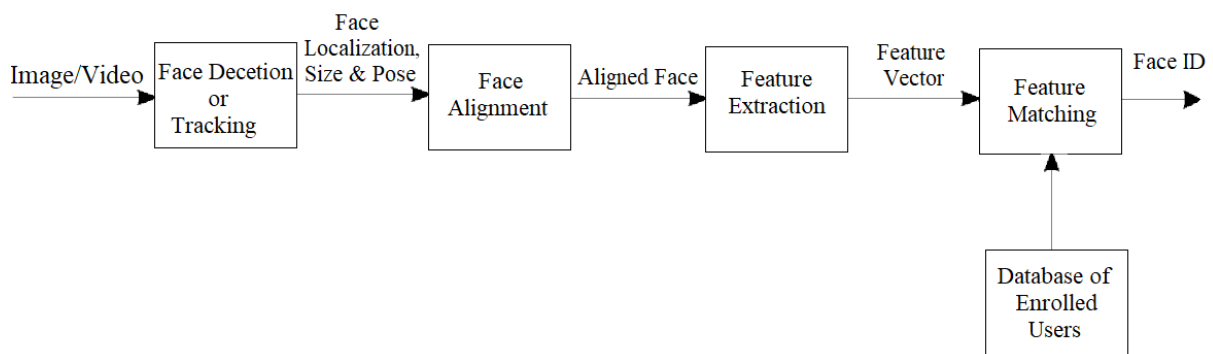


Рис.3.1. Процес розпізнавання обличчя [20]

На першому етапі проходить локалізація обличчя на зображенні. На етапі розпізнавання відбувається геометричне та експозиційне вирівнювання зображення обличчя, знаходження унікальних ознак та безпосередньо розпізнавання - порівняння розрахованих результатів з еталонами, що наявні в базі. Основними відмінностями представлених нижче алгоритмів буде пошук та виділення унікальних ознак, а також порівняння сукупностей цих ознак між собою.

3.1.1. Метод гнучкого порівняння на графах.

Метод гнучкого порівняння графів для ідентифікації облич базується на зіставленні графів, що представляють зображення обличчя. Зазвичай обличчя описуються графами з зваженими вершинами та ребрами. Існує два основних графи: еталонний, який залишається незмінним, і деформований граф, що намагається максимально точно збігтися з еталонним. Графи можуть бути представлені як прямокутні решітки або структури, що

складаються з антропометричних точок обличчя, які розташовуються в певних місцях (Рис.3.2) [21].

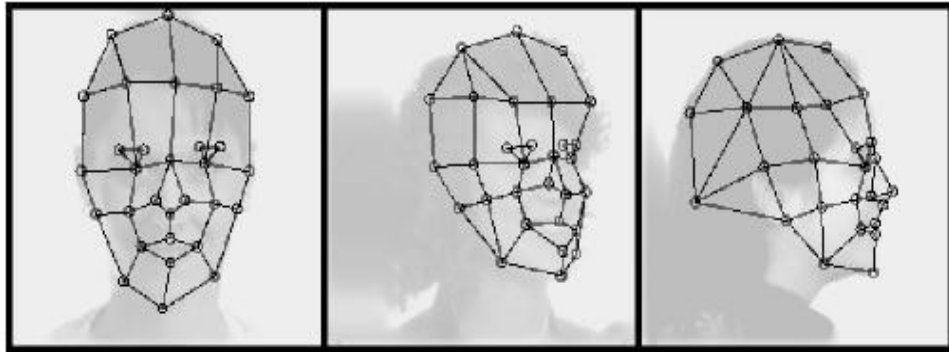


Рис.3.2. Граф, що складається з антропометричних точок обличчя

Математичний розрахунок унікальних ознак обличчя здійснюється у вершинах графу. Різниця між двома графами визначається через ступінь деформації ребер і результати обчислень у вершинах. Процес трансформації графу полягає в поступовому зміщенні кожної вершини в певному напрямку, що дозволяє максимально наблизити її до еталонного графу. Після того як для певної вершини знаходиться оптимальна точка, алгоритм повторюється для наступних вершин, поки різниця між графами не досягне мінімального значення.

Результатом цієї процедури є визначення ступеня схожості між вхідним графом і еталонним, що дає змогу ідентифікувати особу. Той граф, для якого функція деформації дає найменше значення, і є результатом розпізнавання обличчя.



Рис.3.3. Процес деформації вершин та ребер графу

Метод має точність 95%, однак для досягнення цієї точності важливо, щоб кут нахилу обличчя на зображенні не перевищував 15 градусів. Недоліком методу є його висока обчислювальна складність, яка зростає зі

збільшенням бази даних, що може знижувати ефективність системи в умовах великих обсягів інформації.

3.1.2. Нейронні мережі.

Нейронні мережі стали основою для більшості сучасних систем розпізнавання облич, що активно використовуються в світі. Лідери на глобальному ринку таких технологій побудовані саме на нейронних мережах. У задачах класифікації часто застосовуються повнозв'язні нейронні мережі, що складаються з кількох шарів, кожен з яких взаємодіє з усіма нейронами попереднього шару. Для створення цих зв'язків та налаштування ваг нейронів, мережа повинна пройти етап навчання. Процес навчання здійснюється за допомогою навчальної вибірки, де подаються зображення разом з відповідними мітками, наприклад, "кіт" або "пес" для класифікації. Під час навчання нейронна мережа виділяє характерні особливості для кожного класу та будує зв'язки між ними, що дозволяє потім застосовувати набуті знання до нових, невідомих зображень.

Найефективнішим методом для ідентифікації обличчя є нейронна мережа зі згорткою. Особливість цих мереж полягає в їхній архітектурі, яка дозволяє накладати фільтри на зображення, оцінюючи рівень активації кожного з них і вибираючи максимальні відхилення. Це значно зменшує обсяг оброблюваної інформації, що покращує ефективність [22].

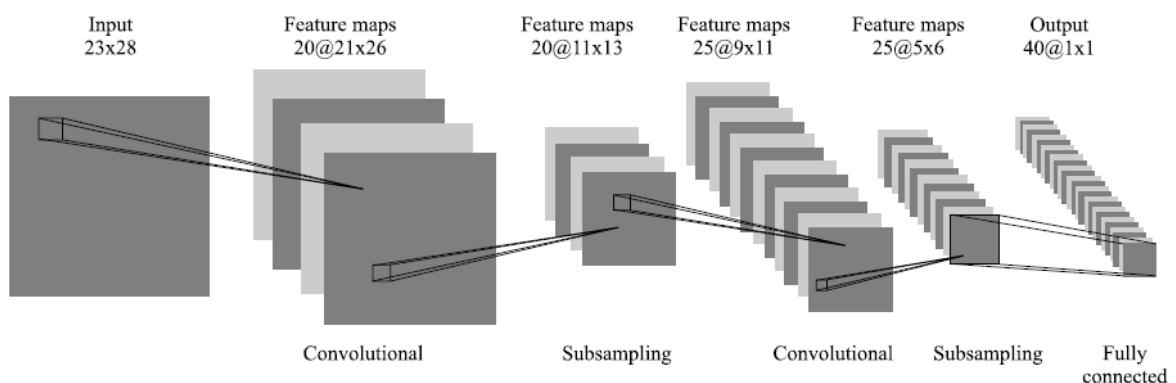


Рис.3.4. Архітектура конволютивної нейронної мережі

Дослідження показали, що нейронні мережі зі згорткою практично не залежать від рівня освітленості, масштабу або кута нахилу обличчя, а точність цього методу складає 96%.

3.1.3. Active Appearance Models (AAM).

Активна модель зовнішнього вигляду є алгоритмом комп'ютерного зору, який дозволяє адаптувати статистичну модель форми та вигляду об'єкта до вхідного зображення. Моделі формуються під час навчання, коли набори зображень разом із координатами точок, присутніх на кожному зображенні, надаються керівнику навчального процесу. Розмітка зображень здійснюється вручну, а кожна мітка вказує на конкретну точку, яку модель повинна знайти під час адаптації до нового зображення. Алгоритм використовує різницю між поточною оцінкою та цільовим зображенням для керування процесом оптимізації. Спочатку активні моделі зовнішнього вигляду використовувалися лише для оцінки параметрів зображень облич. Як згадувалося раніше, такі моделі мають два типи параметрів: пов'язані з формою та описують статистичну модель пікселя або текстури [23].

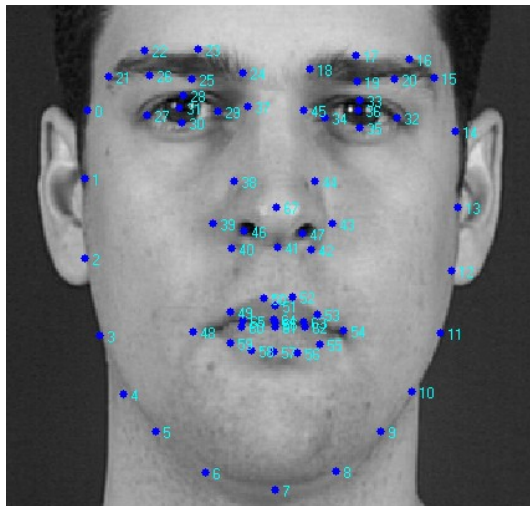


Рис.3.6. Активна модель зовнішнього вигляду, містить 68 точок

Під час навчання активної моделі відбувається нормалізація форм на зображенні, що були позначені мітками, щоб привести обличчя на зображенні до оптимальних умов, коли освітлення є найкращим, а людина дивиться прямо в камеру (Рис.3.7).



Рис.3.7. Координати точок форми обличчя до та після нормалізації

Після нормалізації вибираються основні параметри з множини точок за допомогою методу головних компонент.

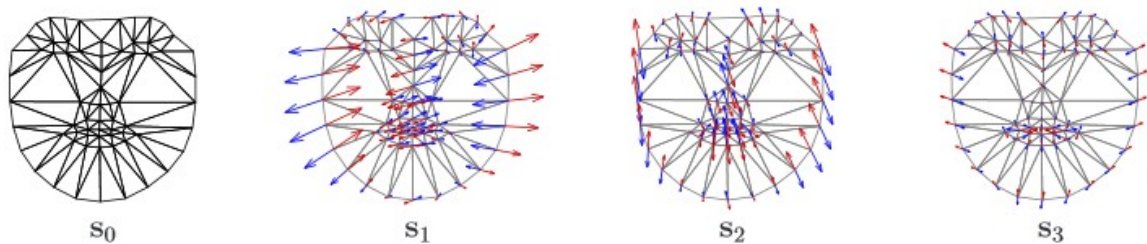


Рис.3.8. Процес зміщення точок відносно триангуляційної решітки (S_0)

Далі формується матриця, що складається з значень пікселів текстури (фільтру), які розраховуються з пікселів всередині трикутників, вершинами яких є точки форми зображення. Фільтри, що використовуються під час навчання, можуть бути як одноканальними (наприклад, градація сірого), так і багатоканальними (RGB). У разі багатоканальних текстур вектори зображення поділяються на кілька окремих зображень для кожного каналу, а потім конкатенуються. Результатом навчання є виявлення головних компонент матриці текстур.

Підсумкова модель обличчя складається з моделі форми та текстури, де перетворення моделі здійснюються під час оптимізації, яка мінімізує функцію помилки методом градієнтного спуску. Градієнтний спуск – це метод пошуку мінімуму функції помилки [24].



Рис.3.9. Процес підгонки моделі за допомогою градієнтного спуску

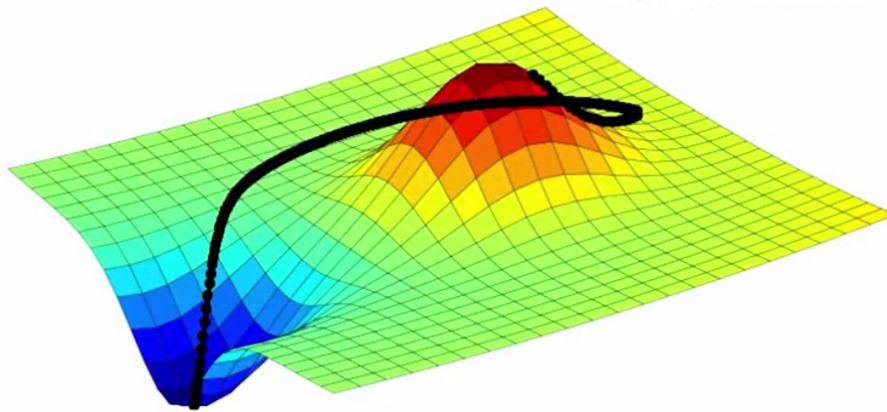


Рис.3.10. Знаходження мінімуму методом градієнтного спуску

Як зазначалося раніше, модель включає параметри, що описують форму обличчя та текстуру, що дозволяє моделювати зображення з різними деформаціями, такими як повороти, масштабування, корекція яскравості чи зміщення оригінального зображення. Для вирішення задачі локалізації обличчя виконується пошук характеристик активної моделі, що створюють синтетичне зображення, яке найбільше відповідає вхідному. За різницею між активною моделлю та вхідним зображенням приймається рішення про наявність або відсутність обличчя.

3.1.4. Active Shape Models (ASM).

Основна мета методу активних моделей форм (AAM) полягає в обчисленні статистичних зв'язків між розташуванням антропометричних точок на зображеннях облич, знятих у фронтальній проекції. На вибірці таких зображень супервайзер повинен позначити всі антропометричні точки, при цьому вони повинні мати однакову послідовність і нумерацію на кожному зображенні.

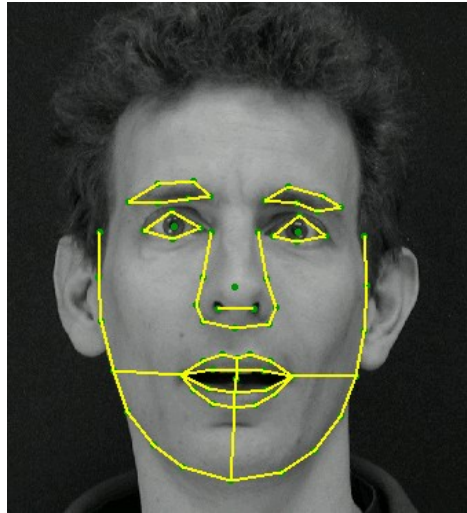


Рис.3.11. Маркування антропогенних точок обличчя

Для узагальнення точок на зображеннях і приведення їх до єдиної системи з одним масштабом використовують прокрустовий аналіз. Це дозволяє розрахувати середню форму та матрицю коваріацій для кожного набору зображень. Власні вектори кожного зображення обчислюються на основі матриці коваріантності, після чого вони сортуються за зменшенням власних значень. Локалізація обличчя на невідомому зображенні за допомогою моделі активних форм здійснюється через розв'язання оптимізаційної задачі [25].



Рис.3.12. Процес локалізації моделі ASM

Проте, як вже зазначалося, основною метою методів AAM та ASM є не стільки розпізнавання облич, скільки локалізація антропометричних точок на зображенні для подальшої обробки. Одним з важливих етапів обробки зображення є його нормалізація, що включає вирівнювання обличчя відносно камери, налаштування оптимального освітлення, зменшення шумів тощо. Першим кроком цього процесу є локалізація обличчя на зображенні, тобто визначення антропометричних точок. Існує багато груп таких точок, але для

зменшення обчислювальних витрат та часу на ідентифікацію зазвичай використовують не більше десяти точок. У багатьох комерційних системах розпізнавання обличч застосовуються моделі AAM та ASM для локалізації цих точок.

Однією з головних проблем при розробці систем ідентифікації користувачів є недостатнє освітлення обличчя або нерівномірне падіння світла, що може спотворити зображення (як показано на рис. 3.13), а також розташування голови (кут нахилу чи повороту).



Рис.3.13. Проблема освітленості



Рис.3.14. Проблема кута нахилу обличчя

З огляду на вищесказане, можна зробити висновок, що для підвищення якості існуючих систем доцільно застосовувати гібридні методи, які поєднують переваги й мінімізують недоліки описаних підходів. Тому для

власної реалізації системи ідентифікації обличчя можна використовувати кілька сучасних методів, які демонструють високі результати. В результаті було обрано нейронні мережі зі згорткою та методи активних зовнішніх моделей і моделей форм, що дозволить при реєстрації користувача швидко побудувати хеш-функцію, що задає шаблон обличчя на основі одного зображення. Це також дозволить коректно ідентифікувати користувача на новому зображенні.

3.2. Вибір та обґрунтування засобів розробки

При виборі мови програмування важливо врахувати кілька чинників. Один з них – популярність мови, оскільки чим вона популярніша, тим більше рішень, бібліотек та документації можна знайти в інтернеті. Однак не варто забувати, що популярність не завжди є показником якості. Часто на зміну популярним мовам приходять нові, більш досконалі та перспективні. У таких умовах конкуренція відіграє ключову роль, надаючи програмістам можливість зробити правильний вибір.

3.2.1 Фреймворки комп'ютерного зору

Vision Framework. Vision – це фреймворк, який забезпечує високопродуктивний аналіз зображень і комп'ютерний зір для обробки зображень та відео. Він автоматично виявляє обличчя, особливості та класифікує сцени, а також здатний розпізнавати штрих-коди, текст, функції та відстежувати об'єкти. Це потужний інструмент для сортування і фільтрації великих обсягів зображень та відео. Використовуючи Core ML і глибоке навчання, Vision забезпечує високу точність і меншу ймовірність помилок, ніж попередні технології. Завдяки цьому фреймворк здатен точно виявляти обличчя навіть в окулярах, шапках і шарфах, а також при частковому перекритті або під сильним кутом. Основні можливості Vision: визначення обличчя та його частин (очі, брови, ніс, губи); розпізнавання тексту та

штрих-кодів; визначення прямокутних поверхонь; виявлення кута горизонту на зображенні та відстеження об'єктів на відео.

Dlib – це набір інструментів створений на C++, який містить алгоритми машинного навчання та засоби для розробки складного програмного забезпечення. Він використовується в різних галузях, від робототехніки до вбудованих пристроїв і мобільних телефонів, і забезпечує високу продуктивність. Основним принципом Dlib є портативність і простота використання, тому весь код розроблений так, щоб бути максимально універсальним, без необхідності в налаштуваннях. Бібліотека працює на різних операційних системах, включаючи ОС X, MS Windows, Linux, Solaris, BSD та HP-UX.

OpenCV (Open Source Computer Vision Library) — це бібліотека з відкритим кодом для комп'ютерного зору та машинного навчання. OpenCV розроблено для надання інфраструктури комп'ютерного зору і прискорення машинного сприйняття в комерційних продуктах. Бібліотека містить понад 2500 оптимізованих алгоритмів, що охоплюють як класичні, так і сучасні методи комп'ютерного зору і машинного навчання. Вона використовується для виявлення облич, розпізнавання об'єктів, класифікації дій на відео, відстеження руху, створення 3D-моделей об'єктів, зшивання зображень для отримання високої роздільної здатності та багато іншого. OpenCV підтримує інтерфейси для C++, Python, Java та MATLAB і працює на Windows, Linux, Android і Mac OS. Бібліотека активно підтримує реалізацію алгоритмів на CUDA та OpenCL і використовується в реальному часі [26].

3.2.2. Система керування базами даних

Для створення баз даних для системи аналізу даних в процесі ідентифікації обличчя в реальному часі можна використовувати сучасні технології та платформи, що забезпечують високу продуктивність і масштабованість. Одним із таких рішень є Firebase, який надає інструмент Firebase Realtime Database. Це документно-орієнтована база даних, яка

дозволяє синхронізувати дані між пристроями та клієнтами в режимі реального часу. База працює на основі концепції "спостерігача" (observer pattern), що дозволяє автоматично оновлювати дані на всіх підключених пристроях при їх зміні.

Основними перевагами Firebase Realtime Database є її простота у використанні, зручність для зберігання та отримання даних у реальному часі за допомогою API. Вона підтримує роботу на різних платформах, таких як веб, мобільні пристрої та сервери. Крім того, Firebase надає інфраструктуру для розробки додатків, що включає бази даних, аутентифікацію, зберігання файлів, хостинг та аналітику. Всі ці сервіси працюють на хмарній інфраструктурі Google, що дозволяє легко масштабувати додатки без необхідності керувати фізичними серверами.

3.2.3. Мова програмування

Python є однією з найбільш популярних мов програмування завдяки своїй простоті, гнучкості та широкому використанню в наукових дослідженнях, аналізі даних і комп'ютерному зорі. Однією з головних причин популярності Python є велика кількість бібліотек і фреймворків, що значно полегшують розробку складних систем, таких як системи розпізнавання та ідентифікації обличчя [27].

Важливі бібліотеки для комп'ютерного зору:

- **dlib** – бібліотека, яка пропонує потужні інструменти для розпізнавання обличчя, включаючи детектори облич, а також алгоритми для трекінгу та аналізу рис обличчя.
- **TensorFlow** та **Keras** – популярні фреймворки для глибокого навчання, які дозволяють створювати складні моделі для класифікації та розпізнавання облич, використовуючи нейронні мережі.
- **PyTorch** – потужний фреймворк для глибокого навчання, який активно використовується для створення моделей машинного навчання та обробки зображень.

Переваги Python:

- Гнучкість. Python підтримує різні стилі програмування, включаючи процедурне, об'єктно-орієнтоване та функціональне програмування.
- Широка екосистема. Велика кількість бібліотек для всіх аспектів розробки, від обробки зображень до глибокого навчання і штучного інтелекту.
- Інтерпретована мова. Python є інтерпретованою мовою, що дозволяє швидко розробляти прототипи та зменшує час на тестування та налагодження.

3.2.4. Програмне середовище.

Для розробки систем розпізнавання обличчя та обробки даних у реальному часі на Python можна використовувати кілька сучасних середовищ розробки та текстових редакторів, що підтримують інтеграцію з бібліотеками для машинного навчання, комп'ютерного зору та обробки великих обсягів даних.

PyCharm є одним з найпопулярніших IDE для Python, що надає потужний набір інструментів для розробки, відладки, тестування і управління проектами. Це ідеальне середовище для великих проектів, яке підтримує автоматичне завершення коду, рефакторинг, інтеграцію з системами контролю версій та можливість налаштування середовища для роботи з бібліотеками комп'ютерного зору. Переваги:

- Повна підтримка Python і додаткових технологій для комп'ютерного зору.
- Інтеграція з бібліотеками для глибокого навчання, такими як TensorFlow, PyTorch і Keras.
- Зручне середовище для тестування та відладки кодів.

Visual Studio Code - це легке і потужне середовище розробки з підтримкою Python через розширення. VS Code добре підходить для швидкого розроблення прототипів, оскільки його можна налаштувати під

потреби розробника за допомогою численних плагінів та розширень.

Переваги:

- Легкість у налаштуванні і використанні.
- Вбудована підтримка для роботи з Jupyter Notebooks, що корисно для аналітики та машинного навчання.
- Підтримка контролю версій через Git.

Jupyter Notebook є ідеальним середовищем для інтерактивного аналізу даних та машинного навчання. Це середовище дозволяє розробникам писати і виконувати код, бачити візуалізації та результати обробки даних в реальному часі, що зручно для прототипування моделей і дослідження даних.

Переваги:

- Підходить для інтерактивного аналізу даних і машинного навчання.
- Підтримка візуалізації даних безпосередньо в середовищі.
- Легка інтеграція з бібліотеками для обробки зображень та розпізнавання облич.

Spyder є ще одним популярним середовищем для Python, особливо для наукових обчислень та аналізу даних. Він має інтеграцію з такими бібліотеками, як NumPy, SciPy, Matplotlib, що робить його підходящим для наукових досліджень і обробки даних. Переваги:

- Зручний інтерфейс для роботи з даними та науковими бібліотеками.
- Підтримка інтерактивних сесій і візуалізацій.
- Потужні інструменти для аналізу та відладки коду.

З огляду на проведений аналіз в якості середовища розроблення було обрано PyCharm.

3.3. Аналіз отриманих результатів

Розглянемо функціональні можливості системи ідентифікації користувача за зображенням обличчя та перевіримо її ефективність на контрольному прикладі. Після розгортання аплікації на локальній машині

стає доступним веб-застосунок, який використовується для автентифікації та подальшого навчання нейронної мережі під керівництвом вчителя.

Щоб почати користуватися системою, необхідно пройти процес автентифікації та авторизації. Вчитель вводить свою електронну адресу як логін і пароль. Після натискання кнопки sign in, введені дані відправляються на сервер для перевірки їх правильності. Якщо дані введено неправильно, користувач отримує відповідне повідомлення про помилку (рис. 3.15).

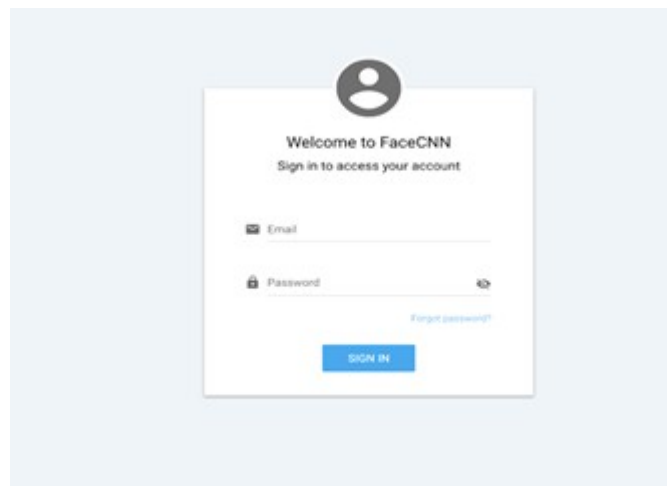


Рис. 3.15. Сторінка автентифікації вчителя

У випадку коректно введених даних авторизація відбувається успішно, і завантажуються сторінка з модальним вікном, що дозволяє завантажити необхідні дані для навчання нейронної мережі – тренувальну вибірку (рис. 3.16). Як показано на екрані, вибірка повинна бути заархівована в форматі zip, що є стандартною практикою для завантаження наборів зображень для навчання, а також відповідних до них даних.

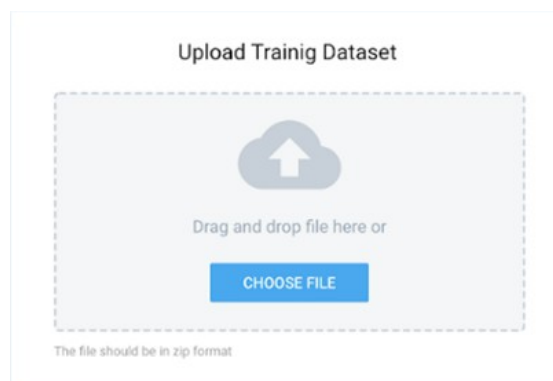


Рис. 3.16. Сторінка завантаження тренувальної вибірки

Після успішного завантаження тренувальної вибірки вчитель може розпочати навчання нейронної мережі, яка буде використовуватись для ідентифікації кінцевих користувачів. Під час навчання вчитель може відслідковувати процес. На стартовій сторінці доступна інформація щодо ходу навчання, а саме:

- Model score vs iteration – значення функції втрат для поточного елемента навчального набору (функція втрат підсумовує помилки нейронної мережі по всіх зображеннях в навчальному наборі).
- Model and training information – загальні дані щодо структури нейронної мережі, зміни ваг нейронів, інформація про тренувальну вибірку, швидкість обробки зображень тощо.
- Parameter ratios – співвідношення параметрів (ваг нейронів та функції помилки) до оновлень по шарам для всіх вагів нейронів мережі та ітерацій.
- Standard deviation – стандартні відхилення для часу активації фільтрів, градієнтів та оновлень.



Рис. 3.17. Сторінка загальної інформації щодо перебігу тренування

Наступним пунктом меню є сторінка моделі нейронної мережі, що містить графічне зображення шарів нейронної мережі, яка функціонує як механізм вибору. Для того щоб побачити інформацію про конкретний шар,

необхідно натиснути на нього. Після вибору шару, користувачеві доступна наступна інформація:

- Таблиця з детальними даними про обраний шар;
- Співвідношення оновлень до параметрів цього шару (згідно з інформацією на загальній сторінці про нейронну мережу). Окремо можна переглянути компоненти цього співвідношення, натискаючи на відповідні вкладки діаграм;
- Прокручуючи сторінку вниз, можна побачити діаграму активацій цього шару по часу;
- Гістограму параметрів та оновлень для кожного типу параметра;
- Швидкість навчання протягом часу.



Рис. 3.18. Сторінка моделі нейронної мережі

Ця сторінка також містить інформацію про машину та систему, на яких здійснюється навчання нейронної мережі. Тут можна знайти дані про використання пам'яті на машині та загальну інформацію щодо сервера – ім'я, операційну систему, архітектуру тощо.



Рис. 3.19. Сторінка з інформацією про навчання

Після завершення навчання нейронної мережі, кінцевий користувач може використовувати розроблену систему для ідентифікації. Головний модуль розробленої програми наведено в додатку А, а модуль роботи з сертифікатами наведено в додатку Б.

Розглянемо сценарії процесу ідентифікації. На етапі навчання в базу даних було внесено зображення актора Джона Сноу (Jon Snow). Дані про іншу акторку Роуз Леслі (Rose Leslie) не були внесені. Як видно, система не змогла розпізнати Роуз Леслі (Rose Leslie) (рис. 3.20), хоча обличчя було коректно локалізовано.

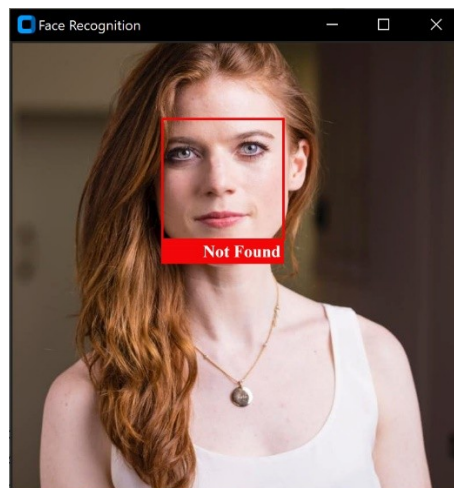


Рис.3.20. Не знайдено обличчя користувача

Додамо до розпізнавання зображення Джона Сноу (рис. 3.21), і система успішно ідентифікує його. Однак варто зазначити, що при зчитуванні зображення з дисплею смартфона виникли труднощі: під певним кутом

нахилу смартфона система не змогла розпізнати обличчя зареєстрованого користувача.

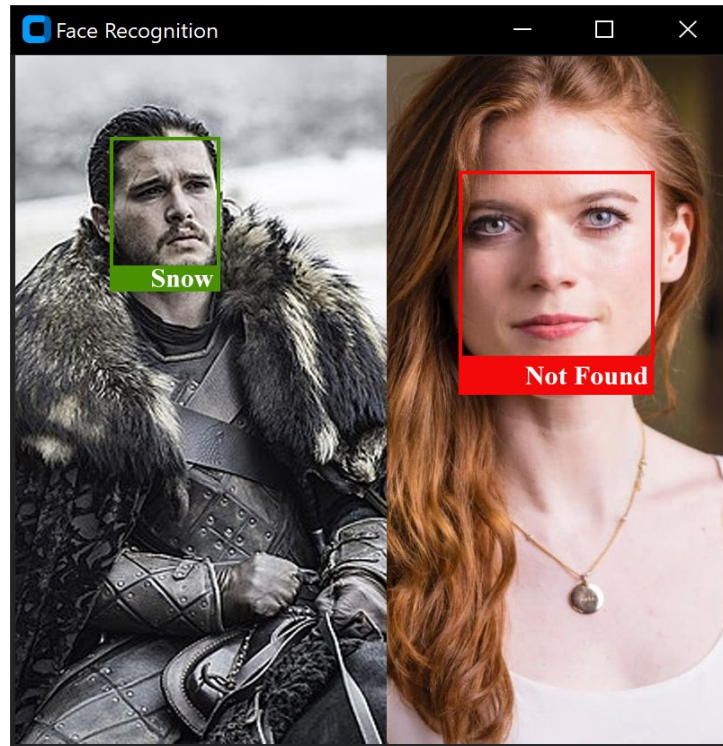


Рис.3.21. Результати розпізнавання користувачів

Зареєструємо Роуз Леслі в системі. Для цього внесемо дані (основне зображення та фрагмент фото з постера) (рис. 3.21).



Рис. 3.22. Внесення фотографій Роуз Леслі

Спробуємо повторно ідентифікувати акторів як було представлено на рис.3.21 так і при використанні сумісного фото. Результат роботи програми представлено на рис.3.23.

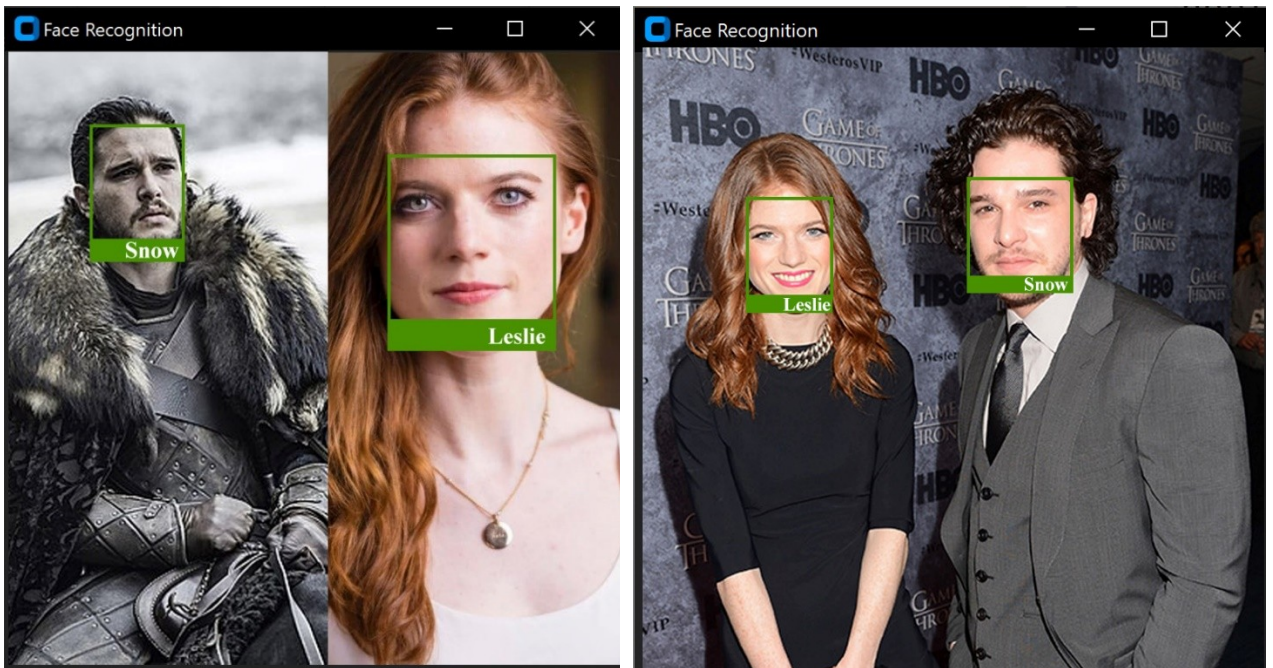


Рис.3.23. Ідентифікація зареєстрованих користувачів

Отже, після розгляду всіх наведених прикладів та проведення кількох тестів на ідентифікацію користувачів, можна зробити висновок, що розроблена система працює коректно та ефективно, успішно вирішуючи поставлені завдання.

Висновки до третього розділу

В даному розділі для реалізації системи було проведено аналіз методів розв'язання задачі, здійснено вибір засобів розроблення та скупструйоване програмне забезпечення, що реалізує візуальну ідентифікацію особи за зображенням обличчя. Здійснено опис основних функцій, що використовуються в системі та наведено контрольний приклад функціонування системи. Для підтвердження коректної роботи було проведено декілька тестів, що показали коректність роботи.

ВИСНОВКИ

У процесі виконання бакалаврської кваліфікаційної роботи була розроблена інформаційна система візуальної ідентифікації особи за зображенням обличчя. Розроблений програмний продукт має зручний та інтуїтивно зрозумілий графічний інтерфейс, що забезпечує ефективне навчання системи на конкретних даних, сформованих підприємством, де можливе впровадження цієї системи. Розроблений інтерфейс дозволяє легко налаштовувати параметри та управляти процесом навчання нейронної мережі, що робить її використання доступним навіть для користувачів без глибоких технічних знань.

Під час виконання роботи було проведено детальний аналітичний огляд методологічних підходів, що використовуються в області розпізнавання обличчя. Були проаналізовані існуючі технології та методи, а також виявлені їх переваги та недоліки. Огляд показав, що обрана методика на основі нейронних мереж, зокрема згорткових мереж, є високоефективною для вирішення задачі ідентифікації, оскільки дозволяє досягти високої точності при обробці зображень.

Також було здійснено проектування, сформульовано мету розробки та функціонування системи, визначено її призначення та можливі сфери використання. Визначено очікувані результати та ефекти від впровадження розробленого програмного продукту. Для цього була розроблена концептуальна модель системи, яка містить детальне описання всіх основних процесів та взаємодій в системі. Було створено діаграми варіантів використання, класів і діяльностей, що дозволило чітко визначити поведінку системи та сформулювати необхідні бізнес-процеси.

Вибір інструментів для реалізації системи був обґрунтований на основі порівняльного аналізу різних технологій. Були оцінені переваги та недоліки кожного з підходів, і в результаті було прийнято рішення використовувати

нейронну мережу зі згорткою, мову програмування Python та середовище розробки PyCharm. Ці інструменти забезпечують високу ефективність, зручність для розробників та сумісність з сучасними технологіями розпізнавання образів.

Програмна система була розроблена, а також було детально описано її функціонування для користувачів з різними ролями. Окремо були розглянуті кроки для ефективної роботи з системою, що включають процеси реєстрації користувачів, налаштування параметрів та виконання ідентифікації. Проведене тестування підтвердило правильність роботи системи, а представлені контрольні приклади продемонстрували правильність прийнятих проєктних рішень.

У підсумку, можна стверджувати, що поставлена мета розробки була досягнута. Створена інформаційна система візуальної ідентифікації особи за зображенням обличчя працює стабільно та ефективно, а після невеликих вдосконалень може бути успішно впроваджена на практиці.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Google, Apple та Microsoft вирішили відмовитися від паролів. URL: <https://ms.detector.media/it-kompanii/post/29442/2022-05-06-google-apple-ta-microsoft-vyrishyly-vidmovytysya-vid-paroliv/>
2. Jain A., Ross A., Nandakumar K. Introduction to Biometrics. Springer, 2011. – 328 p.
3. Kisak P. An Overview of Biometrics: «The New Age of Personal Identification». CreateSpace Independent Publishing Platform, 2017. – 230 p.
4. Грайворонський М. В. Безпека інформаційно-комунікаційних систем: підручник / М. В. Грайворонський, О. М. Новіков. – Київ: Видавнича група BHV, 2009. – 608 с.
5. Богущ В. М. Основи інформаційної безпеки держави / В. М. Богущ, О. К. Юдін. – Київ: МК-Прес, 2005 – 432 с.
6. Антонюк А. О. Основи захисту інформації в автоматизованих системах: навч. посібник / А. О. Антонюк. – Київ. Видавничий дім «КМ Академія», 2003. – 342 с.
7. Когут Ю. І. Корпоративна безпека: практичний посібник/ Ю. І. Когут. – Київ: SIDCON, 2021. – 460с.
8. Кузнецов О. О. Захист інформації в інформаційних системах / О. О. Кузнецов, С. П. Євсєєв, О. Г. Король. – Харків: Вид. ХНЕУ, 2011. – 512 с.
9. Царьов Р. Ю. Біометричні технології: навч. посібник [для вищих навчальних закладів] / Р. Ю. Царьов, Т. М. Лемеха. – Одеса: ОНАЗ ім. О.С. Попова, 2016. – 140 с.
10. Aumasson J.-P. Serious Cryptography: A Practical Introduction to Modern Encryption. No Starch Press, 2017. – 312 p.
11. NeoFace Watch: Face Recognition. [Електронний ресурс] – Режим доступу: <https://www.nec.com/en/global/solutions/biometrics/face/neofacewatch.html>

12. FacevacS-DbScan-Id-Cognitec. [Электронный ресурс] – Режим доступа: <https://www.cognitec.com/facevacS-dbScan-id.html>.
13. Coolseven - Packagist. [Электронный ресурс] – Режим доступа: <https://packagist.org/packages/coolseven/huawei-frs-sdk>.
14. Verilook Face Identification Technology, Algorithm and SDK. [Электронный ресурс] – Режим доступа: <https://www.neurotechnology.com/verilook.html>.
15. Boonkrong S. Authentication and Access Control: Practical Cryptography Methods and Tools. Apress, 2020. – 248 p.
16. Method of Facial Geometric Feature Representation for Information Security Systems. [Электронный ресурс] – Режим доступа: http://elar.khmnu.edu.ua/jspui/bitstream/123456789/12171/1/Radiuk_Method-of-facial-geometric-feature-representation.pdf.
17. What is AI-based Image Recognition? Typical Inference Models and Application Examples Explained. [Электронный ресурс] – Режим доступа: https://www.contec.com/support/blog/2023/230215_ai/.
18. Accurate recognition of object contour based on flexible piezoelectric and piezoresistive dual mode strain sensors. [Электронный ресурс] – Режим доступа: <https://www.sciencedirect.com/science/article/abs/pii/S0924424721005860>
19. A Step-by-Step Explanation of Principal Component Analysis (PSA). [Электронный ресурс] – Режим доступа: <https://builtin.com/data-science/step-step-explanation-principal-component-analysis>.
20. Cusano C. Face Recognition using Three-Dimensional and Multimodal Images: Algorithms and Systems. LAP LAMBERT Academic Publishing, 2019. – 148 p.
21. Ferguson N., Schneier B., Kohno T. Cryptography Engineering: Design Principles and Practical Applications. Wiley; 1st edition, 2010. – 384 p.
22. Reyes C., Rodriguez O.-V., Patricia R.-S., Krikor O. Analysis of Spatio-Temporal Representations for Robust Footstep Recognition with Deep

- Residual Neural Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence. Pp. 1–10.
23. Wiskott L. Face Recognition by Elastic Bunch Graph Matching. IEEE Transactions on Pattern Analysis and Machine Intelligence. Pp. 775–779.
24. Jackson A., Bulat A., Argyriou V., Tzimiropoulos G. Large Pose 3D Face Reconstruction from a Single Image via Direct Volumetric CNN Regression. URL: <http://aaronspplace.co.uk/papers/jackson2017recon/>.
25. Bilan S., Elhoseny M., Hemanth J. Biometric Identification Technologies Based on Modern Data Mining Methods. Springer; 1st ed, 2020. – 210 p.
26. OpenCV Face Recognition. [Электронный ресурс] – Режим доступа: <https://opencv.org/opencv-face-recognition/>.
27. Python. [Электронный ресурс] – Режим доступа: <https://www.python.org/doc/>.

ДОДАТКИ

Додаток 1

Головний модуль програми. Main.py

```

import os
import pickle
import cv2
import face_recognition
import cvzone
import firebase_admin
from firebase_admin import credentials
from firebase_admin import db
from firebase_admin import storage
import numpy as np
from datetime import datetime

cred = credentials.Certificate("serviceAccountKey.json")
firebase_admin.initialize_app(cred, {
    'databaseURL': "https://facerecognition-5ac4c-default-rtdb.firebaseio.com/",
    'storageBucket': "facerecognition-5ac4c.appspot.com"
})

bucket = storage.bucket()

cap = cv2.VideoCapture(0)
cap.set(3, 640)
cap.set(4, 480)

imgBackground = cv2.imread('Resources/background.png')

# Importing the mode images into a list
folderModePath = 'Resources/Modes'
modePathList = os.listdir(folderModePath)
imgModeList = []
for path in modePathList:
    imgModeList.append(cv2.imread(os.path.join(folderModePath, path)))
# print(len(imgModeList))

# Load the encoding file
print("Loading Encode File ...")
file = open('EncodeFile.p', 'rb')
encodeListKnownWithIds = pickle.load(file)
file.close()
encodeListKnown, studentIds = encodeListKnownWithIds
# print(studentIds)
print("Encode File Loaded")

modeType = 0
counter = 0

```

```

id = -1
imgStudent = []

while True:
    success, img = cap.read()

    imgS = cv2.resize(img, (0, 0), None, 0.25, 0.25)
    imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)

    faceCurFrame = face_recognition.face_locations(imgS)
    encodeCurFrame = face_recognition.face_encodings(imgS, faceCurFrame)

    imgBackground[162:162 + 480, 55:55 + 640] = img
    imgBackground[44:44 + 633, 808:808 + 414] = imgModeList[modeType]

    if faceCurFrame:
        for encodeFace, faceLoc in zip(encodeCurFrame, faceCurFrame):
            matches = face_recognition.compare_faces(encodeListKnown, encodeFace)
            faceDis = face_recognition.face_distance(encodeListKnown, encodeFace)
            # print("matches", matches)
            # print("faceDis", faceDis)

            matchIndex = np.argmin(faceDis)
            # print("Match Index", matchIndex)

            if matches[matchIndex]:
                # print("Known Face Detected")
                # print(studentIds[matchIndex])
                y1, x2, y2, x1 = faceLoc
                y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4
                bbox = 55 + x1, 162 + y1, x2 - x1, y2 - y1
                imgBackground = cvzone.cornerRect(imgBackground, bbox, rt=0)
                id = studentIds[matchIndex]
                if counter == 0:
                    cvzone.putTextRect(imgBackground, "Loading", (275, 400))
                    cv2.imshow("Face Attendance", imgBackground)
                    cv2.waitKey(1)
                    counter = 1
                    modeType = 1

            if counter != 0:

                if counter == 1:
                    # Get the Data
                    studentInfo = db.reference(f'Students/{id}').get()
                    print(studentInfo)
                    # Get the Image from the storage
                    blob = bucket.get_blob(f'Images/{id}.png')

```

```

array = np.frombuffer(blob.download_as_string(), np.uint8)
imgStudent = cv2.imdecode(array, cv2.COLOR_BGRA2BGR)
# Update data of attendance
datetimeObject = datetime.strptime(studentInfo['last_attendance_time'],
                                   "%Y-%m-%d %H:%M:%S")
secondsElapsed = (datetime.now() - datetimeObject).total_seconds()
print(secondsElapsed)
if secondsElapsed > 30:
    ref = db.reference(f'Students/{id}')
    studentInfo['total_attendance'] += 1
    ref.child('total_attendance').set(studentInfo['total_attendance'])
    ref.child('last_attendance_time').set(datetime.now().strftime("%Y-%m-%d %H:%M:
%S"))
else:
    modeType = 3
    counter = 0
    imgBackground[44:44 + 633, 808:808 + 414] = imgModeList[modeType]

if modeType != 3:

    if 10 < counter < 20:
        modeType = 2

    imgBackground[44:44 + 633, 808:808 + 414] = imgModeList[modeType]

    if counter <= 10:
        cv2.putText(imgBackground, str(studentInfo['total_attendance']), (861, 125),
                    cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 1)
        cv2.putText(imgBackground, str(studentInfo['major']), (1006, 550),
                    cv2.FONT_HERSHEY_COMPLEX, 0.5, (255, 255, 255), 1)
        cv2.putText(imgBackground, str(id), (1006, 493),
                    cv2.FONT_HERSHEY_COMPLEX, 0.5, (255, 255, 255), 1)
        cv2.putText(imgBackground, str(studentInfo['standing']), (910, 625),
                    cv2.FONT_HERSHEY_COMPLEX, 0.6, (100, 100, 100), 1)
        cv2.putText(imgBackground, str(studentInfo['year']), (1025, 625),
                    cv2.FONT_HERSHEY_COMPLEX, 0.6, (100, 100, 100), 1)
        cv2.putText(imgBackground, str(studentInfo['starting_year']), (1125, 625),
                    cv2.FONT_HERSHEY_COMPLEX, 0.6, (100, 100, 100), 1)

        (w, h), _ = cv2.getTextSize(studentInfo['name'],
cv2.FONT_HERSHEY_COMPLEX, 1, 1)
        offset = (414 - w) // 2
        cv2.putText(imgBackground, str(studentInfo['name']), (808 + offset, 445),
                    cv2.FONT_HERSHEY_COMPLEX, 1, (50, 50, 50), 1)

        imgBackground[175:175 + 216, 909:909 + 216] = imgStudent

    counter += 1

```

```
        if counter >= 20:
            counter = 0
            modeType = 0
            studentInfo = []
            imgStudent = []
            imgBackground[44:44 + 633, 808:808 + 414] = imgModeList[modeType]
    else:
        modeType = 0
        counter = 0
    # cv2.imshow("Webcam", img)
    cv2.imshow("Face Attendance", imgBackground)
    cv2.waitKey(1)
```

Додаток 2

Модуль роботи з сертифікатами. EncodeGenerator.py

```

import cv2
import face_recognition
import pickle
import os
import firebase_admin
from firebase_admin import credentials
from firebase_admin import db
from firebase_admin import storage

cred = credentials.Certificate("serviceAccountKey.json")
firebase_admin.initialize_app(cred, {
    'databaseURL': "https://facerecognition-5ac4c-default-rtdb.firebaseio.com/",
    'storageBucket': "facerecognition-5ac4c.appspot.com"
})

# Importing student images
folderPath = 'Images'
pathList = os.listdir(folderPath)
print(pathList)
imgList = []
studentIds = []
for path in pathList:
    imgList.append(cv2.imread(os.path.join(folderPath, path)))
    studentIds.append(os.path.splitext(path)[0])

    fileName = f'{folderPath}/{path}'
    bucket = storage.bucket()
    blob = bucket.blob(fileName)
    blob.upload_from_filename(fileName)

    # print(path)
    # print(os.path.splitext(path)[0])
print(studentIds)

def findEncodings(imagesList):
    encodeList = []
    for img in imagesList:
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        encode = face_recognition.face_encodings(img)[0]
        encodeList.append(encode)
    return encodeList

print("Encoding Started ...")

```

```
encodeListKnown = findEncodings(imgList)
encodeListKnownWithIds = [encodeListKnown, studentIds]
print("Encoding Complete")
```

```
file = open("EncodeFile.p", 'wb')
pickle.dump(encodeListKnownWithIds, file)
file.close()
print("File Saved")
```