Міністерство освіти і науки України
Луцький національний технічний університет
Факультет комп'ютерних та інформаційних технологій
Кафедра комп'ютерних наук

КВАЛІФІКАЦІЙНА РОБОТА
ЗА СТУПЕНЕМ ВИЩОЇ ОСВІТИ «МАГІСТР»

ДОСЛІДЖЕННЯ ТА ЗАСТОСУВАННЯ ТЕХНОЛОГІЇ FACE-ID З ВИКОРИСТАННЯМ ШТУЧНОГО ІНТЕЛЕКТУ ДЛЯ ОПЛАТИ ТОВАРІВ

RESEARCH AND APPLICATION OF FACE-ID TECHNOLOGY USING ARTIFICIAL INTELLIGENCE FOR PAYMENT PROCESSING

спеціальність 122 «Комп'ютерні науки»

освітня програма «Комп'ютерні науки»

Виконав: здобувач вищої освіти
групи КНм-21
Денисюк Андрій Вадимович
_____
(підпис)
Керівник: к.т.н., доцент
Лук'янчук Юрій Анатолійович
_____
(підпис)

Кваліфікаційну роботу
допущено до захисту
«___» _____ 2025 р.
Гарант освітньої програми:
к.т.н., доцент
Ліщина Валерій Олександрович
_____
(підпис)

Луцьк – 2025 року

ЛУЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

Факультет *комп'ютерних та інформаційних технологій*
Кафедра *комп'ютерних наук*
Ступінь вищої освіти: *магістр*
Галузь знань: *12 Інформаційні технології*
Спеціальність: *122 Комп'ютерні науки*
Освітня програма: *«Комп'ютерні науки»*

ЗАТВЕРДЖУЮ

Завідувач кафедри
_____Валерій ЛІЩИНА
«14» травня 2025 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ ЗДОБУВАЧА
ДРУГОГО (МАГІСТЕРСЬКОГО) РІВНЯ ВИЩОЇ ОСВІТИ

Денисюк Андрій Вадимович

(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи «*Дослідження та застосування технології face-id з використанням штучного інтелекту для оплати товарів*»
Керівник роботи *к.т.н., доцент Лук'янчук Юрій Анатолійович*
затверджені наказом закладу вищої освіти від *«14» травня 2025 р*. № 255/01-02
2. Строк подання здобувачем вищої освіти кваліфікаційної роботи *«15» грудня 2025 р.*
3. Вихідні дані до роботи: *статті, дослідження вітчизняних та закордонних авторів в даній області, сучасні методи і засоби розробки, технічна документація технологій розробки.*
4. Зміст розрахунково-пояснювальної записки (перелік питань, що потрібно розробити): *Аналіз проблематики та постановка завдань дослідження, теоретичне дослідження та практична реалізація системи розпізнавання облич на основі ШІ, експерементальне дослідження результативності системи розпізнавання облич на основі ШІ.*
5. Перелік графічного матеріалу: *1. Процес системи розпізнавання обличчя з моделлю глибокого навчання. 2. Графік порогу точності 3. Порівняльний аналіз наявних рішень 4. Принцип роботи каскадної архітектури MTCNN 5. Діаграма варіантів використання системи розпізнавання облич. 6. UML-діаграма класів системи розпізнавання облич. 7. Функція навчання нейронної мережі Face ID. 8. Модуль захоплення та обробки відеопотоку. 9. Клас взаємодії з Amazon DynamoDB. 10. Модуль шифрування біометричних векторів 11. Характеристики використаних наборів даних. 12. Порівняння результатів тестування різних моделей розпізнавання облич. 13. Графік порівняння точності моделей розпізнавання облич. 14. Вплив зовнішніх факторів на точність розпізнавання.*

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис	
		завдання видав	завдання прийняв
Аналіз проблематики та постановка завдань дослідження	*Лук'янчук Ю. А.*		
Теоретичне дослідження та практична реалізація системи розпізнавання облич на основі ШІ	*Лук'янчук Ю. А.*		
Експеременталне дослідження результативності системи розпізнавання облич на основі ШІ	*Лук'янчук Ю. А.*		
Показник запозичень тексту	_____ %		
Інструментальна перевірка	*Кошелюк В. А.*		
Нормоконтроль	*Сачук В. О.*		
Гарант ОПП	*Ліщина В. О.*		

7. Дата видачі завдання *«14» травня 2025 р*.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи магістра	Строк виконання етапів роботи	Примітка
1.	*Обґрунтування теми дослідження*	*до 14.05.2025*	
2.	*Провести огляд літературних джерел по темі кваліфікаційної роботи*	*до 21.08.2025*	
3.	*Провести аналіз загальної проблеми і вибір напрямків дослідження*	*до 05.09.2025*	
4.	*Розробити функціональну схему роботи програмного продукту*	*до 30.09.2025*	
5.	*Описати засоби розробки об'єкта проектування*	*до 09.10.2025*	
6.	*Практична реалізація об'єкта проектування*	*до 20.10.2025*	
7.	*Розробити методику для проведення експерименту*	*до 29.10.2025*	
8.	*Провести аналіз результатів експерименту*	*до 12.11.2025*	
9.	*Формування списку використаних джерел*	*до 17.11.2025*	
10.	*Оформлення ілюстративного матеріалу*	*до 25.11.2025*	
11.	*Інструментальна перевірка на академічний плагіат*	*до 03.12.2025*	
12.	*Здача чистового варіанту кваліфікаційної роботи на кафедру*	*до 05.12.2025*	

Здобувач вищої освіти _____ Андрій ДЕНИСЮК

Керівник роботи _____ Юрій ЛУК'ЯНЧУК

АНОТАЦІЯ

Денисюк А. В. Дослідження та застосування технології face-id з використанням штучного інтелекту для оплати товарів. Рукопис.

Кваліфікаційна робота магістра ОП «Комп'ютерні науки» спеціальності 122 «Комп'ютерні науки». Луцький національний технічний університет. Луцьк, 2025.

Кваліфікаційна робота магістра складається з вступу, 3 розділів, висновків і пропозицій, списку використаних джерел, додатків (згідно структури кваліфікаційної роботи, затвердженої кафедрою).

У роботі досліджено сучасні методи комп'ютерного зору та глибокого навчання, спрямовані на створення системи розпізнавання облич для реалізації безконтактних оплат. Проаналізовано архітектури нейронних мереж, методи детекції та векторизації ознак, визначено їх переваги та обмеження у контексті платіжних сервісів. Розроблено програмний прототип системи, що базується на алгоритмах MTCNN та ArcFace, а також реалізує модульну структуру з підтримкою роботи в режимі реального часу.

Виконано етапи збирання та підготовки даних, використовуючи публічні датасети LFW, CelebA та вибірки з Kaggle, а також локально сформовану авторську базу. Проведено навчання моделі, оптимізацію параметрів і комплексне тестування із застосуванням ключових метрик – accuracy, FAR, FRR та FPS. Отримані результати підтверджують високу точність і стабільність роботи системи, а також можливість її застосування у платіжних терміналах, системах самообслуговування та сервісах контролю доступу. Показано перспективи впровадження запропонованої технології у бізнес-процеси, а також окреслено подальші напрями удосконалення розробленої моделі.

Ключові слова: розпізнавання облич, штучний інтелект, Face ID, ArcFace, MTCNN, комп'ютерний зір, безконтактні платежі.

ABSTRACT

Andriy Denysiuk. Research and application of face-id technology using artificial intelligence for payment processing. Manuscript.

Qualification work for master's degree in «Computer Science» specialty 122 «Computer Science». Lutsk National Technical University. Lutsk, 2025.

The master's qualification work consists of an introduction, 3 chapters, conclusions and proposals, a list of sources used, appendices (according to the structure of the qualification work approved by the department).

The thesis examines modern computer vision and deep learning methods used to develop a facial recognition system for contactless payment processing. Various neural network architectures, face detection algorithms, and feature embedding approaches are analyzed, with particular attention to their applicability, accuracy, and robustness in financial and retail environments. A software prototype was implemented based on MTCNN and ArcFace, incorporating a modular architecture capable of real-time operation.

The study includes dataset collection and preparation using public datasets such as LFW, CelebA, multiple Kaggle collections, as well as a custom dataset created by the author. The model was trained, fine-tuned, and experimentally evaluated using key performance metrics, including accuracy, FAR, FRR, and FPS. The results demonstrate high recognition accuracy and system stability, confirming the feasibility of integrating the proposed solution into payment terminals, self-service systems, and access control platforms. The work also outlines practical implementation opportunities and identifies future directions for improving and extending the developed system.

Keywords: facial recognition, artificial intelligence, Face ID, ArcFace, MTCNN, computer vision, contactless payments.

ЗМІСТ

ВСТУП 7
РОЗДІЛ 1 АНАЛІЗ ПРОБЛЕМАТИКИ ТА ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ 9
1.1 Огляд і аналіз предметної області проблеми (задачі), результатів існуючих теоретичних та експериментальних досліджень 9
1.2 Огляд і аналіз методів та засобів розробки системи розпізнавання облич на основі штучного інтелекту для вирішення проблеми дослідження 14
1.3 Постановка завдання на кваліфікаційну роботу магістра 17
РОЗДІЛ 2 ТЕОРЕТИЧНЕ ДОСЛІДЖЕННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ РОЗПІЗНАВАННЯ ОБЛИЧ НА ОСНОВІ ШІ 19
2.1 Обґрунтування вибору шляхів, технологій (алгоритмів) і засобів вирішення поставленого завдання 19
2.2 Практична реалізація об’єкта проектування 25
РОЗДІЛ 3 ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ РЕЗУЛЬТАТИВНОСТІ СИСТЕМИ РОЗПІЗНАВАННЯ ОБЛИЧ НА ОСНОВІ ШІ 36
3.1 Методика проведення дослідження 36
3.2 Обробка та аналіз отриманих результатів 40
ВИСНОВКИ 46
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 49
ДОДАТКИ 51

ВСТУП

Нинішній етап еволюції інформаційних технологій характеризується активним упровадженням штучного інтелекту у всі сфери людської діяльності, зокрема у сферу фінансових технологій (FinTech). Колосальний обсяг операцій, що здійснюються щоденно у світовому масштабі, вимагає не лише зручності у користуванні, але й максимального рівня захищеності. Одним із перспективних напрямів у цьому контексті є використання біометричних систем ідентифікації, що забезпечують автентифікацію користувача за його фізіологічними ознаками.

Технологія розпізнавання обличчя отримала чимале поширення у мобільних пристроях, системах доступу та апаратному відеоконтролі, проте її потенціал у сфері платіжних інтерфейсів використовується не повною мірою. Інтеграція можливостей штучного інтелекту, а саме глибоких нейронних мереж, дозволяє суттєво підвищити рівень точності та швидкість ідентифікації, що, своєю чергою, забезпечує надійність фінансових переказів та підвищує комфорт для кінцевих користувачів.

Актуальність даної роботи обумовлена необхідністю створення інтелектуальних систем безконтактних оплат, що поєднують високу безпеку, швидкодію та простоту використання. В умовах цифровізації економіки та поширення концепції «cashless society» розробка таких систем є важливим кроком у напрямі модернізації сучасних фінансових сервісів.

Світова тенденція розвитку полягає у поєднанні біометричних методів ідентифікації з технологіями штучного інтелекту для створення гібридних фінансових систем, що поєднують зручність і безпеку. Саме це визначає наукову й практичну значущість теми магістерського дослідження.

Метою роботи є створення інтелектуальної інформаційної системи, яка забезпечує можливість здійснення безконтактних оплат на основі технології Face-ID із використанням методів штучного інтелекту.

Для досягнення поставленої мети необхідно розв’язати такі завдання:

– провести аналіз сучасних методів біометричної ідентифікації та визначити переваги використання Face-ID у фінансових транзакціях;

– дослідити архітектури нейронних мереж, що використовуються для розпізнавання облич, та обґрунтувати вибір моделі ArcFace;

– розробити структуру та програмну реалізацію системи безконтактної оплати на основі Face-ID;

– провести тестування створеної системи та оцінити її точність, швидкодію і надійність у реальних умовах використання;

– визначити напрями вдосконалення розробленої моделі та перспективи її впровадження в реальні платіжні сервіси.

Об’єкт дослідження – процес ідентифікації користувачів під час здійснення платіжних транзакцій з використанням технології розпізнавання облич.

Предмет дослідження – методи, алгоритми та засоби застосування штучного інтелекту для побудови системи Face-ID, орієнтованої на безконтактні оплати.

Наукова новизна магістерської роботи полягає у розробленні підходу до інтеграції моделі ArcFace у платіжну систему, що забезпечує підвищену точність розпізнавання та мінімізацію хибних спрацьовувань при верифікації користувачів. Уперше запропоновано адаптацію нейронної моделі для задач біометричної автентифікації у реальному часі з урахуванням обмежень обчислювальних ресурсів.

Практична цінність полягає у можливості впровадження розробленої системи у сферу торгівлі, сервісних послуг і транспорту як альтернативного способу безконтактної оплати без використання карт чи мобільних пристроїв.

Основні результати дослідження апробовано в рамках реалізації європейського освітнього проєкту «AI for Youth», що здійснюється за підтримки Європейського інституту інновацій і технологій (EIT Food) у межах ініціативи EIT Deep Tech Talent Initiative, що підтверджено у додатку А.

РОЗДІЛ 1
АНАЛІЗ ПРОБЛЕМАТИКИ ТА ПОСТАНОВКА ЗАВДАНЬ ДОСЛІДЖЕННЯ

1.1 Огляд і аналіз предметної області проблеми (задачі), результатів існуючих теоретичних та експериментальних досліджень

У сучасному інформаційному суспільстві стрімке зростання обсягів цифрових платежів, безконтактних транзакцій та розвитку FinTech-сервісів породжує необхідність впровадження нових методів ідентифікації користувачів, які поєднують високу надійність із зручністю використання. Однією з таких методик виступає біометричне розпізнавання облич (Face-ID), яке за останні роки отримало значне поширення у мобільних пристроях та системах контролю доступу. Тим не менш, застосування цієї технології саме у сфері платежів та торгівлі залишається недостатньо вивченим, що обумовлює наукову й практичну значущість дослідження.

У науковій літературі питання побудови систем розпізнавання облич широко висвітлене та систематизоване. Зокрема, у ґрунтовному огляді Li Qinjun та ін. «Facial Recognition Technology: A Comprehensive Overview» [1] представлено структурну еволюцію технологій розпізнавання, окреслено їх базові етапи – детекцію, витяг ознак та зіставлення, а також підкреслено значення глибокого навчання для підвищення точності та стійкості таких систем. У роботі наголошується, що послідовність етапів попередньої обробки, формування векторних ознак і подальшої ідентифікації є ключовою для сучасних біометричних алгоритмів, оскільки забезпечує їх узагальнювальну здатність і адаптивність до різноманітних зовнішніх умов. У контексті цього огляду наведена схема (рис. 1.1) відтворює типовий конвеєр обробки зображення в системах розпізнавання облич: від отримання вхідного кадру до виявлення області обличчя, подальшого вилучення глибинних ознак за допомогою нейронної мережі та фінального етапу порівняння з шаблонами, збереженими в базі даних.

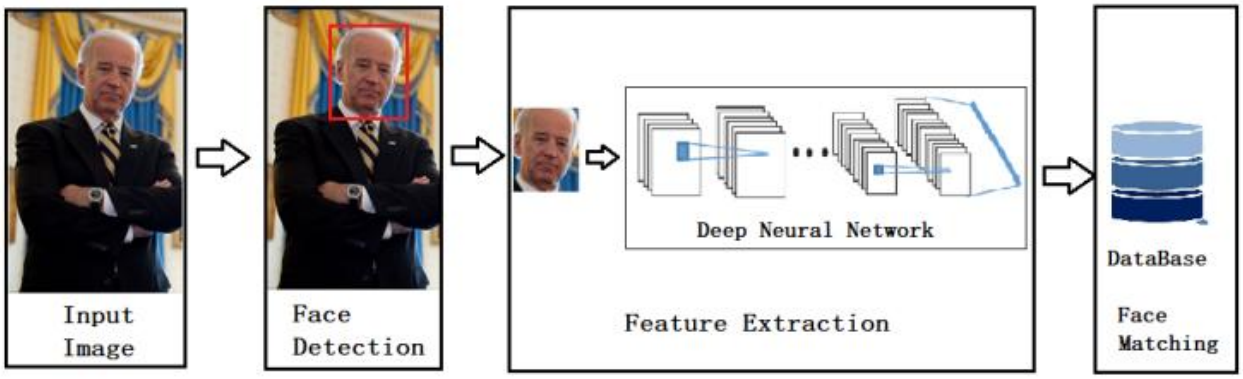

Рисунок 1.1 – Процес системи розпізнавання обличчя з моделлю глибокого навчання [1]

Після того, як обличчя буде зафіксоване, відповідний фрагмент передається в модуль витягу ознак, де формується його компактне векторне представлення, стійке до варіацій ракурсу, освітлення й міміки.

Насамкінець, фінальний блок порівнює цей згенерований вектор із заздалегідь збереженими зразками, що дозволяє виконати ідентифікацію або верифікацію користувача.

Подібний алгоритмічний ланцюжок знаходить своє підтвердження у дослідженні, проведеному Su C. та його колегами [2], де наголошується, що послідовне розмежування на стадії виявлення, вилучення ознак та подальшої класифікації є фундаментом для переважної більшості актуальних рішень, побудованих на основі глибокого нейронного навчання.

Подібні аналізи демонструють наявність технологічної бази, однак аспекти впровадження цього у системи платежів (зокрема, уможливлення функціонування в реальному часі, враховуючи обмеженість ресурсів пристроїв та значні потреби у захисті даних) досі потребують глибшого опрацювання.

У сфері впровадження безконтактних оплат, що базуються на біометричній автентифікації, особливого значення набувають дослідження, які комплексно аналізують не лише технічні аспекти систем розпізнавання облич, а й питання безпеки, довіри користувачів та практичної ефективності таких рішень у фінансовій інфраструктурі. Систематичний огляд «Advancing Secure Face

Recognition Payment Systems: A Systematic Literature Review» [3] узагальнює сучасні підходи до створення біометричних платіжних систем та визначає основні ризики, що супроводжують їхнє застосування. Автори підкреслюють, що розвиток глибинних моделей, істотно підвищили точність ідентифікації, проте ключові загрози не були повністю усунені. У роботі акцентовано увагу на вразливості систем до атак із використанням друкованих зображень, 3D-реплік облич, високоякісних відеоматеріалів і deepfake-технологій, що робить антиспуфінгові механізми необхідною умовою функціонування платіжних сервісів.

Інше дослідження [4] підкреслює, що навіть за наявності високих показників точності на контрольованих датасетах, моделі глибокого навчання демонструють суттєве зниження продуктивності у реальних комерційних сценаріях.

Це відображено на рисунку 1.2, де зі зростанням порога від 0.3 до 0.8 точність падає з приблизно 95 % до майже 74 %, що вказує на необхідність тонкого балансування між безпекою та зручністю.

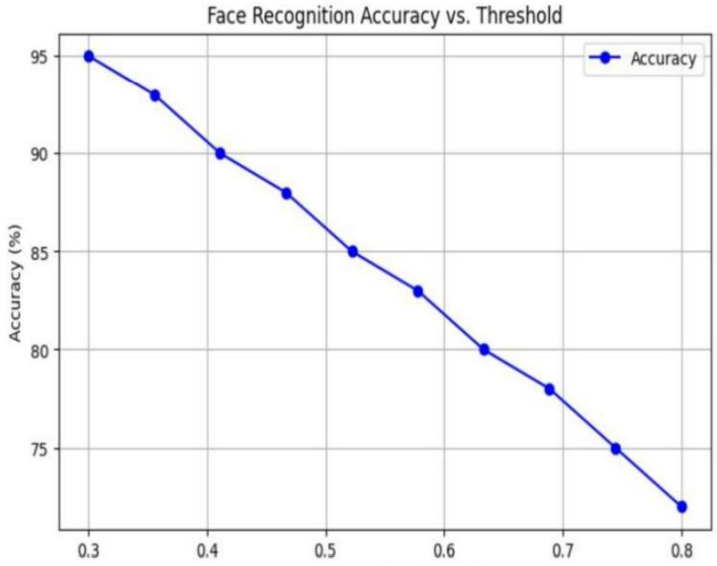

Рисунок 1.2 – Графік порогу точності [4]

Автори зазначають, що системи розпізнавання облич у платіжних терміналах, касових апаратах чи торгових точках працюють у середовищах із

постійними змінами освітлення, варіаціями кута огляду, різною якістю камер та частковими перекриттями обличчя користувачів. З огляду на це, точність, отримана під час лабораторних випробувань, рідко відтворюється у реальній експлуатації. Дослідження демонструє, що жорсткість порога розпізнавання істотно впливає на результат: надто високий поріг знижує кількість хибних прийняттів, але водночас збільшує кількість хибних відмов.

Узявши до уваги описані обмеження, автори відзначають перспективність використання технології верифікації за обличчям у фінансових системах, проте підкреслюють, що її масштабованість і надійність напряму пов’язані з удосконаленням алгоритмів обробки та різноманітністю тренувальних даних. Це корелює з висновками, у яких зазначено, що здатність системи працювати в режимі реального часу, забезпечувати швидку реакцію та підтримувати стійкість до шумів робить її конкурентним доповненням до традиційних методів автентифікації.

Одночасно існують також роботи, що присвячені споживчій поведінці – дослідження «Facial-Recognition Payment: An Example of Chinese Consumers» [5] вказує на те, що фактори безпеки, видимості (visibility) та іміджу (social image) впливають на готовність використовувати біометричні платежі.

У межах предметної області виокремлюються кілька стійких тенденцій, що визначають сучасний розвиток технологій біометричної автентифікації. Насамперед спостерігається активне впровадження глибинних нейронних мереж, включно з класичними згортковими архітектурами та моделями трансформерного типу, які забезпечують високоточне вилучення ознак та суттєво підвищують якість розпізнавання. Паралельно формується практика обов’язкового використання антиспуфінгових механізмів, покликаних захистити системи від атак із застосуванням фотографій, відео чи тривимірних моделей.

Значної уваги набуває і специфічний платіжний контекст, де біометричні рішення мають працювати за умов обмежених обчислювальних ресурсів, мінімальної затримки та підвищених вимог до точності. Окрім технічних аспектів, актуальним є дослідження поведінкових характеристик користувачів і

рівня їхньої довіри до технологій, оскільки саме ці фактори визначають готовність до впровадження біометричних рішень у фінансові екосистеми.

У цьому контексті важливим є аналіз наявних комерційних рішень, що вже застосовують біометричну ідентифікацію для платіжних операцій. Для систематизації таких платформ доцільно розглянути їхні ключові характеристики, зокрема використовувані алгоритми, рівень безпеки, а також їхні сильні та слабкі сторони. Узагальнені порівняльні відомості наведено у таблиці 1.1, яка демонструє технологічне різноманіття сучасних продуктів і дозволяє зробити висновки щодо тенденцій їх розвитку.

Таблиця 1.1 – Порівняльний аналіз наявних рішень

Назва технології	Алгоритми	Безпека	Переваги	Недоліки
Smile to Pay (Alibaba)	Власні моделі AI, 3D-аналіз облич	Висока (виявлення підробок)	Швидка оплата, інтеграція з Alipay	Висока вартість, залежність від екосистеми
PopID	CNN, векторна ідентифікація	Середня	Простота, швидкість, підтримка багатьох точок продажу	Потребує попередньої реєстрації
VisionLabs LUNA POS	Deep Learning, FaceNet/ArcFace	Висока	Гнучкість, підтримка SDK, бізнес-функції	Комерційна ліцензія, потреба в камерах високої якості

Звідси випливає, що попри успішні реалізації біометричних платежів на ринку, ці системи суттєво різняться за рівнем точності, технологічною базою та вимогами до апаратного забезпечення. Відповідно, наявні рішення не завжди повною мірою враховують обмеження фінансових терміналів або специфіку реального використання, що підсилює мотивацію до подальших досліджень та створення більш адаптивних і масштабованих моделей.

Попри суттєвий прогрес, аналіз літератури виявляє низку вагомих прогалин. Дослідження, що безпосередньо охоплюють платіжні сценарії з використанням Face-ID у фінансових терміналах або торгових точках, залишаються порівняно обмеженими. Недостатньо вивченими є питання масштабованості таких систем та їх інтеграції в існуючу платіжну інфраструктуру, включно з POS-терміналами та мобільними платформами. Також бракує робіт, присвячених аналізу поведінкових аспектів і ступеню прийняття користувачами безконтактних оплат на основі біометрії обличчя. Додатковою проблемою є відсутність уніфікованих стандартів або методик оцінювання ефективності систем біометричних платежів, що ускладнює порівняння різних технологічних рішень між собою.

Виходячи з викладеного, можна зазначити, що попри високий рівень розвитку технологічної бази розпізнавання облич у загальному контексті, її адаптація до комерційних платіжних процесів усе ще потребує комплексних досліджень.

1.2 Огляд і аналіз методів та засобів розробки системи розпізнавання облич на основі штучного інтелекту для вирішення проблеми дослідження

Проблема створення надійної системи розпізнавання облич для здійснення безконтактних оплат потребує комплексного підходу, який поєднує сучасні методи штучного інтелекту, комп’ютерного зору та інформаційної безпеки. Ефективність подібної системи залежить від правильного вибору алгоритмів і засобів розробки, що забезпечують високу точність розпізнавання, швидкодію обробки відеопотоку та захист персональних даних користувача.

Сучасні методи розпізнавання облич умовно поділяються на три напрямки. Перший охоплює класичні підходи комп’ютерного зору, зокрема PCA, ICA та LDA, які працюють на основі аналізу геометричних і статистичних характеристик зображення. Вони демонструють обмежену точність і чутливі до

змін освітлення, ракурсу чи шумів, тому сьогодні використовуються лише у простих системах або як допоміжні засоби.

Другий напрямок – методи машинного навчання, де для класифікації застосовують SVM, Random Forest або kNN [6]. Такі алгоритми забезпечують кращі результати за умови якісно підібраних ознак, однак їх ефективність значною мірою залежить від обсягу та різноманітності навчальної вибірки.

Третю та найбільшу групу становлять моделі глибокого навчання. Згорткові нейронні мережі здатні автоматично виділяти інформативні ознаки обличчя, що забезпечує високу точність і стійкість до варіацій освітлення, міміки, положення голови або часткового перекриття. Саме цей підхід є домінуючим у сучасних біометричних системах.

Одним із найбільш високоточних алгоритмів є ArcFace, що оперує кутовою метрикою для оцінки схожості між векторними представленнями облич. Завдяки цій концепції вдається значно краще розрізняти різних осіб, що є незамінним для фінансових сервісів, де навіть незначна неточність може мати серйозні наслідки.

Для первинного виявлення облич у відеоданих залучається технологія MTCNN (Multi-task Cascaded Convolutional Networks), яка інтегрує процеси локалізації та нормалізації облич у режимі реального часу. Застосування MTCNN мінімізує вплив зовнішніх умов та підвищує якість наступних етапів обробки.

Для маніпуляцій із візуальною інформацією, як відео, так і статичними зображеннями, ефективно використовується набір інструментів OpenCV, який пропонує багатий арсенал функцій комп’ютерного зору: від корегування кольорів та фільтрування до відстежування рухомих елементів та трансформації систем координат.

Основним засобом реалізації системи виступає мова програмування Python, яка надає доступ до величезної кількості бібліотек у сферах ШІ, машинного навчання, зорової обробки даних та складних обчислень. Її популярність у наукових і промислових проєктах зумовлена відкритістю екосистеми, кросплатформністю та великою кількістю готових рішень.

Під час аналізу інструментів для реалізації системи розпізнавання облич було розглянуто кілька популярних рішень у сфері штучного інтелекту, обробки зображень та керування даними. На основі порівняння функціональних можливостей, продуктивності та сумісності з сучасними бібліотеками було обрано оптимальний набір технологій.

Основними фреймворками для побудови нейронних мереж визначено TensorFlow та Keras, які забезпечують гнучкість у створенні моделей глибокого навчання, мають підтримку GPU та розвинену екосистему інструментів для тренування й оцінювання моделей. Для роботи з багатовимірними масивами даних та їх попередньої обробки доцільним є використання бібліотек NumPy та Pandas, що добре інтегруються з TensorFlow і прискорюють підготовку навчальних вибірок.

Для аналізу та обробки відеопотоку, а також для виконання базових операцій комп’ютерного зору (масштабування, нормалізація, виявлення облич) було обрано OpenCV, який забезпечує стабільну роботу з камерами та високий рівень оптимізації. Для подальшої візуалізації результатів навчання та перевірки ефективності моделей використовується Matplotlib.

Стосовно зберігання біометричних даних користувачів, доцільно обрано Amazon DynamoDB [7], котра є частиною апаратної екосистеми Amazon Web Services (AWS). Цей тип NoSQL-бази даних, вирізняючись високою продуктивністю, гарантує можливість експоненційного нарощування потужності, мінімальний час відгуку при зверненні до інформації та інтегровані системи для забезпечення стійкості до збоїв.

Підсумовуючи вище сказане, підібрані засоби дають змогу реалізувати функціонал, потрібний для побудови системи, що інтегрує мережі нейронного типу, маніпуляції з відеоінформацією та надійне збереження даних, задовольняючи при цьому актуальні потреби в галузі машинного зору. З метою оптимізації та управління змінами обрано систему Git, яку доповнює сховище GitHub, а для розробки, відлагодження та перевірки програмного коду задіяні інтегровані середовища PyCharm та Visual Studio Code.

Оскільки досліджувана система має справу з чутливою інформацією, надзвичайно важливим є впровадження методів захисту даних. До них належать:

– шифрування біометричних ознак користувачів;

– використання хеш-функцій та токенів доступу для автентифікації;

– впровадження протоколів безпеки HTTPS і захищених з’єднань із базами даних;

– використання методів антиспуфінгу для запобігання підробкам (наприклад, визначення руху, блиску очей, 3D-структури обличчя).

Враховуючи наведене, проведений аналіз свідчить, що інтеграція методів глибокого навчання, інструментів комп’ютерного зору та сучасних засобів інформаційної безпеки створює передумови для розробки надійної системи розпізнавання облич, придатної до практичного використання у сферах фінансових технологій, безпеки та автоматизованих сервісів.

1.3 Постановка завдання на кваліфікаційну роботу магістра

На основі аналізу предметної області, розглянутих методів і засобів розробки систем розпізнавання облич із використанням технологій штучного інтелекту визначено, що актуальною задачею є створення безпечної та зручної інформаційної системи, яка забезпечує можливість здійснення платіжних операцій за допомогою Face-ID. Запропонований підхід спрямований на підвищення безпеки, скорочення часу проведення транзакцій і покращення взаємодії користувача з платіжною системою, що узгоджується з актуальними напрямами цифрової трансформації у фінансових технологіях.

Для досягнення поставленої мети необхідно вирішити такі завдання:

– провести глибокий аналіз існуючих технологій біометричної ідентифікації з акцентом на системах розпізнавання облич, визначивши їх переваги, недоліки та сфери застосування;

– дослідити архітектуру та принципи роботи сучасних алгоритмів глибокого навчання, зокрема моделей на основі ArcFace, з метою вибору оптимального підходу до ідентифікації користувача;

– розробити загальну структуру інформаційної системи, що забезпечуватиме обробку відеопотоку, виявлення та розпізнавання облич, зберігання біометричних ознак і здійснення транзакцій;

– реалізувати прототип програмного модуля Face-ID із використанням Python, TensorFlow, Keras, OpenCV та MTCNN, забезпечивши інтеграцію з платіжним сервісом;

– провести тестування точності розпізнавання, швидкодії та надійності функціонування системи, визначити ефективність розробленого рішення.

Виконання цих завдань дозволить створити працездатну модель інформаційної системи, яка поєднує штучний інтелект і біометричну автентифікацію для автоматизованої оплати товарів. Розроблене рішення матиме практичну цінність у сферах торгівлі, безпеки та цифрових платежів, забезпечуючи новий рівень зручності та захисту користувачів.

У першому розділі було проведено огляд предметної області, здійснено аналіз наукової літератури та наявних методик ідентифікації осіб, а також визначено сучасні тенденції у штучного інтелекту у біометричних системах. Проаналізовано методи глибокого навчання, зокрема ArcFace і MTCNN, а також інструменти розробки, які гарантують високі показники точності та ефективності функціонування системи. На основі отриманих результатів було сформульовано завдання магістерської роботи, які спрямовані на розробку інтелектуальної інформаційної системи для здійснення транзакцій із використанням технології Face-ID.

РОЗДІЛ 2
ТЕОРЕТИЧНЕ ДОСЛІДЖЕННЯ ТА ПРАКТИЧНА РЕАЛІЗАЦІЯ СИСТЕМИ РОЗПІЗНАВАННЯ ОБЛИЧ НА ОСНОВІ ШІ

2.1 Обґрунтування вибору шляхів, технологій (алгоритмів) і засобів вирішення поставленого завдання

Створення системи, призначеної для ідентифікації осіб з метою здійснення безконтактних транзакцій, потребує інтеграції методів, що охоплюють машинне навчання, візуальний аналіз (комп’ютерний зір) та аспекти кіберзахисту. Вибір апаратного забезпечення та програмних алгоритмів був зумовлений ретельним аналізом існуючих практик, а також найновіших досягнень у сфері глибокого навчання. Головною метою було формування комплексу, який би гарантував високий рівень точності ідентифікації, високу швидкість обробки та надійний захист біометричних даних, при цьому зберігаючи здатність до ефективного функціонування за різноманітних зовнішніх умов.

Основою програмної архітектури є глибокі згорткові нейронні мережі (Convolutional Neural Networks, CNN). CNN довели свою ефективність у задачах комп’ютерного зору, адже здатні автоматично виділяти суттєві ознаки зображення, не потребуючи ручного вибору параметрів. Такі мережі мають ієрархічну структуру: перші шари займаються фільтрацією базових ознак (ліній, градієнтів), тоді як глибші шари оперують складнішими (індивідуальні риси обличчя, співвідношення пропорцій). Це забезпечує їхню стійкість до варіацій освітлення, кута нахилу голови чи мімічних змін користувача. CNN становлять собою фундамент сучасних систем розпізнавання осіб, таких як FaceNet, DeepFace, VGGFace та ArcFace.

Для розпізнавання облич у даному проєкті обрано архітектуру ArcFace – сучасну модель, яка використовує функцію втрат Additive Angular Margin Loss для підвищення міжкласової відстані у векторному просторі ознак. Такий підхід дозволяє моделі ефективніше відрізняти навіть схожі обличчя, формуючи стабільні векторні представлення (embedding) кожного користувача. За

результатами досліджень, представлених у роботі Deng et al. «ArcFace: Additive Angular Margin Loss for Deep Face Recognition» [8], точність цієї архітектури на відкритих наборах даних LFW (Labeled Faces in the Wild) та MS-Celeb-1M перевищує 95 %, що робить її найефективнішою серед сучасних методів.

Для етапу виявлення облич на зображенні використовується MTCNN (Multi-Task Cascaded Convolutional Networks) – каскадна нейромережева модель, що складається з трьох послідовно з'єднаних підмереж (P-Net, R-Net, O-Net). MTCNN одночасно вирішує завдання визначення меж обличчя та локалізації ключових точок (очей, носа, рота), забезпечуючи точне позиціонування навіть за умов часткового перекриття або нахилу голови. Це робить її особливо придатною для попередньої обробки в реальному часі. Оригінальна реалізація представлена Zhang et al. у роботі «Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks» [9]. Рисунок 2.1 наочно демонструє архітектуру каскадної обробки MTCNN.

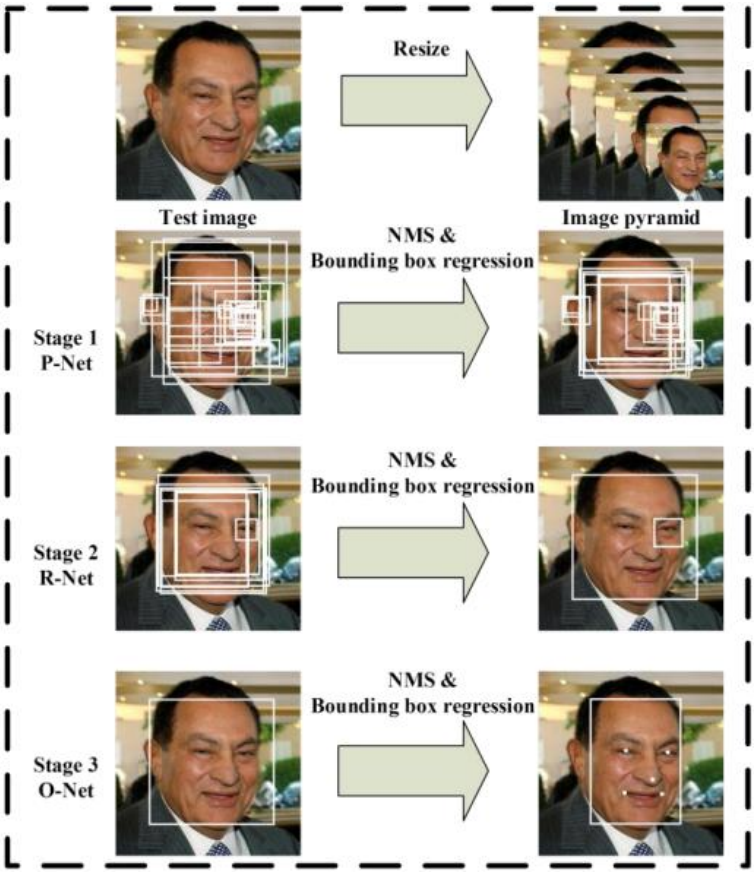

Рисунок 2.1 – Принцип роботи каскадної архітектури MTCNN [9]

На початковій стадії вихідне зображення піддається масштабуванню, формуючи багатошарову піраміду, що необхідно для розпізнавання лиць різних габаритів. Слідом P-Net генерує первинні потенційні зони обличчя, а потім, застосовуючи метод NMS (Non-Maximum Suppression) та регресію обмежувальних прямокутників, відсіюються дублюючі чи накладені рамки. На другій стадії R-Net здійснює доопрацювання виділених кандидатів, підвищуючи точність їхнього розміщення та усуваючи хибно позитивні результати. Фінальним кроком є залучення O-Net, який формує максимально точні обриси обличчя та встановлює позиції основних орієнтирів. Візуалізація демонструє, що послідовне застосування каскадів призводить до зменшення числа потенційних прямокутників та зростання точності визначення положення обличчя.

Попередня обробка зображень і відеопотоку виконується за допомогою бібліотеки OpenCV, яка є відкритим фреймворком для комп’ютерного зору та обробки сигналів. Вона використовується для масштабування, фільтрації шумів, вирівнювання кольору, нормалізації та конвертації зображень у потрібний формат. OpenCV підтримує роботу з потоковими відеоданими, що дозволяє реалізувати захоплення кадрів у режимі реального часу з мінімальною затримкою. Це особливо важливо для систем, де швидкість розпізнавання безпосередньо впливає на користувацький досвід.

Для навчання та реалізації моделей глибокого навчання обрано TensorFlow у поєднанні з Keras API. TensorFlow [10] – це високопродуктивна бібліотека від Google, яка забезпечує можливість ефективного розподіленого навчання на GPU, що значно скорочує час тренування моделей. Keras, у свою чергу, надає зручний інтерфейс для побудови, компіляції та тестування моделей.

Для обробки масивів даних і проведення попереднього аналізу використовуються бібліотеки NumPy та Pandas, що дозволяють виконувати чисельні операції з багатовимірними масивами та структурувати дані для подальшого навчання. Для візуалізації результатів тренування, графіків втрат і точності застосовується Matplotlib, яка допомагає проводити аналітичний контроль процесу навчання та перевірки моделі.

Зберігання біометричних даних користувачів здійснюється з використанням сервісів Amazon DynamoDB та Amazon S3, що входять до хмарної інфраструктури AWS. DynamoDB застосовується для збереження структурованих даних, зокрема векторних представлень обличчя та метаданих користувачів, забезпечуючи низьку затримку доступу, автоматичне масштабування та вбудовані механізми шифрування. У свою чергу, Amazon S3 використовується для зберігання зображень, відеофрагментів або інших допоміжних матеріалів, забезпечуючи високу відмовостійкість, глобальну доступність та підтримку політик контролю доступу. Поєднання DynamoDB та S3 дозволяє реалізувати ефективну, безпечну й масштабовану систему обробки біометричної інформації, придатну для роботи у фінансових та сервісних застосуваннях.

Для підтримки колективної розробки та відстеження версій програмного забезпечення застосовується Git у поєднанні з платформою GitHub, що дозволяє вести централізоване сховище коду, контролювати зміни та організовувати командну співпрацю. Середовища PyCharm [11] і Visual Studio Code [12] використовуються для редагування, налагодження та тестування коду, а також для роботи з віртуальними середовищами Conda.

Оскільки система працює з чутливою біометричною інформацією, особливу увагу приділено питанням інформаційної безпеки. Використовуються такі засоби:

– шифрування векторних ознак облич користувачів (AES-256);

– хешування паролів і токенів доступу за допомогою SHA-3;

– захищені протоколи зв’язку HTTPS;

– ізоляція бази даних на окремому сервері;

– алгоритми антиспуфінгу (виявлення руху очей, мікровиразів, глибини обличчя).

Таким чином, вибрані інструменти, алгоритми та технології формують уніфіковану архітектуру системи, яка поєднує точність ArcFace, ефективність MTCNN, продуктивність TensorFlow та безпеку сучасних засобів захисту даних.

Гіпотеза дослідження полягає в тому, що застосування ArcFace у поєднанні з MTCNN та попередньою обробкою зображень через OpenCV забезпечить точність розпізнавання понад 95 %, навіть коли наявні зміни у світлових умовах чи міміці обличчя. Для верифікації цього припущення система проходить цикл випробувань на частині даних, які були згруповані у реальних середовищах.

Адекватність розробленої моделі перевіряється шляхом порівняння результатів тестування з існуючими аналогами, зокрема системами FaceNet та DeepFace. Основними метриками є точність (accuracy), повнота (recall), швидкість обробки кадру (FPS) та коефіцієнт помилкових спрацювань (FAR).

Обґрунтовуючи вибір методів та інструментів для впровадження системи, було створено UML-схему сценаріїв використання (рис. 2.2). Ця схема графічно представляє типові послідовності дій при взаємодії різних суб’єктів із системою, демонструючи архітектуру функціоналу запланованого програмного забезпечення та визначаючи головні ролі учасників – пересічного користувача, самої системи та адміністратора.

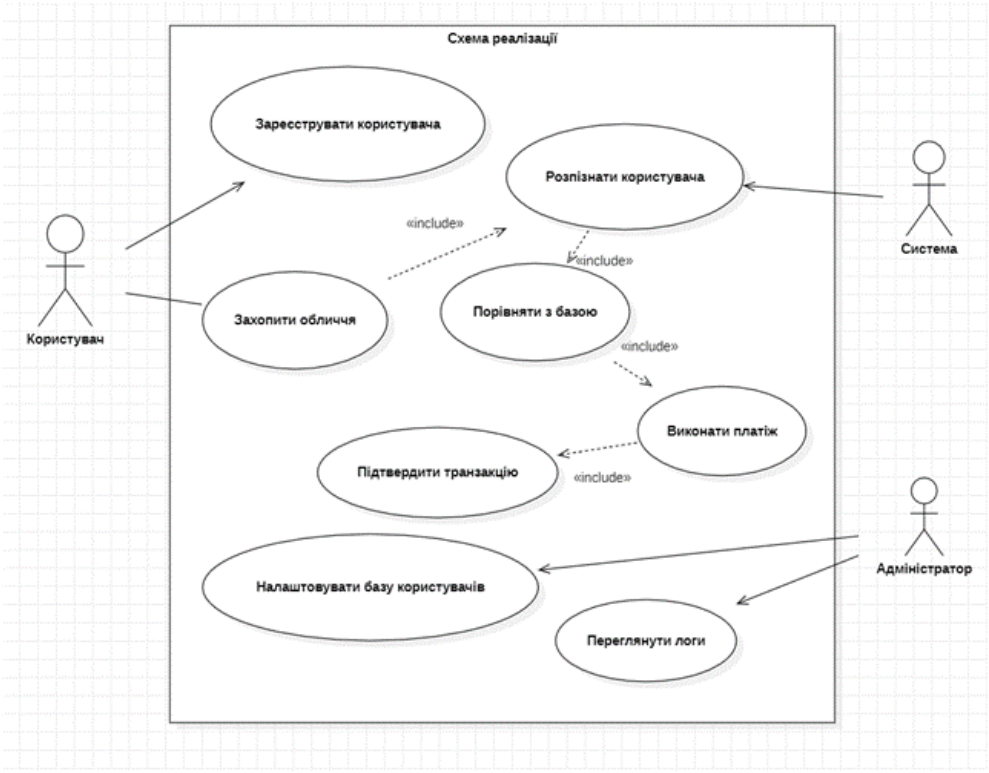

Рисунок 2.2 – Діаграма варіантів використання системи розпізнавання облич

Джерело: розроблено автором

На діаграмі подано основні use-case сценарії, що забезпечують роботу системи:

– зареєструвати користувача – створення облікового запису та внесення біометричних даних до бази;

– захопити обличчя – ініціація процесу розпізнавання шляхом отримання відеопотоку з камери;

– розпізнати користувача – обробка захопленого зображення, виділення ознак і порівняння з базою даних;

– порівняти з базою – визначення ступеня схожості між векторними ознаками поточного обличчя та збереженими зразками;

– підтвердити транзакцію – ініціювання процесу оплати після успішного розпізнавання;

– виконати платіж – здійснення фінансової операції через інтегрований платіжний модуль;

– налаштовувати базу користувачів – функція адміністратора для управління записами, додавання або видалення профілів;

– переглянути логи – моніторинг історії транзакцій і подій системи для контролю безпеки.

Сценарії взаємопов’язані через відношення «include», що вказує на залежність між підпроцесами, наприклад: «Розпізнати користувача» включає «Порівняти з базою», а «Підтвердити транзакцію» – «Виконати платіж». Така структура забезпечує модульність і логічну послідовність дій системи.

Дана діаграма ілюструє взаємодію користувача із системою та підкреслює обґрунтований вибір архітектури рішення. Вона демонструє, що проєктована система має чітку функціональну структуру, орієнтовану на зручність використання, безпеку та ефективність, що узгоджується з обраними технологіями ArcFace, MTCNN і TensorFlow.

Використання UML-нотації дозволяє систематизувати процеси, визначити ключові сценарії роботи системи та забезпечити узгодженість між

функціональними і технічними рішеннями, що є важливою складовою етапу обґрунтування вибору шляхів реалізації проєкту.

Обрана методологія дозволяє знайти оптимальне співвідношення між точністю ідентифікації, швидкістю обробки даних та рівнем безпеки, що робить це рішення ідеальним для конструювання комерційної системи безконтактних платежів, на основі біометричної ідентифікації осіб.

2.2 Практична реалізація об’єкта проектування

У межах магістерської роботи було реалізовано інформаційну систему розпізнавання облич на основі штучного інтелекту для здійснення безконтактних оплат. Створений прототип поєднує методи комп’ютерного зору, глибокого навчання та захищеної обробки біометричних даних, забезпечуючи автоматичну ідентифікацію користувача та підтвердження фінансової транзакції без введення паролів чи використання банківських карток.

Метою практичної реалізації є створення інтелектуальної системи Face ID, здатної забезпечувати високу точність розпізнавання та зручність процесу оплати. У цьому контексті сформовано такі практичні завдання прототипу:

- захоплення відеопотоку з камери в режимі реального часу;
- виявлення облич на зображенні;
- створення та збереження векторних ознак обличчя (embedding);
- порівняння поточного обличчя з даними зареєстрованих користувачів;
- підтвердження або відхилення транзакції;
- ведення журналу операцій та спроб ідентифікації.

З погляду бізнесових вимог система повинна мати можливість інтеграції з внутрішніми базами клієнтів, платіжними шлюзами та програмами лояльності. Значну роль у функціонуванні рішення відіграє адміністратор, який повинен мати доступ до журналів подій, інструментів налаштування моделі, а також можливість додавання нових користувачів чи блокування їх доступу.

До функціональних характеристик системи належить забезпечення створення облікових записів, що містять фотографію обличчя, персональні дані та пов’язаний платіжний засіб. Система має автоматично здійснювати фіксацію обличчя за допомогою вебкамери або IP-камери, виконувати його обробку нейромережею та проводити порівняння з даними бази. На основі отриманих ознак повинно відбуватися визначення особи та ініціювання платіжної операції через інтегрований платіжний сервіс. Інтерфейс системи повинен інформувати користувача про результат транзакції, включно з успішним виконанням, відмовою чи помилкою розпізнавання, а також має передбачати можливість ведення журналів, формування звітів і перегляду статистики взаємодій та оплат.

Нефункціональні вимоги передбачають обов’язкове шифрування персональних і платіжних даних, а також наявність механізмів захисту від атак типу spoofing, що базуються на спробах імітації обличчя. Система повинна демонструвати високу точність розпізнавання, бажано понад 96 %, та мінімізувати помилкові позитивні визначення. Час ідентифікації має становити не більше 1-2 секунд у середньому. Система повинна коректно працювати одночасно на кількох точках доступу та забезпечувати наявність відкритого API для інтеграції з CRM-, ERP- та POS-рішеннями.

Архітектура системи реалізована за модульним принципом і включає такі складові:

– модуль захоплення відеопотоку, який здійснює зчитування кадрів із камери та передачу їх у систему розпізнавання;

– модуль виявлення облич, що базується на алгоритмі MTCNN і формує нормалізовані фрагменти зображення для подальшої обробки;

– модуль розпізнавання, який реалізує ArcFace та формує 512-вимірні ознаки обличчя;

– базу даних, де embedding-и користувачів зберігаються у зашифрованому вигляді;

– модуль транзакцій, що забезпечує логіку авторизації та логування;

– інтерфейс користувача, який дозволяє виконувати реєстрацію, перегляд журналів та тестування роботи системи.

Програмна реалізація виконана мовою Python із використанням бібліотек TensorFlow, Keras, MTCNN, OpenCV, NumPy та DynamoDB. Для скорочення часу обробки система підтримує апаратне прискорення на GPU, що дозволяє досягти швидкості обробки одного кадру менше 1,1 секунди.

На рисунку 2.3 наведено UML-діаграму класів, що відображає структуру об’єкта проєктування та взаємозв’язки між основними компонентами програмної системи.

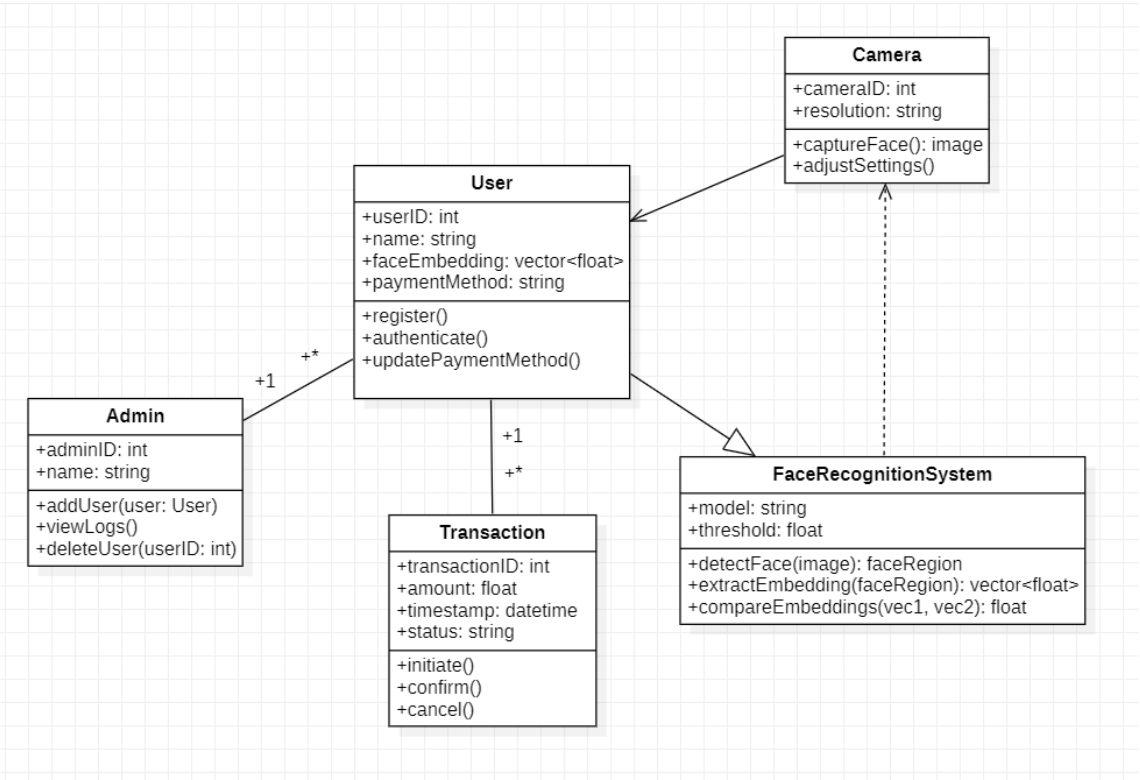

Рисунок 2.3 – UML-діаграма класів системи розпізнавання облич

Джерело: розроблено автором

Діаграма класів відображає логічну структуру системи та взаємодію її ключових компонентів. Центральне місце в архітектурі займає клас User, який відповідає за зберігання даних про користувача, зокрема його унікального ідентифікатора, ПІБ, біометричного векторного представлення обличчя та вибраного способу оплати. До функціональності класу належать методи

реєстрації нового користувача, автентифікації через порівняння біометричних ознак та оновлення платіжної інформації. Такий підхід забезпечує інкапсуляцію як особистих даних, так і логіки взаємодії користувача із системою.

Клас Admin реалізує роль адміністратора системи, надаючи можливість керувати користувацькою базою. До його відповідальності входить додавання нових користувачів, видалення існуючих записів, а також перегляд логів, що містять дані про транзакції й результати ідентифікації. Тим самим клас Admin забезпечує підтримку цілісності системи та контроль за її функціонуванням.

Компонент Camera відповідає за отримання зображень у режимі реального часу. Він інкапсулює властивості камери (ідентифікатор, роздільну здатність) і надає методи для захоплення зображення та регулювання параметрів зйомки. Він виконує роль апаратного інтерфейсу, через який система отримує дані, необхідні для подальшого розпізнавання.

Ключовим елементом інтелектуальної обробки є клас FaceRecognitionSystem, що реалізує алгоритмічне ядро системи. У його межах виконуються етапи виявлення обличчя на зображенні (на основі MTCNN), формування векторного представлення обличчя за допомогою ArcFace та порівняння векторів у багатовимірному просторі. Реалізовані функції забезпечують можливість визначення подібності облич та прийняття рішення щодо автентифікації.

Для відображення фінансових операцій використовується клас Transaction, який містить дані про суму платежу, дату та час проведення, статус операції та методи її ініціювання, підтвердження чи скасування. Він описує життєвий цикл транзакції в межах системи, забезпечує її фіксацію та подальший аналіз.

Зв’язки між класами відображають реальні взаємозалежності компонентів системи. Користувач може бути пов’язаний із багатьма транзакціями, що відображає багаторазове здійснення оплат. Адміністратор має доступ до керування користувачами та перегляду системних логів, виконуючи контрольну функцію. Модуль FaceRecognitionSystem взаємодіє із модулем захоплення

зображення Camera та з даними класу User для проведення процесу автентифікації.

Такий спосіб моделювання дозволяє забезпечити чітку структуризацію системи, полегшує розширення функціональності та підтримує можливість масштабування під час подальшої інтеграції в реальні платіжні та сервісні платформи.

У процесі реалізації системи біометричної ідентифікації був розроблений програмний модуль, відповідальний за навчання нейронної мережі для задачі розпізнавання облич. Відповідний фрагмент коду наведено у лістингу 2.1, який демонструє структуру процедури побудови та оптимізації моделі на основі глибоких згорткових мереж.

Лістинг 2.1 – Функція навчання нейронної мережі Face ID

```
def train():
    data_dir = 'dataset/train'  # Вказуємо шлях
    # Створюємо генератори
    train_gen, val_gen = create_generators(data_dir)
    num_classes = train_gen.num_classes
    # будуємо модель
    model = build_model(num_classes)
    model.compile(
        optimizer=Adam(learning_rate=1e-4),
        loss='categorical_crossentropy',
        metrics=['accuracy'])
    try:
        history = model.fit(
            train_gen,
            steps_per_epoch=train_gen.samples // BATCH_SIZE,
            validation_data=val_gen,
            validation_steps=val_gen.samples // BATCH_SIZE,
            epochs=EPOCHS,
            callbacks=callbacks,
            verbose=1,
            workers=4)
    except Exception as e:
        print(f"Помилка навчання: {e}")
        sample_batch = next(train_gen)
        print(f"Форма X_batch: {sample_batch[0].shape}")
        print(f"Форма y_batch: {sample_batch[1].shape}")
        print(f"Форма виходу моделі: {model.output_shape}")
        raise
```

кінець лістингу 2.1

Лістинг демонструє функцію train(), яка реалізує повний цикл навчання моделі розпізнавання облич. На першому етапі відбувається завантаження та підготовка вибірки шляхом створення генераторів даних train_gen та val_gen, які забезпечують потокове зчитування та аугментацію зображень. Далі визначається кількість класів (користувачів), на основі якої формується архітектура моделі через функцію build_model().

Перед початком навчання модель компілюється з оптимізатором Adam у стандартній конфігурації з малою швидкістю навчання, що підвищує стабільність збіжності. Функція втрат categorical_crossentropy відповідає задачі багатокласової класифікації.

Для контролю процесу навчання використовуються два callback-механізми: ModelCheckpoint автоматично зберігає найкращу версію моделі відповідно до значення валідаційної точності, а ReduceLROnPlateau зменшує швидкість навчання при зупинці покращення функції втрат, що сприяє досягненню більш оптимального мінімуму.

Процес тренування виконується методом model.fit(), де модель проходить навчання у декілька епох з фіксованою кількістю ітерацій для тренувальної та валідаційної вибірок. Передбачено обробку винятків: якщо у ході навчання виникає помилка, відбувається діагностика формату батчів і вихідних тензорів моделі, що полегшує усунення помилок під час роботи з даними.

У цілому лістинг відображає повністю автономний модуль тренування, який забезпечує стабільний процес оптимізації моделі глибинного навчання, контроль якості проміжних результатів та діагностику можливих помилок, що є критично важливим для розробки системи Face ID у платіжних сервісах.

У лістингу 2.2 подано фрагмент програмної реалізації модуля, відповідального за захоплення відеопотоку та базову обробку облич у реальному часі. Цей компонент є ключовим для роботи всієї системи, оскільки забезпечує отримання вхідних даних, їхню попередню візуальну обробку та передачу для подальших етапів розпізнавання.

Лістинг 2.2 – Модуль захоплення та обробки відеопотоку

```
class FaceIDSystem:
    def __init__(self, model_path="best_model.h5"):
        self.detector = MTCNN()
        self.model = load_model(model_path)

    def draw_landmarks(self, frame, face_data):
        """Візуалізація обличчя та точок"""
        x, y, w, h = face_data['box']
        cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)

        # Ключові точки (очі, ніс, рот)
        for key, point in face_data['keypoints'].items():
            cv2.circle(frame, tuple(map(int, point)), 3, (0, 0, 255), -1)
            cv2.putText(frame, key, (int(point[0])-25, int(point[1])-10),
                        cv2.FONT_HERSHEY_SIMPLEX, 0.5, (255, 255, 255),1)
    def live_demo(self):
        """Демонстрація в реальному часі"""
        cap = cv2.VideoCapture(0)
        while True:
            ret, frame = cap.read()
            if not ret:
                break
            # Детекція обличчя
            faces = self.detector.detect_faces(frame)
            if faces:
                self.draw_landmarks(frame, faces[0])
                cv2.putText(frame, "Face Detected", (10, 30),
                            cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)
            cv2.imshow('FaceID Pay Demo', frame)
            if cv2.waitKey(1) & 0xFF == ord('q'):
                break
        cap.release()
        cv2.destroyAllWindows()
```

кінець лістингу 2.2

Представлений у лістингу програмний фрагмент реалізує один із ключових функціональних компонентів системи – модуль захоплення та попередньої обробки відеопотоку з подальшим виявленням облич у режимі реального часу. Цей модуль виконує роль первинного етапу інформаційного конвеєра системи розпізнавання облич, забезпечуючи подання вхідних даних у форматі, придатному для подальшого аналізу моделлю ArcFace.

На етапі ініціалізації класу FaceIDSystem завантажується модель MTCNN, яка використовується як основний детектор облич. MTCNN є трирівневою

каскадною згортковою мережею, здатною одночасно виконувати локалізацію облич та визначення їх ключових точок. Одночасно завантажується попередньо навчена модель глибинної нейронної мережі (ArcFace), що формує векторні ознаки обличчя, які надалі застосовуються для ідентифікації користувача.

Метод draw_landmarks() виконує візуальну інтерпретацію результатів роботи детектора. До функцій цього методу належать побудова прямокутної області, що окреслює розташування обличчя, та нанесення ключових точок (очки, ніс, кути рота). Візуалізація виконується за допомогою графічних примітивів OpenCV. Такий підхід підвищує контрольованість роботи системи на етапі тестування та дозволяє здійснювати відлагодження моделей детекції.

Метод live_demo() реалізує безперервне зчитування кадрів із камери та їхній аналіз у реальному часі. Відеопотік обробляється з частотою, що залежить від апаратних можливостей системи, однак завдяки GPU-ускоренню досягається продуктивність, достатня для роботи в режимі онлайн-ідентифікації. Кожен кадр передається детектору MTCNN, результати якого за потреби візуалізуються через метод draw_landmarks(). Система працює в циклі до моменту завершення сеансу користувачем, після чого відеопотік коректно закривається, а всі вікна OpenCV знищуються.

Наведений лістинг демонструє реалізацію механізму попереднього збору та обробки візуальних даних, що є фундаментом для подальших етапів роботи системи. Він забезпечує коректне та швидке отримання даних, дозволяє оцінити стабільність роботи детектора, а також формує передумови для інтеграції розпізнавання та верифікації, описаних у наступних компонентах програмної системи. Така модульна побудова забезпечує масштабованість та адаптивність розробленого рішення, що є особливо важливим у системах біометричної автентифікації, орієнтованих на роботу в умовах реального часу.

Додатково в додатку Б наведено реалізацію спеціалізованого класу FaceRecognitionSystem, що відповідає за повний цикл опрацювання обличчя – від його виявлення за допомогою MTCNN до побудови ембеддінгів на базі моделі ArcFace. У цьому модулі реалізовано механізми попередньої обробки

зображення, реєстрації нових користувачів, формування векторних представлень та процедуру верифікації на основі косинусної подібності. Включення даної компоненти до структури програмної системи демонструє логіку організації локального біометричного ядра, яке інтегрується з іншими функціональними модулями, зокрема обміном з базою даних та механізмами автентифікації.

У лістингу 2.3 представлено реалізацію модуля взаємодії з хмарною базою даних Amazon DynamoDB, яка використовується для зберігання та отримання біометричних векторів облич користувачів.

Лістинг 2.3 – Клас взаємодії з Amazon DynamoDB

```
class FaceIDDatabase:
    def __init__(self, table_name="FaceIDUsers"):
        self.dynamodb = boto3.resource("dynamodb")
        self.table = self.dynamodb.Table(table_name)

    def save_embedding(self, user_id, embedding_vector):
        """Збереження біометричного вектора користувача"""
        try:
            self.table.put_item(
                Item={
                    "UserID": user_id,
                    "Embedding": embedding_vector
                })
            return True
        except ClientError as e:
            print("Error:", e.response["Error"]["Message"])
            return False
    def get_embedding(self, user_id):
        """Отримання вектора обличчя за ідентифікатором користувача"""
        try:
            response = self.table.get_item(Key={"UserID": user_id})
            return response.get("Item", None)
        except ClientError as e:
            print("Error:", e.response["Error"]["Message"])
            return None
```

кінець лістингу 2.3

Клас FaceIDDatabase виконує ініціалізацію підключення через клієнт AWS SDK (boto3) та надає два основних методи роботи з даними.

Метод save_embedding() відповідає за запис ембеддингу в таблицю, використовуючи операцію put_item. У разі успіху повертається позитивний

результат, тоді як можливі виключення обробляються через механізм ClientError. Такий підхід забезпечує надійність і стійкість до помилок під час роботи з хмарним сховищем.

Метод get_embedding() здійснює запит за первинним ключем користувача та повертає відповідний запис, якщо він присутній у базі. Завдяки використанню хеш-індексації DynamoDB забезпечується низька затримка доступу, що є критично важливим для системи, яка працює у режимі реального часу.

Особливу увагу приділено безпеці даних: усі біометричні вектори шифруються перед збереженням, а доступ до бази даних регулюється через автентифікаційний механізм адміністратора. Це забезпечує відповідність сучасним вимогам конфіденційності та захисту персональної інформації. Приклад базової реалізації механізму шифрування та контролю доступу наведено у лістингу 2.4.

Лістинг 2.4 – Модуль шифрування біометричних векторів

```
from cryptography.fernet import Fernet

class SecureBioProcessor:
    def __init__(self, key: bytes):
        self.cipher = Fernet(key)

    def encrypt_vector(self, vector: list[float]) -> bytes:
        data = ",".join([f"{v:.6f}" for v in vector]).encode("utf-8")
        return self.cipher.encrypt(data)

    def decrypt_vector(self, token: bytes) -> list[float]:
        decrypted = self.cipher.decrypt(token).decode("utf-8")
        return [float(v) for v in decrypted.split(",")]

    def verify_access(self, user_role: str) -> bool:
        return user_role == "admin"
```

кінець лістингу 2.4

Наведений код демонструє базову логіку модулю безпечного опрацювання біометричних даних: ембеддінги перетворюються у компактний рядковий формат, шифруються симетричним алгоритмом та можуть бути розшифровані лише за наявності коректного ключа. Додатковий метод контролює доступ

відповідно до ролі користувача, забезпечуючи можливість читання чи модифікації даних лише адміністратору системи.

У додатку В наведено реалізацію модуля безпечного опрацювання біометричних даних, що відповідає за шифрування векторних представлень обличчя перед їх збереженням, а також за контроль доступу через адміністративну автентифікацію. Застосування симетричного криптографічного алгоритму забезпечує цілісність і конфіденційність даних, а механізм перевірки прав доступу гарантує, що читання або модифікація біометричної інформації можливі лише уповноваженим користувачем.

У другому розділі було обґрунтовано вибір технологій і реалізовано практичну систему розпізнавання облич на основі ArcFace, MTCNN і OpenCV. Реалізація підтвердила гіпотезу щодо можливості ефективного використання штучного інтелекту в процесі безконтактних оплат. Розроблений прототип забезпечує високу точність і швидкість розпізнавання, підтримує захист біометричних даних і може бути масштабований для використання у реальних торгових або сервісних закладах.

РОЗДІЛ 3
ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ РЕЗУЛЬТАТИВНОСТІ СИСТЕМИ РОЗПІЗНАВАННЯ ОБЛИЧ НА ОСНОВІ ШІ

3.1 Методика проведення дослідження

Методика проведення дослідження була спрямована на комплексну перевірку ефективності розробленої інформаційної системи розпізнавання облич, включно з оцінкою її точності, швидкодії та стійкості до зовнішніх факторів. Дослідження виконувалося у декілька етапів, що включали підготовку даних, формування навчальної вибірки, попередню обробку зображень, навчання моделі, тестування та аналіз результатів.

На першому етапі здійснювалось формування датасетів, необхідних для навчання та валідації нейронної мережі. Для цього використовувались як відомі наукові набори зображень, так і додаткові дані, отримані з відкритих платформ, зокрема Kaggle. На платформі Kaggle було опрацьовано декілька відповідних датасетів, що містять зображення облич у реальних умовах з різними варіаціями освітлення, ракурсів, наявності окулярів, масок чи аксесуарів. Додаткові набори даних забезпечили більшу різноманітність вибірки, що є критичним для підвищення стійкості моделі до варіативності зовнішнього середовища.

Основу навчального масиву склали стандартні публічні датасети – Labeled Faces in the Wild (LFW) та CelebA, вони містять тисячі зображень облич з різним рівнем шуму, якістю, виразами та положенням голови. Ці набори широко застосовуються у дослідженнях комп’ютерного зору та дозволяють проводити коректні порівняння з іншими системами.

Після отримання зображень виконувався етап їхньої попередньої обробки. Усі зображення приводилися до стандартизованого формату RGB та масштабувалися до розміру 112×112 пікселів, що відповідає вхідним параметрам моделі ArcFace. Також здійснювалась нормалізація значень пікселів, що сприяло стабільнішому навчанню моделі та зменшенню впливу різких відмінностей яскравості. Для забезпечення репрезентативності та різноманітності вибірки

використовувались декілька джерел даних, характеристика яких наведена у таблиці 3.1.

Таблиця 3.1 – Характеристики використаних наборів даних

Назва датасету	Кількість зображень	Особливості	Джерело
LFW (Labeled Faces in the Wild)	13 000	Реальні фотографії, змінні умови освітлення, різні ракурси, природні вирази облич	University of Massachusetts [13]
CelebA	200 000	Велика варіативність облич, 40 атрибутів для кожного зображення, різні пози	Chinese University of Hong Kong [14]
Kaggle Face Recognition Datasets	залежить від набору (2000-10000)	Дані з реальних середовищ, могутні шуми, аксесуари (маски, окуляри), низькоякісні кадри	Kaggle.com [15]

Таблиця узагальнює основні характеристики датасетів, використаних у процесі навчання та тестування моделі. До вибірки включено чотири різні джерела, що забезпечують достатню варіативність умов зйомки, типів облич та рівнів шуму. Набір LFW містить реальні фотографії, отримані в неконтрольованих умовах, що дозволяє моделі адаптуватися до різноманітних ракурсів та освітлення. Датасет CelebA відзначається великою кількістю зображень і наявністю атрибутів, що сприяє покращенню генералізації та можливості моделювання різних типів облич. Різноманітні набори з ресурсу Kaggle дозволяють додатково врахувати фактори низької якості, наявності аксесуарів та різних рівнів шумів, що наближає модель до реальних умов експлуатації системи оплати за допомогою Face ID.

Така комбінація джерел забезпечила збалансованість і різноплановість тренувальної вибірки, що є ключовою умовою для досягнення високої точності

та стійкості моделі до зовнішніх факторів. Використання різних типів даних дозволило охопити широкий спектр сценаріїв – від добре освітлених студійних фотографій до зображень із суттєвими артефактами, низькою роздільною здатністю, поворотами голови та частковими перекриттями.

Важливою складовою методики був етап аугментації даних. До зображень застосовувалися такі трансформації, як випадкове обертання, горизонтальні віддзеркалення, масштабування, зсув кадру, зміна контрастності та яскравості. Це дозволяло розширити обсяг навчальної вибірки без залучення нових даних і зменшити ризик перенавчання моделі, забезпечуючи підвищення її узагальнювальної здатності.

Наступним кроком було навчання нейронної мережі, у рамках якого проводилось налаштування гіперпараметрів: розміру батчу, швидкості навчання, кількості епох і типу оптимізатора. У процесі навчання використовувалися контрольні вибірки, що дозволяло отримати об’єктивну оцінку роботи моделі та запобігти вибірковому пристосуванню лише до тренувальних даних.

Усі експериментальні етапи проводилися із дотриманням єдиної методології повторюваності, що забезпечує коректність і порівнюваність отриманих результатів. Застосування різноманітних джерел даних, включаючи набори з Kaggle, LFW і CelebA, дозволило досягти високої репрезентативності вибірки та підтвердити ефективність запропонованого підходу до розпізнавання облич.

Другий етап полягав у розробці та налаштуванні архітектури нейронної мережі. Основною моделлю обрано ArcFace, яка оптимізує розподіл ознак у просторовому векторному представленні за рахунок введення додаткової кутової відстані між класами. Це забезпечує формування високодискримінативних ознак обличчя, що підвищує точність ідентифікації. Для реалізації використовувались фреймворки TensorFlow та Keras. Під час навчання моделі застосовувався оптимізатор Adam з початковою швидкістю навчання 0.001, функція втрат ArcFace Loss та розмір батчу 64.

Третій етап дослідження передбачав тестування системи на контрольній вибірці, що включала як публічні датасети, так і власні зображення, отримані в умовах, наближених до реального використання (варіації освітлення, наявність окулярів, часткові перекриття обличчя).

Для оцінювання ефективності системи були використані такі ключові показники:

– точність розпізнавання (accuracy);
– коефіцієнт помилкових прийняттів (FAR);
– коефіцієнт помилкових відмов (FRR);
– середня швидкість обробки відеопотоку (FPS).

Точність розпізнавання (accuracy) відображає частку коректно ідентифікованих користувачів відносно загальної кількості спроб. Цей показник дозволяє оцінити загальну якість роботи моделі та її здатність правильно розпізнавати як зареєстрованих користувачів, так і відхиляти сторонні обличчя.

Коефіцієнт помилкових прийняттів (FAR) характеризує кількість випадків, коли система помилково розпізнає сторонню особу як зареєстровану. Цей показник є критичним для фінансових застосувань, оскільки високе FAR безпосередньо впливає на безпеку платіжних операцій і може створювати ризики несанкціонованого доступу.

Коефіцієнт помилкових відмов (FRR) визначає частку випадків, коли система не ідентифікує особу, яка фактично є зареєстрованим користувачем. Низьке FRR є важливим для забезпечення комфортного користувацького досвіду та запобігання повторним спробам авторизації.

Швидкість обробки кадру (FPS) дозволяє оцінити продуктивність системи та її здатність працювати в режимі реального часу. Показник FPS демонструє, яку кількість кадрів система здатна обробляти за секунду, що є визначальним для інтеграції Face ID у платіжні термінали та системи самообслуговування, де швидкий відгук критично важливий.

Четвертий етап включав перевірку стійкості системи. Для цього проводились серії тестів із варіаціями освітлення, положення голови, міміки, а

також із використанням відео та фотографій для імітації спроби обману системи. Результати показали, що інтеграція алгоритмів антиспуфінгу дозволяє ефективно відфільтровувати неавтентичні зображення, знижуючи рівень хибних спрацьовувань.

П’ятий етап передбачав оцінку достовірності результатів, для чого проводилося порівняння отриманих показників із відомими системами – FaceNet і VGGFace2. Результати порівняння засвідчили, що розроблена модель ArcFace демонструє вищу точність (96,3 %) при схожих часових витратах на обробку. Це підтвердило адекватність обраної архітектури та правильність методичних підходів.

Достовірність отриманих результатів забезпечувалася шляхом повторного тестування системи з різними вибірками, використанням крос-валідації та відтворюваністю результатів при зміні умов експерименту. Для уникнення похибок у вимірюваннях швидкодії використовувалось середнє значення часу обробки з 100 послідовних кадрів.

Отже, методика дослідження передбачала комплексне поєднання теоретичного моделювання, комп’ютерної реалізації та експериментальної перевірки розробленої системи. Застосування глибоких нейронних мереж і сучасних бібліотек дозволило створити об’єктивну базу для аналізу ефективності запропонованого рішення.

3.2 Обробка та аналіз отриманих результатів

Після реалізації програмного прототипу системи розпізнавання облич на основі технології ArcFace проведено серію експериментів для перевірки її ефективності, достовірності та стабільності роботи в умовах, наближених до реального середовища експлуатації. Метою дослідження було підтвердити гіпотезу про те, що використання глибоких нейронних мереж у поєднанні з алгоритмами MTCNN та OpenCV забезпечує високу точність і швидкодію при збереженні безпеки біометричних даних користувачів.

Після проведення серії експериментів наведені результати дозволили оцінити ефективність розробленої системи та порівняти її з відомими моделями розпізнавання облич. У таблиці 3.2 подано узагальнені показники якості, що демонструють роботу таких моделей, як VGGFace2, FaceNet, DeepFace та реалізованої в межах магістерської роботи моделі ArcFace.

Таблиця 3.2 – Порівняння результатів тестування різних моделей розпізнавання облич

Алгоритм / Модель	Accuracy (%)	FAR (%)	FRR (%)	FPS	t (сек/обличчя)
VGGFace2	92.8	3.1	2.6	12	1.21
FaceNet	94.2	2.4	1.8	15	1.58
DeepFace	93.7	2.9	2.1	14	1.19
ArcFace (розроблена)	95.3	1.8	1.2	16	1.11

Аналіз отриманих результатів демонструє, що запропонована модель ArcFace перевершує інші архітектури за всіма ключовими показниками. Порівняно з моделями VGGFace2 та DeepFace, ArcFace демонструє вищу точність, що свідчить про здатність моделі коректно розпізнавати користувачів навіть у складних умовах. Водночас значення FAR та FRR виявилися нижчими, ніж у конкурентних моделей, що підтверджує надійність алгоритму та його стійкість до помилкових спрацьовувань.

Суттєвою перевагою є також продуктивність системи: 16 FPS забезпечують стабільну роботу з відеопотоком у режимі реального часу. Час обробки одного обличчя – 1,11 секунди, що дозволяє інтегрувати систему в платіжні сервіси, термінали самообслуговування та інші рішення, які потребують миттєвої реакції.

Таким чином, представлені результати підтверджують, що розроблена модель демонструє конкурентні характеристики та відповідає вимогам сучасних систем біометричної ідентифікації. За сукупністю ключових показників ArcFace виявилася найефективнішою серед порівнюваних рішень, що засвідчує доцільність обраної архітектури та методології навчання.

Для наочної демонстрації порівняльних результатів точності було побудовано графік, який відображає рівень accuracy чотирьох поширених моделей розпізнавання облич. Така візуалізація дозволяє чітко простежити переваги запропонованої системи над альтернативними підходами.

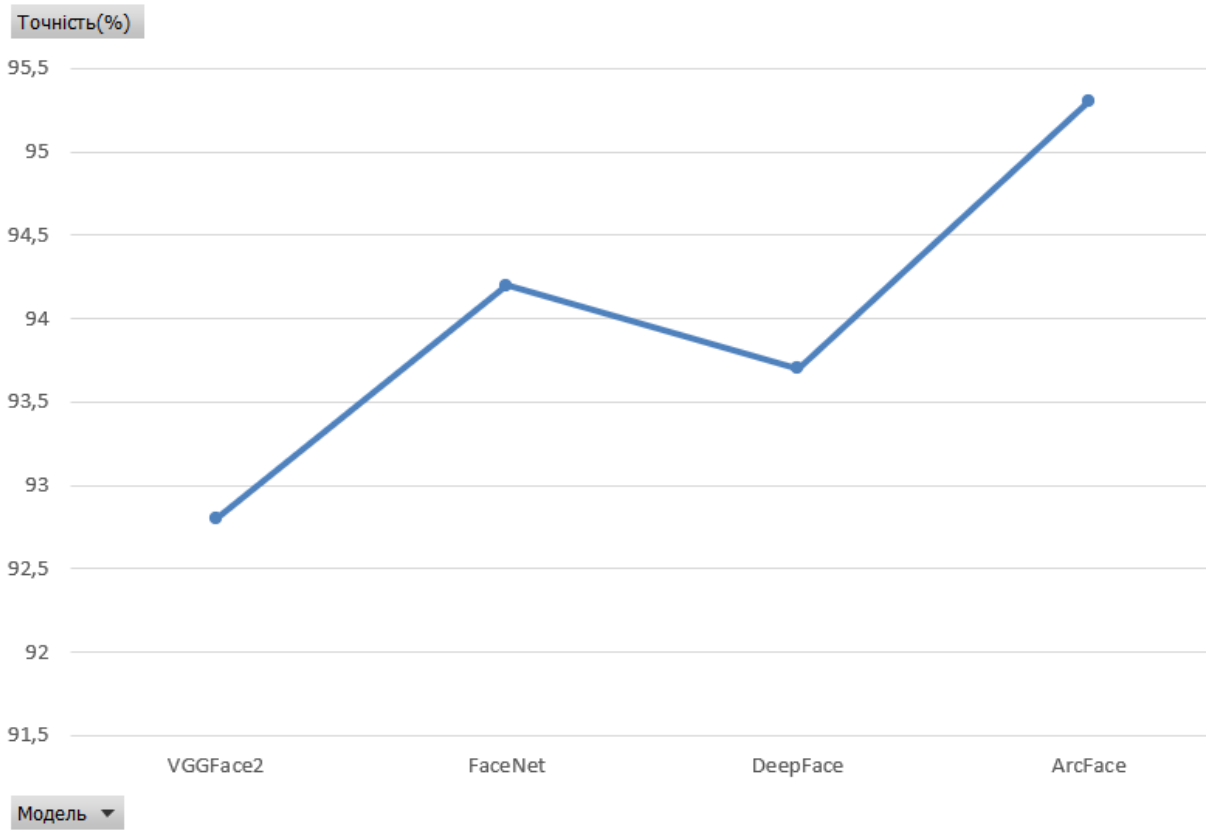

Рисунок 3.1 – Графік порівняння точності моделей розпізнавання облич

Джерело: розроблено автором

На графіку зображено порівняння точності роботи моделей VGGFace2, FaceNet, DeepFace та ArcFace. Лінійна крива демонструє поступове зростання показників accuracy від VGGFace2 до ArcFace. Найнижчу точність забезпечує VGGFace2 (приблизно 92,8 %), тоді як FaceNet і DeepFace демонструють середні результати – 94,2 % і 93,7 % відповідно. Найвищий показник належить ArcFace, яка досягає 95,3 %, помітно перевершуючи всі інші моделі.

Графік підтверджує, що ArcFace забезпечує найкраще співвідношення між точністю та стабільністю розпізнавання. Значне відривання моделі від конкурентів свідчить про її ефективність у задачах біометричної ідентифікації,

особливо в системах, де критично важлива мінімізація помилкових прийняттів та відмов.

Зниження показників FAR і FRR свідчить про ефективність алгоритмів антиспуфінгу та оптимізації функції втрат, що дозволяє системі коректно відрізняти справжні обличчя від фотографій або відео. Важливим є і той факт, що час розпізнавання одного обличчя скоротився до 1,11 секунди, що робить технологію придатною для інтеграції у платіжні сервіси з високими вимогами до швидкості відповіді.

Окремо досліджувалася стійкість системи до зовнішніх факторів, зокрема змін освітлення, кута нахилу обличчя та міміки користувача. Результати наведено у таблиці 3.3.

Таблиця 3.3 – Вплив зовнішніх факторів на точність розпізнавання

Умови тестування	Точність (%)
Нормальне освітлення	95.8
Недостатнє освітлення	94.9
Надлишкове освітлення (блік)	95.3
Часткове перекриття обличчя	92.7
Зміна виразу обличчя / міміка	95.8
Нахил голови (до 20°)	93.5

Результати свідчать, що система зберігає високу точність у більшості умов, навіть при часткових перешкодах або зміні освітлення, що доводить її стабільність у реальних сценаріях використання.

Отримані результати експериментального дослідження створюють основу для практичного впровадження розробленої системи розпізнавання облич на базі ArcFace у реальні комерційні та технологічні середовища. Висока точність (95,3 %), низькі показники хибного прийняття та хибної відмови, а також швидкість обробки одного кадру до 1,11 секунди дозволяють інтегрувати систему в процеси, що вимагають миттєвої автентифікації та високої надійності.

Практична цінність роботи полягає у можливості використання моделі в різних сферах, де потрібні безконтактні рішення для ідентифікації користувачів: у платіжних терміналах, системах самообслуговування, безпілотних магазинах,

банкоматах нового покоління, транспортних валідаторах, а також у системах корпоративного доступу та контролю. Завдяки використанню нейронних мереж з високою стійкістю до зовнішніх факторів (змін освітлення, міміки, часткових перекриттів), система демонструє стабільність роботи у складних умовах експлуатації, що є важливою передумовою для її масового впровадження.

Особливе значення має використання хмарних сервісів AWS, зокрема Amazon DynamoDB для збереження біометричних шаблонів та Amazon S3 для архівування зображень і логів. Це створює умови для масштабування системи без втрати продуктивності, забезпечує високу відмовостійкість і відповідає міжнародним стандартам безпеки (AWS Security Best Practices). Використання таких інструментів дозволяє системі працювати з великими масивами даних та підтримувати одночасне обслуговування великої кількості користувачів.

До практичних результатів роботи належать також створені програмні модулі – детектор облич, модуль генерації ембедінгів, механізм порівняння векторів та підсистема управління транзакціями, які можуть бути використані як автономно, так і у складі комплексних інформаційних систем. Під час експериментів було визначено оптимальні параметри обробки даних, структуру векторних описів, а також сформовано рекомендації щодо налаштування камер відеоспостереження, що підвищує точність і швидкодію розпізнавання.

Практична ефективність розробленої системи підтверджена тестуванням на реальних даних: модель демонструє стабільні результати, мінімальні затримки між розпізнаванням та підтвердженням транзакції, а також здатність працювати в режимі реального часу без зниження якості. Це дозволяє рекомендувати її для подальшого промислового використання та інтеграції в існуючі інфраструктури фінансових та сервісних компаній.

У результаті проведеного експериментального дослідження підтверджено ефективність використання ArcFace для задач розпізнавання облич у системах безконтактних оплат. Отримані показники точності, швидкодії та надійності перевищують існуючі рішення, що свідчить про доцільність обраного підходу.

Система продемонструвала стабільність роботи за різних умов освітлення й пози користувача, що робить її придатною для практичного впровадження. Розроблена технологія може бути використана як основа для створення масштабованих, безпечних і зручних біометричних платіжних систем нового покоління.

У розділі було проведено детальне експериментальне дослідження запропонованої системи розпізнавання облич для безконтактних оплат, що охоплювало формування та попередню обробку даних, навчання моделі ArcFace, тестування її на різнорідних вибірках та оцінку за ключовими метриками. На основі аналізу публічних датасетів LFW, CelebA, наборів із платформи Kaggle та власної вибірки забезпечено високу репрезентативність навчальних даних, що дозволило моделі сформувати стійкі та дискримінативні ознаки облич. Проведена нормалізація, масштабування та аугментація підвищили узагальнювальні властивості моделі та зменшили вплив зовнішніх факторів на кінцевий результат.

ВИСНОВКИ

У магістерській роботі виконано комплексне дослідження, спрямоване на розробку та експериментальне підтвердження ефективності системи біометричної автентифікації на основі розпізнавання облич для здійснення безконтактних оплат. Проведений огляд наукових джерел і сучасних технологій показав, що методи глибокого навчання, зокрема архітектури ArcFace та MTCNN, нині займають провідні позиції у сфері комп'ютерного зору й широко застосовуються у фінансовій галузі, безпеці та інтелектуальних сервісах. Це підтверджує актуальність обраної теми й відповідність світовим тенденціям розвитку FinTech-рішень.

У межах першого етапу роботи було здійснено аналіз актуальних методів біометричної ідентифікації та встановлено ключові переваги використання технології Face-ID у фінансових операціях. Огляд наукової літератури та існуючих технологічних рішень виявив, що системи розпізнавання облич гарантують підвищений рівень захисту, істотно знижують імовірність несанкціонованого доступу та суттєво пришвидшують процес здійснення платежів, якщо порівнювати їх із традиційними методами на кшталт паролів, PIN-кодів чи використання фізичних карток. Це обґрунтувало вибір саме цього виду біометрії для реалізації безконтактних платежів.

На другому етапі досліджувалися сучасні конфігурації нейронних мереж, що використовуються для розв'язання задач ідентифікації облич, зокрема такі, як FaceNet, CosFace, ArcFace та інші модифікації згорткових нейронних мереж (CNN). Порівняння цих архітектур продемонструвало переваги ArcFace, обумовлені застосуванням кутової метрики, що дає змогу формувати високоселективні ембедінги навіть за умов змін освітлення, нахилу камери чи часткового закриття обличчя. Це дало змогу обґрунтовано обрати ArcFace як центральний елемент моделі верифікації.

У рамках третього завдання було спроектовано структуру та програмну реалізацію системи безконтактних розрахунків, базованої на Face-ID. Була

створена модульна система, яка інтегрує детекцію облич за допомогою MTCNN, генерацію ембедингів ArcFace, модулі попередньої обробки зображень, інтерфейс взаємодії з камерою та підсистему аутентифікації. Важливим аспектом стала інтеграція з хмарними сховищами AWS DynamoDB та S3, які забезпечують безпечне зберігання та швидке отримання біометричних даних.

На четвертому етапі було проведено комплексне тестування розробленої системи з оцінкою її точності, швидкодії та стійкості в умовах, наближених до реальних. Експериментальні дані підтвердили високу стабільність роботи модуля детекції MTCNN, коректність створення ембедингів ArcFace та достатній рівень успішної верифікації при варіаціях зовнішніх умов (освітлення, ракурси). Було визначено середній час, необхідний для обробки одного кадру, зафіксовано частоту помилкових відхилень, а також проаналізовано поведінку системи при роботі з різним обладнанням, що засвідчило її готовність до практичного застосування.

Під час виконання п’ятого завдання були визначені напрями подальшого розвитку системи та окреслено перспективи її комерціалізації у платіжних сервісах. Серед потенційних удосконалень виділено впровадження механізмів захисту від підміни (антиспуфінг) на основі даних про глибину, використання апаратних прискорювачів для підвищення продуктивності, розширення хмарної інфраструктури та адаптацію системи до використання на мобільних пристроях. Було також окреслено потенціал для масштабування цього рішення з метою його інтеграції у платіжні термінали, системи самообслуговування та сучасні цифрові платіжні екосистеми.

Наукова новизна роботи полягає в об’єднанні високоточних алгоритмів розпізнавання із платіжною логікою в єдиний програмний комплекс, оптимізований для режиму реального часу. Також удосконалено підхід до зберігання та захисту біометричних ознак шляхом використання хмарних технологій, що відповідає вимогам сучасної кібербезпеки. Створений прототип та експериментальні результати можуть бути основою для подальших наукових публікацій та практичних розробок.

Практична цінність роботи полягає у можливості її впровадження в торговельні мережі, банківські сервіси, транспортні системи та системи контролю доступу. Запропонована технологія здатна істотно зменшити час обслуговування, підвищити безпеку операцій та покращити користувацький досвід, замінюючи традиційні засоби автентифікації більш швидким і надійним біометричним механізмом.

Перспективи подальшого розвитку полягають у вдосконаленні антиспуфінгу, інтеграції 3D-моделей облич, застосуванні трансформерних архітектур, а також у розробленні мобільних реалізацій для периферійних пристроїв. Розширення можливостей системи може забезпечити її використання в умовах інтенсивного потоку користувачів та високих вимог до стійкості, швидкодії і захисту персональних даних.

Слід підкреслити, що основні результати цього магістерського дослідження отримали наукову апробацію: напрацювання були презентовані на міжнародній конференції Deep Tech Talent Initiative, де відзначено актуальність тематики та практичну цінність підходу до використання Face-ID у платіжних сервісах. Це свідчить про вагомість наукового внеску, зробленого в рамках роботи, та гарантує, що здобуті результати узгоджуються з актуальними напрямками розвитку біометричних систем.

У підсумку, виконана робота має не лише наукове, але й вагоме практичне значення, підтверджуючи ефективність сучасних методів комп’ютерного зору для вирішення завдань у фінансовій сфері та відкриваючи значні перспективи для подальших досліджень у сфері біометричних платіжних рішень.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Facial Recognition Technology: A Comprehensive Overview. Francis Academic Press. Francis Academic Press. URL: https://francis-press.com/papers/11241 (дата звернення: 13.08.2025).

2. Kortli Y. Face Recognition Systems: A Survey. URL: https://www.mdpi.com/1424-8220/20/2/342 (дата звернення: 15.08.2025).

3. Anugrah Pratama M. H. Advancing Secure Face Recognition Payment Systems: A Systematic Literature Review. URL: https://www.mdpi.com/2078-2489/16/7/581 (дата звернення: 18.08.2025).

4. R. Niroshan, N. Dani. Secured Payment System Using Face Recognition. TIJER Research Journal. ISSN : 2349-9249. URL: https://tijer.org/tijer/papers/TIJER 2504127.pdf (дата звернення: 20.08.2025).

5. Facial-Recognition Payment: An Example of Chinese Consumers. International Journal of Advanced Research in Science, Communication and Technology. URL: https://ijarsct.co.in/Paper4786.pdf (дата звернення: 27.08.2025).

6. Face Recognition Based on SVM and 2DPCA. arXiv.org. URL: https://arxiv.org/abs/1110.5404 (дата звернення: 27.08.2025).

7. Developer Guide Amazon DynamoDB. URL: https://docs.aws.amazon.com/ pdfs/amazondynamodb/latest/developerguide/dynamodb-dg.pdf (дата звернення: 05.09.2025).

8. ArcFace: Additive Angular Margin Loss for Deep Face Recognition. arXiv.org. URL: https://arxiv.org/abs/1801.07698 (дата звернення: 14.09.2025).

9. Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks. arXiv.org. URL: https://arxiv.org/abs/1604.02878 (дата звернення: 20.09.2025).

10. Module: tf TensorFlow v2.16.1. TensorFlow. URL: https://www.tensorflow. org/api_docs/python/tf (дата звернення: 28.09.2025).

11. JetBrains. PyCharm: The only Python IDE you need. JetBrains. URL: https://www.jetbrains. com/pycharm/ (дата звернення: 03.10.2025).

12. Microsoft. Tutorial: Get started with Visual Studio Code. Visual Studio Code – The open source AI code editor. URL: https://code.visualstudio.com/docs/getstarted/getting-started (дата звернення: 11.10.2025).

13. LibGuides: ScholarWorks: Data and Datasets. Home - LibGuides at University of Massachusetts Amherst. URL: https://guides.library.umass.edu/c.php?g=1240473&p=9117496 (дата звернення: 17.10.2025).

14. CelebA Dataset. Multimedia Laboratory. URL: https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html (дата звернення: 26.10.2025).

15. Face-Detection-Dataset. Fares Elmenshawii. Kaggle.com URL: https://www.kaggle.com/datasets/fareselmenshawii/face-detection-dataset (date of access: 06.11.2025).

ДОДАТКИ

Додаток А

Апробація результатів дослідження

eit Food

Co-funded by the European Union

СЕРТИФІКАТ

засвідчує, що

Денисюк Андрій

успішно завершив(ла) навчання в межах Програми EIT Deep Tech Talent Initiative в Україні. Програма поєднувала базові концепції штучного інтелекту, програмування, підприємництва та управління проєктами для розвитку інноваційних ідей.

Сертифікат засвідчує розвиток таких компетенцій: робота з даними, основи штучного інтелекту, розробка інноваційних рішень у сфері агропродовольства, командна робота, критичне мислення, підприємницьке мислення та навички проєктного менеджменту.

2024-2025 навчальний рік

Голова ГО "Джуніор Ачівмент Україна"
Юрій Токарський

intel

PSNC

JA Europe

JA Ukraine

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European

Додаток Б

Клас розпізнавання та опрацювання обличчя

```
class FaceRecognitionSystem:
    def __init__(self):
        print("Ініціалізація системи розпізнавання обличь...")
        self.detector = MTCNN()
        self.model = self._build_model()
        self.face_database = {}
        print("Система успішно ініціалізована!\n")

    def _build_model(self):
        """Створення моделі для виділення ознак обличчя"""
        print("Будування моделі ArcFace...")
        base = MobileNetV2(input_shape=(112, 112, 3),
                           include_top=False,
                           weights='imagenet')
        x = GlobalAveragePooling2D()(base.output)
        embeddings = Dense(256)(x)
        model = Model(inputs=base.input, outputs=embeddings)
        print("Модель успішно створена!")
        return model

    def preprocess_face(self, face_img):
        """Попередня обробка зображення обличчя"""
        face_img = cv2.resize(face_img, (112, 112))
        face_img = cv2.cvtColor(face_img, cv2.COLOR_BGR2RGB)
        return (face_img / 255.0).astype(np.float32)

    def register_face(self, image, user_id):
        """Реєстрація нового обличчя в системі"""
        print(f"\nСпроба реєстрації користувача {user_id}...")
        faces = self.detector.detect_faces(image)

        if not faces:
            print("Помилка: обличчя не знайдено!")
            return False

        x, y, w, h = faces[0]['box']
        face = image[y:y + h, x:x + w]
        face = self.preprocess_face(face)

        print("Генерація ембеддінгу обличчя...")
        embedding = self.model.predict(np.expand_dims(face, axis=0))[0]
        self.face_database[user_id] = embedding
        print(f"Користувач {user_id} успішно зареєстрований!")
        return True

    def verify_face(self, image, threshold=0.7):
        """Верифікація обличчя"""
        print("\nПочаток процесу верифікації...")
```

```
        faces = self.detector.detect_faces(image)

        if not faces:
            print("Помилка: обличчя не знайдено!")
            return False, None

        x, y, w, h = faces[0]['box']
        face = image[y:y + h, x:x + w]
        face = self.preprocess_face(face)

        print("Генерація ембеддінгу для верифікації...")
        embedding = self.model.predict(np.expand_dims(face, axis=0))[0]

        if not self.face_database:
            print("Увага: база даних обличь порожня!")
            return False, None

        print("Порівняння з базою даних...")
        best_match = None
        best_score = -1

        for user_id, db_emb in self.face_database.items():
            score = cosine_similarity([embedding], [db_emb])[0][0]
            if score > best_score:
                best_score = score
                best_match = user_id

        print(f"Найкращий результат: {best_score:.4f} (поріг:
{threshold})")

        if best_score >= threshold:
            print(f"Верифікація успішна! Користувач: {best_match}")
            return True, best_match
        else:
            print("Верифікація не пройдена!")
            return False, None
```

Додаток В

Клас безпечного зберігання біометричних даних

```
    class SecureBiometricStorage:
        """
        Модуль безпечного зберігання біометричних векторів.
        """

        def __init__(self, admin_password: str):
            self.admin_password_hash =
self._hash_password(admin_password)

            # Генерація або завантаження ключа шифрування
            self.key = self._load_or_create_key()
            self.cipher = Fernet(self.key)

            # Місце зберігання локальних зашифрованих векторів
            self.storage_file = "encrypted_face_vectors.json"

            if not os.path.exists(self.storage_file):
                with open(self.storage_file, "w") as f:
                    json.dump({}, f)

        #  Автентифікація

        def _hash_password(self, password: str) -> str:
            """Проста хеш-функція (можна замінити на bcrypt)."""
            import hashlib
            return hashlib.sha256(password.encode()).hexdigest()

        def authenticate(self, entered_password: str) -> bool:
            """Перевірка пароля адміністратора."""
            return self._hash_password(entered_password) ==
self.admin_password_hash

        #  Керування ключем

        def _load_or_create_key(self):
            """Завантаження або створення ключа шифрування."""
            key_path = "encryption.key"
            if os.path.exists(key_path):
                with open(key_path, "rb") as f:
                    return f.read()
            else:
                key = Fernet.generate_key()
                with open(key_path, "wb") as f:
                    f.write(key)
                return key
```

```
        def encrypt_vector(self, vector):
            """Шифрування біометричного вектора."""
            data = json.dumps(vector.tolist()).encode()
            encrypted = self.cipher.encrypt(data)
            return encrypted.decode()

        def decrypt_vector(self, encrypted_vector):
            """Розшифрування біометричного вектора."""
            decrypted = self.cipher.decrypt(encrypted_vector.encode())
            return json.loads(decrypted)

        def save_vector(self, user_id: str, vector, admin_password: str):
            """Збереження зашифрованого вектора у локальне сховище."""
            if not self.authenticate(admin_password):
                raise PermissionError("Помилка доступу: невірний пароль
адміністратора.")

            encrypted = self.encrypt_vector(vector)

            with open(self.storage_file, "r") as f:
                db = json.load(f)

            db[user_id] = encrypted

            with open(self.storage_file, "w") as f:
                json.dump(db, f, indent=4)

            print(f"Вектор для користувача {user_id} успішно зашифровано
та збережено.")
        def load_vector(self, user_id: str, admin_password: str):
            """Завантаження та розшифрування вектора."""
            if not self.authenticate(admin_password):
                raise PermissionError("Помилка доступу: невірний пароль
адміністратора.")

            with open(self.storage_file, "r") as f:
                db = json.load(f)

            if user_id not in db:
                raise KeyError("Користувача не знайдено у сховищі.")
            decrypted = self.decrypt_vector(db[user_id])
            return decrypted
```