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

КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

Галузь знань 12 – Інформаційні технології

Спеціальність 123 –Комп'ютерна інженерія

на тему «Метод та програмно-технічний засіб для системи пропуску на основі біометричних даних»

КвРКІП. 2303214.23.03.59 ПЗ

Виконав: студент 2 курсу, група КІ2м-23-3 Павло ЮРКО
Підпис Ім'я, прізвище

Керівник зав. кафедри КІІС. доцент, Ольга ПАВЛОВА
Науковий ступінь, вчене звання Підпис Ім'я, прізвище

До захисту допускаю:
Зав. кафедри КІІС, доктор філософії, доцент
Ольга ПАВЛОВА
09 04 2025 р.

Хмельницький, 2025

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет Інформаційних технологій

Кафедра Комп'ютерної інженерії та інформаційних систем

Освітній рівень магістр

Галузь знань 12 Інформаційні технології

Спеціальність 123 Комп'ютерна інженерія

Освітня програма освітньо-наукова програма «Комп'ютерна інженерія та програмування»

ЗАТВЕРДЖУЮ

Зав. кафедри Ольга ПАВЛОВА

"01" 09 2024 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА

Павла ЮРКА

Прізвище, ім'я, по батькові студента

1. Тема проекту (роботи) Метод та програмно-технічний засіб для системи пропуску на основі біометричних даних

Керівник проекту (роботи) Ольга ПАВЛОВА, зав. кафедри КІІС, доцент

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

Затверджена наказом ректора університету від 08.01.2025 №8

2. Строк подання студентом проекту (роботи) на кафедру 01.05.2025 р.

3. Вихідні дані до проекту (роботи) Завдання на дипломне проектування

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

Аналіз існуючих рішень для систем пропуску

Застосування технології розпізнавання зображень для систем пропуску на основі біометричних даних

Метод та алгоритм використання біометричних даних для систем пропуску

Результати роботи програмно-технічного засобу для для системи пропуску на основі біометричних даних

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

6. Консультанти розділів кваліфікаційної роботи магістра

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийн
Нормоконтроль	Сергій ЛИСЕНКО, професор кафедри КІІС		
Антиплагіат	Андрій НІЧЕПОРУК, доцент кафедри КІІС		

7. Дата видачі завдання « 01 » 09 2024р.

КАЛЕНДАРНИЙ ПЛАН

№з/п	Назва етапів (розділів) кваліфікаційної роботи магістра	Термін виконання етапів проекту (роботи)	Приміт
1	Вибір напряму дослідження та узгодження тематики КвРМ з керівником	01.09.2024	виконан
2	Ознайомлення з предметною областю; формулювання мети та задач дослідження; визначення об'єкта та предмета дослідження	01.10.2024	виконан
3	Робота над розділом 1 – аналіз відомих моделей, методів за темою; постановка задачі	01.11.2024	виконан
4	Робота над розділом 2 – розробка моделей для вирішення поставленої задачі	01.12.2024	виконан
5	Робота над науковою статтею	01.02.2025	виконан
6	Робота над розділом 3 – розробка методів для вирішення поставленої задачі	15.02.2025	виконан
7	Робота над розділом 4 – проектування та розробка ПЗ для вирішення поставленої задачі, експериментальна частина	01.04.2025	виконан
8	Оформлення пояснювальної записки згідно вимог	18.04.2025	виконан
9	Попередній захист ДРМ	29.04.2025	виконан
10	Захист ДРМ на засіданні ЕК	До 15.05.2025	

Студент _____ Павло ЮРКО
Підпис Ім'я, прізвище

Керівник роботи _____ Ольга ПАВЛОВА
Підпис Ім'я, прізвище

РЕФЕРАТ

Тема кваліфікаційної роботи магістра: Метод та програмно-технічний засіб для системи пропуску на основі біометричних даних

Автор роботи: Павло ЮРКО

Керівник роботи: Ольга ПАВЛОВА

Пояснювальна записка: 102 с., 34 рис., 9 табл., 3 дод., 82 джерел.

БІОМЕТРИЧНА ІДЕНТИФІКАЦІЯ, РОЗПІЗНАВАННЯ ОБЛИЧ, КОНТРОЛЬ ДОСТУПУ, ШТУЧНИЙ ІНТЕЛЕКТ, МАШИННЕ НАВЧАННЯ, СИСТЕМА ПРОПУСКУ.

Об’єктом дослідження є системи пропуску на основі біометричних даних

Предметом дослідження є застосування розпізнавання зображень для системи пропуску на основі біометричних даних

Метою кваліфікаційної роботи магістра є розробити метод та програмно-технічний засіб для системи пропуску на основі біометричних даних

Методи дослідження: аналіз існуючих рішень у сфері біометричної ідентифікації, застосування алгоритму максимальної ентропії для навчання моделі розпізнавання обличь, використання метрик оцінки точності.

Результати роботи рекомендується використовувати для підвищення безпеки у корпоративних, державних та приватних об'єктах шляхом впровадження біометричних систем контролю доступу. Запропонований метод може бути інтегрований у існуючі системи безпеки для посилення їхньої ефективності та адаптації до сценаріїв реального часу.

Очікувані наукові результати:

розробка методу застосування розпізнавання зображень для системи пропуску на основі біометричних даних

Очікувана практична цінність: розроблений програмно-технічний засіб для системи пропуску на основі біометричних даних.

ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

МСП - механічні системи пропуску

СП – системи пропуску

БСП – біометричні системи пропуску

ЕСП - електроні системи пропуску

БС - біометричні системи

ВСТУП

В умовах стрімкого розвитку інформаційних технологій та зростання загроз кібербезпеці особливої уваги потребують засоби ідентифікації особи для забезпечення контролю доступу до об'єктів, що мають підвищені вимоги до безпеки. Традиційні методи аутентифікації, такі як паролі, PIN-коди або картки доступу, дедалі частіше стають недостатньо надійними через ризики компрометації, втрати або несанкціонованого використання [1]. Водночас біометричні технології, які базуються на унікальних фізіологічних або поведінкових характеристиках людини, забезпечують значно вищий рівень ідентифікації, оскільки вони важко підробляються або передаються третім особам.

Сучасні системи біометричної ідентифікації включають розпізнавання відбитків пальців, райдужної оболонки ока, обличчя, голосу та навіть манери друку на клавіатурі. Серед цих методів особливої популярності набуло розпізнавання обличчя, оскільки воно поєднує високу зручність використання з достатньо високим рівнем точності. Доступність технологій комп’ютерного зору та машинного навчання дозволяє автоматизувати процес обробки зображень та підвищити швидкість ідентифікації. Впровадження таких систем є особливо актуальним для об'єктів критичної інфраструктури, банківських установ, підприємств, навчальних закладів та інших організацій, де необхідно контролювати доступ осіб до визначених зон.

Проблема надійності та точності розпізнавання обличчя залишається одним із ключових викликів у розвитку подібних систем. Недостатня якість вхідних зображень, змінні умови освітлення, часткове закриття обличчя або спроби обману системи за допомогою фотографій або 3D-масок можуть призводити до зниження точності ідентифікації [2]. Окрім того, важливою проблемою є швидкість обробки даних, особливо у випадках, коли система повинна працювати в режимі реального часу [3]. Саме тому дослідження, спрямовані на вдосконалення методів біометричного розпізнавання та підвищення ефективності їх реалізації, є вкрай важливими для забезпечення безпеки доступу.

Існуючі рішення у сфері біометричних систем доступу можна розділити на апаратні та програмні. Апаратні рішення, що включають вбудовані сенсори та спеціалізовані процесори, демонструють високу продуктивність, проте вони часто є дорогими у виробництві та мають обмежену гнучкість для модифікації алгоритмів [4]. Програмні рішення, зокрема ті, що базуються на використанні бібліотек глибокого навчання та нейронних мереж, забезпечують більшу адаптивність, дозволяючи системам навчатися новим шаблонам та вдосконалюватися в процесі експлуатації. Сучасні бібліотеки, такі як OpenCV, TensorFlow, Dlib та ML.NET, надають розширені можливості для розробки ефективних систем розпізнавання облич з мінімальними вимогами до обчислювальних ресурсів [5].

У світовій практиці широко використовують БС контролю доступу, що інтегрують з іншими засобами безпеки, такими як RFID-картки або PIN-коди, що дозволяє підвищити рівень надійності за рахунок багатофакторної аутентифікації. В Україні застосування біометричних систем є поки що менш поширеним через фінансові обмеження та недостатню обізнаність підприємств щодо їх переваг. Водночас, із зростанням потреби у захисті інформаційних систем та фізичних об'єктів, інтеграція подібних рішень набуває стратегічного значення.

Однією з вагомих переваг біометричних систем є можливість централізованого контролю доступу із застосуванням хмарних технологій та автоматизованих баз даних. Це дозволяє не лише відстежувати доступ осіб у режимі реального часу, а й оперативно змінювати політики безпеки в залежності від рівня загрози. Дослідження у цьому напрямку сприяє створенню інтелектуальних систем, що можуть самостійно адаптуватися до змін навколишнього середовища та умов експлуатації.

Отже, розробка нових методів і програмно-технічних засобів біометричної ідентифікації є важливим кроком на шляху до вдосконалення систем контролю доступу. Поєднання сучасних технологій машинного навчання, алгоритмів обробки зображень та інтеграції з інформаційними системами безпеки дозволить

створити гнучку, надійну та масштабовану систему пропуску, здатну відповідати сучасним викликам у сфері кібербезпеки та фізичного контролю доступу.

Об’єктом дослідження є процеси біометричної ідентифікації особи для автоматизованих систем контролю доступу.

Предметом дослідження є методи та алгоритми обробки біометричних зображень для розпізнавання обличчя, а також програмно-технічні засоби реалізації автоматизованої СП, що включають моделі машинного навчання, алгоритми навчання нейронних мереж та способи підвищення точності ідентифікації.

Метою даної роботи є розробка методу та програмно-технічного засобу для СП на основі біометричних даних, що забезпечує високу точність, швидкість обробки та мінімальні вимоги до апаратних ресурсів.

Для досягнення цієї мети необхідно вирішити такі основні задачі:

- провести аналіз існуючих рішень для систем пропуску, розглянувши їх класифікацію, переваги та недоліки, а також оцінити використання біометричних технологій у світовій та українській практиці;
- розробити та адаптувати метод машинного навчання для розпізнавання обличчя, включаючи вибір відповідної архітектури нейронної мережі, алгоритму обробки похибок та методів підвищення точності моделі;
- розробити алгоритм функціонування СП, що включає етапи збирання, попередньої обробки та класифікації біометричних даних, а також забезпечує високу швидкість роботи та захист від можливих атак або підробок;
- реалізувати та експериментально оцінити програмно-технічний засіб, здійснивши тестування роботи системи, аналізуючи її точність, швидкодію та ефективність у реальних сценаріях використання.

Для досягнення поставленої мети та вирішення основних задач дослідження було використано наступні методи:

- аналіз та узагальнення наукових і технічних джерел. Застосовувався для вивчення існуючих методів біометричної ідентифікації, систем контролю доступу та технологій машинного навчання, що використовуються у подібних розробках;

− методи машинного навчання. Застосовувалися для побудови моделі розпізнавання облич на основі нейронних мереж, вибору архітектури, оптимізації параметрів та вдосконалення точності класифікації;

− математичне моделювання. Використовувалося для розробки математичної моделі процесу розпізнавання обличчя, оцінки ймовірностей похибок та підвищення стійкості системи до зовнішніх факторів, таких як освітлення чи часткове закриття обличчя.

− експериментальні методи. Застосовувалися для тестування програмно-технічного засобу, оцінки його продуктивності, точності та швидкості розпізнавання.

У процесі виконання роботи розроблено програмно-технічний засіб для СП на основі біометричних даних, який використовує машинне навчання для розпізнавання облич. Відмінною особливістю є застосування алгоритму максимальної ентропії на основі обмеженого спуску за градієнтом для багатокласової класифікації, що дозволило підвищити точність ідентифікації. Також було розроблено механізм попередньої обробки зображень, який включає нормалізацію та перетворення колірних характеристик для покращення якості навчання моделі. Проведене тестування системи підтвердило її здатність забезпечувати швидке та точне розпізнавання осіб, що робить її перспективною для застосування в системах контролю доступу.

Результати дослідження мають практичне значення для розробки та впровадження автоматизованих систем контролю доступу, що використовують біометричні дані для ідентифікації осіб. Запропонований програмно-технічний засіб може бути інтегрований у системи безпеки підприємств, навчальних закладів, державних установ, банків та інших об'єктів з підвищеними вимогами до контролю пропускного режиму.

Розроблена модель машинного навчання забезпечує високу точність розпізнавання осіб, навіть за умов змінного освітлення та часткового перекриття обличчя. Використання механізмів попередньої обробки зображень дозволяє

зменшити вплив шумів та покращити якість ідентифікації, що є важливим для реальних умов експлуатації.

Запропоновані алгоритми та реалізоване програмне забезпечення можуть бути використані для підвищення рівня безпеки на об'єктах, де важливо виключити можливість несанкціонованого доступу. Також результати дослідження можуть бути адаптовані для подальшої інтеграції із системами відеоспостереження, платіжними терміналами, мобільними застосунками та іншими технологічними рішеннями у сфері біометричної ідентифікації.

1 АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ДЛЯ СИСТЕМ ПРОПУСКУ

1.1 Види систем пропуску

СП відіграють ключову роль у забезпеченні контролю доступу на об'єктах з обмеженим входом, що можуть включати державні установи, підприємства, навчальні заклади, банки, медичні заклади та інші організації. Основне завдання таких систем – забезпечити ідентифікацію та автентифікацію осіб, що намагаються отримати доступ до певної території чи інформаційних ресурсів. Всі СП можна поділити на механічні, електронні та біометричні.

МСП є найпростішим способом обмеження доступу до об'єктів, що не потребують високого рівня автоматизації. Вони базуються на фізичних пристроях, які дозволяють або блокують прохід, забезпечуючи захист приміщень, територій та інших зон з обмеженим доступом [8]. Основні їхні переваги – надійність, простота використання та незалежність від електроживлення, що робить такі системи ефективними для застосування в місцях без постійного доступу до електромереж [9].

МСП можна поділити на ключі і замки та турнікети й ворота, кожна з яких має свої особливості та сфери застосування. Ці системи часто поєднуються з іншими методами контролю доступу, такими як електронні чи БС, для підвищення рівня безпеки.

На рис. 1.1 наведено основні особливості механічних систем пропуску, що визначають їхню функціональність та варіанти застосування.

Ключі та замки є найбільш розповсюдженим видом механічного контролю доступу, що забезпечує фізичне блокування входу. Вони широко використовуються у житлових, офісних та промислових приміщеннях, де важливо розмежувати рівні доступу та забезпечити контрольований вхід. Основна перевага таких систем – відсутність потреби в електроживленні, що робить їх надійним рішенням у будь-яких умовах експлуатації.

Рисунок 1.1 – Особливості механічних систем пропуску

Турнікети та ворота забезпечують фізичне обмеження доступу через прохід, що є важливим для громадських місць, підприємств та об'єктів з високою прохідністю [10]. Вони можуть бути односторонніми або двосторонніми, що дозволяє гнучко налаштовувати їхню роботу відповідно до вимог об’єкта..

МСП є ефективним способом контролю доступу завдяки простоті, надійності та незалежності від електроживлення. Вони широко використовуються у місцях, де необхідно фізично обмежити прохід, але не завжди потребується автоматизована перевірка особи.

ЕСП є сучасним способом контролю доступу, що базується на цифрових технологіях для ідентифікації користувачів. Вони використовуються у бізнес-центрах, навчальних закладах, готелях, підприємствах та інших об'єктах, де необхідно забезпечити швидкий, автоматизований та безконтактний контроль входу [11]. На відміну від механічних систем, електронні дозволяють реєструвати кожен вхід і вихід, а також централізовано керувати рівнями доступу для різних категорій осіб [12]. Основні переваги таких систем – зручність використання,

гнучкість у налаштуванні прав доступу та можливість інтеграції з іншими цифровими рішеннями.

Оскільки електронні системи можуть використовувати різні способи ідентифікації – картки доступу, PIN-коди, мобільні ідентифікатори – їхня функціональність є досить широкою. Важливим аспектом таких рішень є поєднання програмного та апаратного забезпечення, що дає змогу реалізовувати гнучку систему контролю, адаптовану до потреб конкретного об'єкта.

На рис. 1.2 наведені три основні категорії електронних систем пропуску: карткові системи доступу, кодові панелі та PIN-коди, а також мобільні ідентифікатори, які відрізняються за технологіями, рівнем персоналізації та способом взаємодії з користувачем.

Рисунок 1.2 – Особливості електронних систем пропуску

Карткові системи доступу – використовують RFID, NFC або магнітні картки, що містять унікальний ідентифікатор користувача. Вони широко застосовуються в офісах та готелях, де потрібно швидко перевіряти право доступу [13]. Основний

принцип роботи полягає у зчитуванні та перевірці коду картки з базою даних системи.

Кодові панелі та PIN-коди – забезпечують доступ через введення персонального пароля. Ця технологія не потребує фізичного носія, адже користувач повинен лише пам’ятати свій унікальний код [14].

ЕСП є гнучким, безконтактним і надійним способом контролю доступу, що поєднує автоматизацію процесів та персоналізовану ідентифікацію. Вони дозволяють уникнути недоліків механічних систем, таких як втрата ключів або дублювання доступу, і забезпечують можливість централізованого керування користувачами.

БСП є найбільш сучасним та безпечним способом контролю доступу, який базується на використанні унікальних фізіологічних та поведінкових характеристик людини. Вони застосовуються у сферах, де необхідно забезпечити високий рівень захисту та виключити можливість передачі ключів, карток або паролів стороннім особам [15]. Основною перевагою біометричних технологій є їхня здатність ідентифікувати саме людину, а не її носій доступу, що значно підвищує рівень безпеки [16].

Основні технології біометричної ідентифікації включають розпізнавання відбитків пальців, обличчя, райдужної оболонки ока, голосу та навіть манери друку на клавіатурі. Системи розпізнавання обличчя є одними з найбільш зручних, оскільки вони не потребують фізичного контакту та можуть працювати в режимі реального часу, що особливо важливо для організацій із великим потоком користувачів [17]. БС активно впроваджуються в аеропортах, банках, підприємствах, державних установах та навіть у мобільних пристроях, що підтверджує їхню високу ефективність та зручність у використанні.

Функціональні особливості біометричних систем детально проілюстровані на рис. 1.3, де наведено їх ключові характеристики та принципи застосування в системах контролю доступу.

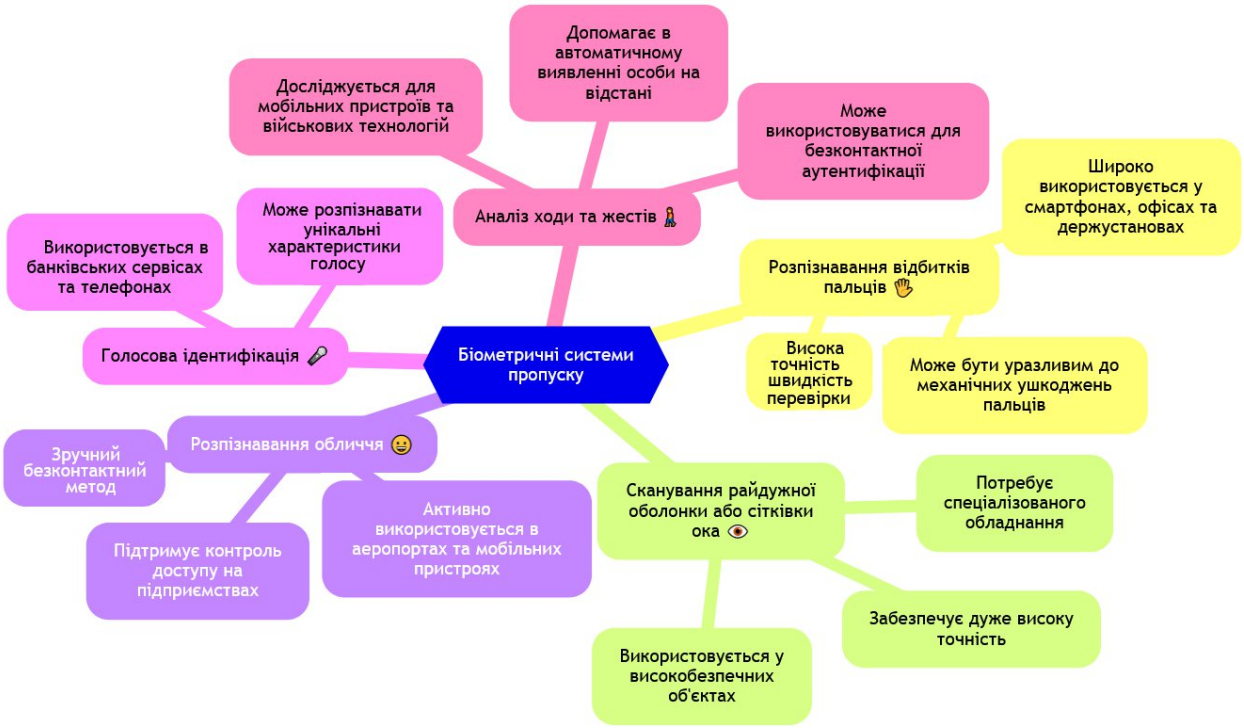


Рисунок 1.3 – Особливості біометричних систем пропуску

До основних методів біометричної ідентифікації у сфері контролю доступу належать [18]:

– розпізнавання відбитків пальців. Його перевага – це висока точність, але недоліком може бути чутливість до пошкоджень шкіри або забруднень;

– розпізнавання обличчя. Системи можуть працювати навіть у складних умовах освітлення, що робить їх зручними для громадських місць та підприємств;

– ідентифікація за райдужною оболонкою ока це один із найточніших методів біометрії, який використовується у військових та урядових структурах. Недоліком є потреба у спеціальному обладнанні для сканування;

– голосова ідентифікація це метод, що аналізує унікальні параметри голосу користувача. Часто застосовується в телефонних службах підтримки та фінансових установах. Основним викликом є залежність від навколишнього шуму та змін у голосі.

– динамічний аналіз поведінкових характеристик включає розпізнавання ходи, стилю друку на клавіатурі або жестів. Використовується як додатковий рівень безпеки, наприклад, у банківських системах.

БСП є найперспективнішим напрямком розвитку контролю доступу, оскільки забезпечують високу точність, персоналізовану ідентифікацію та захист від несанкціонованого проникнення.

Для оцінки ефективності різних видів систем пропуску доцільно порівняти їх за основними характеристиками, такими як надійність, зручність використання, вартість впровадження, рівень ризиків компрометації, точність ідентифікації, кількість спроб для доступу та швидкість роботи. Кожен тип системи має свої переваги та обмеження, які впливають на вибір рішення для конкретного об’єкта чи сфери застосування.

Таблиця 1.1 – Характеристики біометричних систем пропуску

Характеристика	Система		
	Механічна	Електронна	Біометрична
Рівень надійності	Низький	Середній	Високий
Зручність використання	Низький	Високий	Високий
Вартість впровадження	Низький	Середній	Високий
Ризики компрометації	Високий (ключі можна втратити, передати чи підробити)	Середній (можлива крадіжка або передача картки/коду)	Низький (за умови правильної реалізації)
Точність ідентифікації (%)	100	95	99.5
Кількість спроб для доступу	Необмежено	3	1
Середній час ідентифікації (сек)	5	2	0.5

Аналізуючи дані таблиці, можна зробити кілька ключових висновків. БС забезпечують найвищий рівень безпеки завдяки персоналізованій ідентифікації, що унеможливлює несанкціонований доступ через передачу носія або підробку ідентифікатора. Це робить їх ефективним рішенням для об’єктів, де контроль доступу є критично важливим.

Механічні системи залишаються доступним та простим варіантом, однак вони мають обмежену гнучкість та підвищені ризики компрометації. Використання таких рішень доцільне як базовий рівень захисту, проте для підвищення ефективності контролю їх варто доповнювати електронними або біометричними технологіями.

Електронні системи поєднують у собі баланс між безпекою, зручністю та вартістю, що робить їх оптимальними для організацій, де важливий швидкий і контрольований доступ. Водночас можливість крадіжки або передачі карток та кодів потребує додаткових заходів безпеки, серед яких двофакторна автентифікація або обмеження кількості спроб входу.

1.2 Аналіз технологій для систем пропуску в Україні та світі

Системи контролю доступу активно розвиваються як в Україні, так і в світі, забезпечуючи безпеку на підприємствах, державних установах, громадських місцях та житлових комплексах. Використання сучасних технологій дозволяє автоматизувати процес ідентифікації особи, мінімізувати ризики несанкціонованого доступу та покращити управління доступом до об'єктів.

БСП активно застосовуються у США, країнах ЄС, Японії та Південній Кореї, забезпечуючи високий рівень безпеки та автоматизації процесів ідентифікації. В аеропортах, таких як Heathrow (Велика Британія) та John F. Kennedy International Airport (США), впроваджені системи розпізнавання обличчя для прискореного проходу прикордонного контролю, що дозволяє мінімізувати час перевірки документів та знизити ризик шахрайства [19].

У фінансовому секторі, наприклад, в Японії низка банків, включаючи MUFG Bank, активно використовують ідентифікацію за відбитками пальців для доступу до банківських рахунків без використання карток чи паролів [20].

Китай є одним із лідерів у впровадженні біометричних технологій у повсякденне життя. У Пекіні та Шанхаї пасажири метро можуть використовувати розпізнавання обличчя замість квитків, а в системі Alipay FacePay клієнти здійснюють покупки в магазинах лише за допомогою сканування обличчя без потреби у смартфоні чи банківській картці [21].

RFID-картки та безконтактні ключі залишаються одним із найпоширеніших рішень у сфері контролю доступу, особливо в корпоративному сегменті, готелях та промислових підприємствах [22].

У Європі активно розвиваються системи доступу на основі NFC (Near Field Communication), що дають змогу використовувати смартфони як електронні ключі без потреби у фізичних картках [23]. Наприклад, технологія Apple Wallet Hotel Key уже інтегрована в системи контролю доступу таких готельних мереж, як Hyatt, дозволяючи гостям відкривати двері номерів через смартфон [24].

Штучний інтелект у системах контролю доступу застосовується для прогнозування потенційних загроз, аналізу поведінки користувачів та автоматичного виявлення підозрілих дій [25]. У великих комерційних та державних установах впроваджуються розумні системи безпеки, що використовують відеоаналітику та алгоритми машинного навчання для підвищення ефективності контролю доступу.

Наприклад, система Avigilon Access Control від Motorola Solutions інтегрується з технологією Appearance Search AI, що дозволяє не лише ідентифікувати особу за допомогою біометричних параметрів, а й відстежувати її переміщення по об’єкту в режимі реального часу [26].

У корпоративному секторі України найбільш поширеними є RFID-системи, безконтактні картки та кодові панелі, які використовуються для контролю доступу до офісних приміщень, виробничих зон та стратегічних об'єктів.

Такі рішення дозволяють ефективно керувати рівнями доступу співробітників і забезпечують зручний моніторинг відвідуваності. Наприклад, система UNIPASS широко застосовується у великих компаніях для контролю доступу на основі карток і мобільних ідентифікаторів [27].

Деякі великі підприємства в Україні впроваджують БС доступу, що працюють на основі ідентифікації за відбитками пальців або розпізнавання обличчя. Наприклад, у бізнес-центрах Києва, таких як UNIT.City, використовується Ajax KeyPad Plus, що підтримує безконтактну ідентифікацію через картки та мобільні додатки [28]. Водночас компанії на кшталт Нова Пошта впроваджують розпізнавання обличчя для входу в логістичні центри, що дозволяє прискорити пропускний процес та знизити ризики несанкціонованого доступу.

У житловому секторі України все частіше впроваджуються сучасні домофонні системи з біометричним розпізнаванням, що підвищують рівень безпеки та зручності для мешканців багатоквартирних будинків. Наприклад, багатофункціональні домофонні системи BAS-IP підтримують Face ID, що дозволяє автоматично відкривати двері за допомогою біометрії без потреби у фізичних ключах чи кодах [29].

Окрім того, зростає популярність мобільних додатків для дистанційного відкриття дверей, що дозволяють керувати доступом до під'їздів та прибудинкових територій безпосередньо зі смартфона. Одним із таких рішень є Slinex Smart Call, який інтегрується зі смартфонами мешканців і надає можливість переглядати відео з домофона та відкривати двері дистанційно [30]. Також Hikvision Villa Door Station, яка активно використовується у сучасних житлових комплексах Києва та інших міст України, підтримує керування доступом через мобільний застосунок, що спрощує вхід для мешканців та їхніх гостей.

У державних установах України поступово впроваджуються системи розпізнавання обличчя та багатофакторна ідентифікація, що сприяє підвищенню рівня безпеки доступу до конфіденційної інформації та стратегічних об'єктів [31].

Одним із таких рішень є система Identigraf, розроблена для аналізу та верифікації осіб за допомогою технологій штучного інтелекту, яка вже

використовується правоохоронними органами та силовими структурами для контролю доступу та виявлення потенційних загроз.

Також у Міністерстві цифрової трансформації України активно розвивається Єдина біометрична система, що інтегрується з базами даних державних органів і дозволяє здійснювати автоматизовану ідентифікацію осіб за відбитками пальців та обличчям.

Крім того, на об'єктах критичної інфраструктури, таких як Центр обробки даних Державної податкової служби України, впроваджено Hikvision Face Recognition Terminals, які дозволяють здійснювати контроль доступу працівників та відвідувачів на основі біометричних параметрів [32].

У транспортній інфраструктурі України активно впроваджуються електронні квитки, мобільні ідентифікатори та безконтактні платіжні системи, що значно покращують швидкість та зручність проходу через контрольні пункти.

У Києві діє система «Kyiv Smart Card», яка дозволяє пасажирам оплачувати проїзд у громадському транспорті за допомогою безконтактних карток, мобільних додатків та навіть банківських карт із підтримкою NFC [33]. Аналогічні рішення реалізовані у Львові та Харкові, де працюють електронні квитки «ЛеоКарт» та «Е-квиток» відповідно, що забезпечують можливість безготівкової оплати проїзду та централізованого контролю пасажиропотоку [34].

Окрім того, в українських аеропортах впроваджуються автоматизовані системи контролю доступу на основі біометричних технологій. Наприклад, у Міжнародному аеропорту "Бориспіль" використовується система FacePay24 від ПриватБанку, що дозволяє пасажирам проходити реєстрацію та контроль безконтактно, лише за допомогою сканування обличчя [35].

Для оцінки рівня впровадження сучасних систем контролю доступу в Україні та світі виділено ключові аспекти, які наведені у табл. 1.2.

Таблиця 1.2 – Характеристики біометричних систем пропуску

Характеристика	Україна	Світові тенденції	Характеристика
Популярність біометричних систем	Обмежене впровадження у державних та корпоративних структурах	Широке застосування у державному секторі, бізнесі та транспорті	Популярність біометричних систем
Рівень використання RFID-карт	Високий у комерційному секторі	Частково замінюється мобільними ідентифікаторами та NFC-рішеннями	Рівень використання RFID-карт
Мобільні технології доступу	Розвиваються, але впроваджені обмежено	Активне застосування в поєднанні з біометричними методами	Мобільні технології доступу
Інтеграція ШІ у системи безпеки	Поки що на початковому рівні	Використовується для аналітики поведінки та прогнозування загроз	Інтеграція ШІ у системи безпеки
Рівень законодавчого регулювання	Розвиток нормативних актів у сфері захисту персональних даних	Високий рівень регулювання та вимог до безпеки інформації	Рівень законодавчого регулювання

Як видно з аналізу, Україна поступово адаптує світові технології контролю доступу, однак їх впровадження поки що відбувається нерівномірно. Основні тенденції розвитку включають перехід до цифрових та біометричних рішень, але

процес упровадження потребує додаткової модернізації та гармонізації з міжнародними стандартами.

В Україні впровадження цих технологій відбувається поступово, зосереджуючись на корпоративному та державному секторі. Незважаючи на наявні бар'єри, зростаючий попит на БС та мобільні ідентифікатори свідчить про активну інтеграцію сучасних технологій у сферу контролю доступу.

1.3 Постановка задачі та вибір технологій для реалізації

Розробка системи контролю доступу на основі біометричних даних спрямована на створення програмного забезпечення, яке дозволить автоматично ідентифікувати осіб за їхнім обличчям. Вона має бути простою у використанні, здатною навчатися на нових даних і працювати з уже збереженими моделями для подальшого прогнозування. Це дозволить інтегрувати її у реальні системи контролю доступу, покращуючи безпеку та зручність входу на об'єкти.

Основні етапи навчання моделі:

- завантаження та попередня обробка зображень. Усі вхідні дані проходять через процес нормалізації, зміни розміру та перетворення у числові вектори для коректного представлення інформації;
- побудова конвеєра обробки даних. Виконується кодування міток класів, нормалізація піксельних значень та формування набору ознак, необхідних для навчання;
- розподіл вибірки на навчальний та тестовий набори. 80% даних використовується для навчання, а 20% – для тестування, що дозволяє оцінити здатність моделі до узагальнення;
- навчання моделі. Оптимізується процес класифікації шляхом мінімізації функції втрат, що дозволяє підвищити точність прогнозування та забезпечити ефективне розпізнавання обличь у реальних умовах експлуатації.

У системі повинна бути передбачена можливість вибору моделей із бази та їх використання для розпізнавання нових зображень. Збережені моделі можуть

використовуватися повторно, що спрощує інтеграцію системи у реальні сценарії експлуатації.

Для тестування розробленої моделі передбачено модуль прогнозування, який завантажує збережену модель, обробляє вхідні дані та формує висновок щодо ідентифікації особи. Ключовими аспектами тестування є:

− час розпізнавання. Система вимірює затримку під час обробки та класифікації зображення;

− поріг впевненості визначає, наскільки впевнено модель зробила передбачення;

− точність класифікації розрахунок метрик, які оцінюють продуктивність моделі.

Для реалізації методу контролю доступу на основі біометричних даних необхідно визначити оптимальну мову програмування, яка забезпечить високу продуктивність, підтримку бібліотек машинного навчання та зручність інтеграції з іншими компонентами системи. Серед найпопулярніших мов для реалізації подібних рішень варто розглянути Python, Java та C#, кожна з яких має свої особливості та переваги.

Python є однією з найпоширеніших мов для розробки рішень у сфері машинного навчання та штучного інтелекту [36]. Завдяки бібліотекам TensorFlow, OpenCV, scikit-learn та багатій екосистемі він широко використовується для побудови моделей розпізнавання обличь [37].

Java є універсальною мовою програмування, яка забезпечує високу продуктивність та кросплатформеність, що робить її популярною для корпоративних рішень [38].

C# є потужною мовою програмування, яка добре підходить для створення продуктивних програмних рішень у середовищі Windows [39]. C# також має зручний синтаксис та високу продуктивність, що робить його хорошим вибором для реалізації системи контролю доступу з використанням біометричних даних.

Для прийняття обґрунтованого рішення щодо вибору мови програмування для розробки методу контролю доступу необхідно провести їх порівняльний аналіз

за основними характеристиками. У Таблиці 1.3 наведено порівняння мов Python, Java та C# за цими параметрами.

Таблиця 1.3 – Порівняння мов програмування

Характеристика	Python	Java	C#
Швидкодія виконання коду	Низька	Висока	Висока
Оптимізація під багатопотоковість	Обмежена	Висока	Висока
Зручність у розробці	Висока	Середня	Висока
Інтеграція з .NET	Неможлива	Обмежена	Повна
Підтримка Windows-середовища	Обмежена	Середня	Висока
Сумісність із десктопними застосунками	Низька	Середня	Висока
Підтримка машинного навчання	Широка (TensorFlow, OpenCV)	Обмежена	Середня (ML.NET)
Середній час компіляції (сек)	Відсутній (інтерпретується)	5	3
Середній час виконання ML-моделі (мс)	100-200	50-100	30-80

Виходячи з проведеного аналізу, для реалізації методу контролю доступу доцільно використовувати C#. Він забезпечує високу швидкодію, підтримку багатопотоковості та ефективну інтеграцію з платформою .NET, що є важливим для створення продуктивних і масштабованих застосунків. Використання ML.NET спрощує інтеграцію методів машинного навчання у програмний продукт, забезпечуючи високу продуктивність моделі при розпізнаванні обличь. Завдяки компіляції та оптимізованому виконанню код на C# працює швидше, ніж інтерпретований Python, що критично важливо для обробки біометричних даних у

режимі реального часу. Усі ці фактори роблять C# оптимальним вибором для розробки програмно-технічного засобу для автоматизованої СП.

Оскільки для розробки методу контролю доступу було обрано мову програмування C#, необхідно визначити оптимальне середовище розробки, яке забезпечить зручне написання коду, налагодження, інтеграцію з бібліотеками машинного навчання та підтримку сучасних технологій. Серед найпопулярніших інструментів для C# можна виділити Visual Studio 2022, Visual Studio Code та JetBrains Rider.

Visual Studio 2022 є повноцінним середовищем розробки (IDE), розробленим Microsoft, яке надає широкий набір інструментів для створення, тестування та налагодження C#-застосунків [40]. Середовище підтримує ML.NET, інтеграцію з Azure, аналіз продуктивності та відлагодження багатопотокових застосунків, що робить його ідеальним для масштабних проєктів [41].

Visual Studio Code є легковаговим редактором коду, який підтримує C# через розширення C# for VS Code [42]. Він забезпечує мінімалістичне середовище з можливістю кастомізації, що підходить для невеликих проєктів та розробників, яким потрібен швидкий доступ до коду без перевантаження інструментами [43].

JetBrains Rider є потужною альтернативою Visual Studio, яка відрізняється швидкістю роботи, ефективною інтеграцією з .NET та кросплатформеністю [44]. Він підтримує аналіз коду, рефакторинг, потужний механізм підказок та глибоку інтеграцію з Unity, що робить його популярним серед розробників складних застосунків [45].

У табл. 1.4 проведено порівняння трьох основних середовищ розробки – Visual Studio 2022, Visual Studio Code та JetBrains Rider – за ключовими характеристиками, що впливають на вибір оптимального рішення.

Таблиця 1.4 – Порівняння середовищ розробки

Характеристика	Visual Studio 2022	Visual Studio Code	JetBrains Rider
Повний набір інструментів для C#	Так	Ні	Так
Підтримка ML.NET	Так	Обмежена	Обмежена
Інструменти для профілювання коду	Так	Ні	Так
Вбудовані засоби налагодження	Так	Обмежені	Так
Автоматизована інтеграція з Azure	Так	Ні	Ні
Підтримка .NET Framework та .NET Core	Так	Так	Так
Швидкість роботи	Середня	Висока	Висока
Споживання ресурсів	Високе	Низьке	Середнє
Вартість	Безкоштовна (Community Edition)	Безкоштовна	Платна

Враховуючи проведений аналіз, для реалізації системи контролю доступу доцільно використовувати Visual Studio 2022. Воно пропонує повний набір інструментів для C#-розробки, вбудовану підтримку ML.NET, засоби профілювання коду та інтеграцію з хмарними сервісами, що є важливим для розширюваності системи. Хоча Visual Studio 2022 має високе споживання ресурсів, його можливості компенсують цей недолік, особливо у складних проєктах із машинним навчанням та розширеною обробкою даних. Доступність безкоштовної версії Community Edition робить його оптимальним вибором для академічних досліджень та впровадження у промислових проєктах.

1.4 Висновки

У рамках даного розділу проведено комплексний аналіз існуючих рішень для систем контролю доступу, що дозволило визначити їхні переваги, недоліки та актуальні тенденції розвитку. Розглянуто механічні, електронні та БСП, що дало змогу оцінити їхню надійність, зручність використання, рівень безпеки та швидкодію. БС продемонстрували найвищу ефективність завдяки високій точності ідентифікації, мінімальним ризикам компрометації та швидкій обробці даних, що обґрунтовує доцільність їх застосування в сучасних рішеннях контролю доступу.

На основі аналізу світових тенденцій та практик впровадження систем пропуску в Україні встановлено, що в розвинених країнах широко застосовуються біометричні технології, інтегровані з RFID-картками, NFC-ідентифікаторами та штучним інтелектом. В Україні розвиток таких систем відбувається поступово, з акцентом на корпоративний та державний сектори, однак поки що існують бар’єри, пов’язані з фінансуванням, рівнем обізнаності користувачів та необхідністю вдосконалення нормативно-правової бази у сфері захисту персональних даних.

Здійснено постановку задачі щодо розробки програмного засобу для СП на основі біометричних даних. Для реалізації проєкту проаналізовано мови Python, Java та C#, серед яких обрано C# завдяки швидкодії, підтримці багатопотоковості та інтеграції з .NET. Також розглянуто середовища розробки, і на основі критеріїв підтримки машинного навчання та налагодження вибрано Visual Studio 2022 як оптимальне рішення.

Отримані результати та обґрунтовані вибори технологій створюють основу для подальшої реалізації системи, зокрема для розгляду застосування технологій розпізнавання зображень у наступному розділі.

2 ЗАСТОСУВАННЯ ТЕХНОЛОГІЇ РОЗПІЗНАВАННЯ ЗОБРАЖЕНЬ ДЛЯ СИСТЕМ ПРОПУСКУ НА ОСНОВІ БІОМЕТРИЧНИХ ДАНИХ

2.1 Принцип використання біометричних даних для систем пропуску

Ідентифікація особи може здійснюватися на основі її унікальних фізіологічних та поведінкових характеристик, які дозволяють точно розпізнати людину. До таких характеристик належать: відбитки пальців, структура райдужної оболонки ока, геометрія руки, риси обличчя, інфрачервоний портрет, голосові особливості, манера письма та характер натискання клавіш на клавіатурі. Ці параметри є унікальними для кожної людини, що робить їх надійним способом ідентифікації в системах контролю доступу.

Головною перевагою біометричних методів є їх унікальність та низька ймовірність збігу даних між різними особами, що робить їх значно надійнішими порівняно з традиційними способами ідентифікації, такими як паролі чи картки доступу [46]. Біометричні характеристики є практично незмінними протягом життя людини, що забезпечує стабільність ідентифікації у довгостроковій перспективі. Наприклад, шанси на те, що у двох людей співпадуть відбитки пальців на однакових пальцях рук, складають приблизно 1 до 24 мільйонів, що робить цей метод надзвичайно точним і складним для підробки.

Завдяки цій особливості біометричні технології широко застосовуються у сферах, де необхідний високий рівень безпеки, швидкий процес ідентифікації та мінімізація ризиків компрометації персональних даних. Вони активно використовуються у державних установах, фінансових організаціях, аеропортах, корпоративних структурах та навіть у мобільних пристроях для автентифікації користувачів. Однією з ключових переваг таких методів є можливість швидкої перевірки особи без необхідності носіння фізичних ідентифікаторів, що значно підвищує зручність їх використання.

У той же час біометричні технології потребують високого рівня захисту збережених даних, оскільки витік біометричної інформації може мати серйозні наслідки [47]. Саме тому сучасні системи розпізнавання використовують

криптографічні методи шифрування та локальне зберігання даних, що знижує ризики несанкціонованого доступу. Основні характеристики різних біометричних методів ідентифікації наведено в табл. 2.1, що дозволяє оцінити їх ефективність та сфери застосування.

Таблиця 2.1 – Характеристики біометричних методів ідентифікації

Метод отримання біометричних параметрів	Ймовірність відмови у доступі (%)	Ймовірність помилкової ідентифікації (без муляжу) (%)	Ймовірність помилкової ідентифікації (з муляжем) (%)	Безпека збереження образу	Вартість реалізації (у.о.)
Геометрія руки	0,2…4	0,2…1	10…75	Неможливо приховати	600 – 3000
Відбитки пальців	2...6	0,0001	10...70	Неможливо приховати	60 – 600
Сканування сітківки	0,4	6...10	–	Неможливо приховати	~4000
Райдужна оболонка	0,2…2	0,0001	–	Неможливо приховати	500 – 6000
Розпізнавання обличчя	1…9	–	–	Неможливо приховати	~55000

Кінець таблиці 2.1

Аналіз почерку	0,5…5	0,5…5	0,5...5	Частково приховується	–
Клавіатурний почерк	3…9	3...9	–	Частково приховується	–
Голосова ідентифікація	0,5...5	0,5...5	25...90 (запис)	Частково приховується	1...60

Як видно з таблиці, найвищу точність ідентифікації забезпечують методи на основі райдужної оболонки ока та відбитків пальців, які мають мінімальні ризики помилкових збігів. Водночас технології, що базуються на аналізі голосу або клавіатурного почерку, є менш надійними та можуть бути піддані атакам, зокрема за допомогою записаних зразків або емуляції поведінки користувача. Вартість реалізації також варіюється: сканування сітківки та розпізнавання обличчя є найдорожчими, тоді як відбитки пальців і геометрія руки є більш доступними для впровадження.

Методи біометричної ідентифікації поділяються на дві основні групи, серед яких найбільш поширеними є фізіологічні (статичні) характеристики, які ґрунтуються на унікальних особливостях будови тіла людини [48]. Вони забезпечують високу точність ідентифікації та можуть використовуватися у широкому спектрі застосувань, включаючи системи контролю доступу.

Реалізація фізіологічних біометричних характеристик передбачає використання спеціальних пристроїв, які реєструють та аналізують відбитки пальців, геометрію кисті, малюнок райдужної оболонки ока, сітківку та риси обличчя (табл. 2.2).

Таблиця 2.2 – Реалізація фізіологічних біометричних характеристик

Біометрична характеристика	Пристрій реєстрації	Зразок	Досліджувані ознаки
Геометрія кисті	Сканер форми руки	Тривимірне зображення кисті	Форма, розміри пальців і суглобів
Відбитки пальців	Оптичний або ультразвуковий сканер	Зображення відбитку	Малюнок гребінців і розгалужень, мікродеталі
Сітківка ока	Сканер сітківки	Зображення кровоносних судин	Розташування судинного малюнка
Райдужна оболонка	І Ч - к а м е р а , відеокамера	Ч о р н о - б і л е зображення райдужки	Смужки, борозенки та їх розташування
Розпізнавання обличчя	Камера високої роздільності	Фотографія або відеозображення	Пропорції обличчя, розташування рис

Розглянемо найбільш поширені фізіологічні методи ідентифікації.

Метод розпізнавання за геометрією кисті руки – це статичний метод біометричної ідентифікації базується на аналізі унікальної форми кисті руки, яка може бути використана для ідентифікації особи. Для цього застосовуються спеціальні пристрої, що сканують тривимірну геометрію кисті та, у деяких випадках, окремих пальців. Отримані дані обробляються, і на їх основі формується унікальний біометричний шаблон, який однозначно ідентифікує людину [49].

Розрізняють два підходи до використання геометричних характеристик кисті: аналіз геометрії (довжина пальців, ширина долоні, співвідношення розмірів кисті) та аналіз додаткових особливостей, таких як текстура шкіри, розташування складок на стиках між фалангами пальців або малюнок кровоносних судин [50]. На рис. 2.1 наведено візерунок на долоні, що складається з п'яти основних ліній (зліва) та контрольних точок із 17 геометричними характеристиками руки (справа).

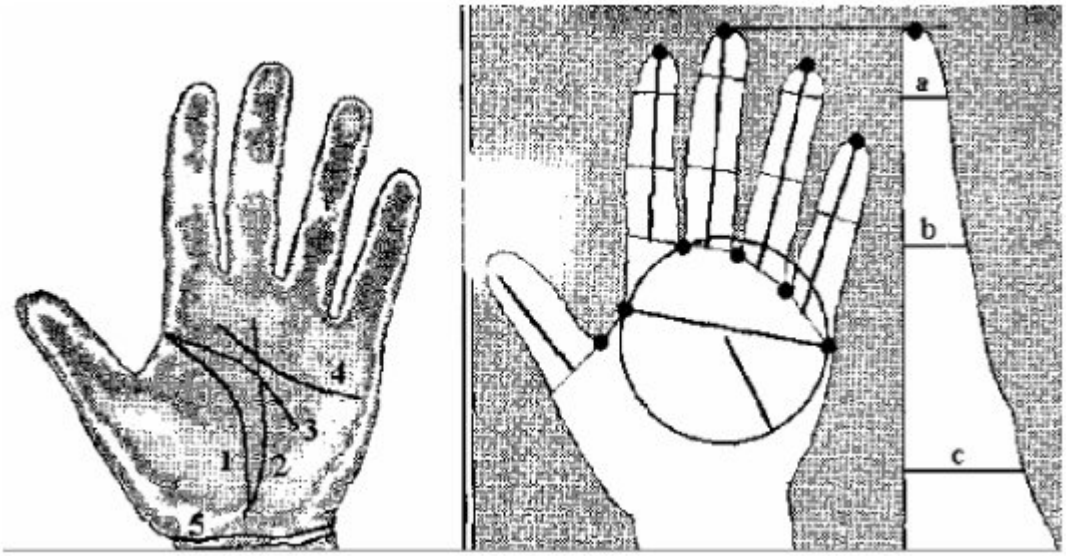


Рисунок 2.1 – Рисунок долоні людини

Основні параметри, що використовуються для аналізу:

- ширина та довжина кисті;
- радіус вписаного в долоню кола;
- довжина та ширина пальців;
- висота кисті в трьох контрольних точках.

Всі ці ознаки формують вектор значень, який використовується для ідентифікації. Під час реєстрації знімається кілька проєкцій руки користувача, для кожної з яких формується окремий вектор. На їх основі створюється еталонний клас – середнє значення параметрів для конкретної особи. У процесі експлуатації нові зразки можуть додаватися до цього класу, уточнюючи його параметри.

Для порівняння отриманого зображення з еталонним застосовуються різні підходи. Найпростіший з них – пошук найменшої відстані між вхідним вектором і центром класу. Більш складні методи включають аналіз кількох характеристик, серед яких три геометричні параметри та півтонове зображення складок шкіри на згинах між фалангами пальців. Використання цього методу значно ускладнює можливість підробки біометричних даних, оскільки текстурні особливості шкіри складно точно відтворити або імітувати.

Метод відбитки пальців – метод біометричної ідентифікації за відбитком пальця базується на аналізі унікального папілярного візерунка, що формується на

поверхні шкіри пальця [51]. Для цього використовуються спеціальні сканери, які отримують цифрове зображення відбитку, після чого воно обробляється та аналізується для виявлення характерних ознак.

Основними деталями, що використовуються для ідентифікації, є особливі точки, які легко виявити на відбитку. Серед них виділяють [52]:

- кінцеві точки – місця, де папілярна лінія різко обривається;
- точки розгалуження – області, де одна лінія розділяється на дві або більше.

Якщо роздільна здатність отриманого зображення становить 300–500 dpi, система може виявити достатню кількість деталей для точного розпізнавання. При вищій якості зображення (приблизно 1000 dpi) стає можливим аналіз внутрішньої структури папілярних ліній, включаючи розташування пор потових залоз. Проте використання таких даних у реальних умовах обмежене через складність отримання настільки високоякісного зображення поза лабораторією.

На рис. 2.2 показано приклад відбитка пальця, де пори, кінцеві точки та точки розгалуження позначені відповідними маркерами.

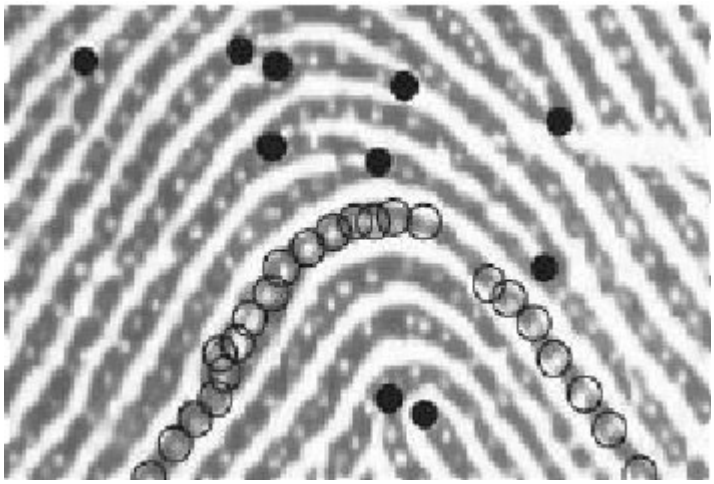


Рисунок 2.2 – Приклад відбитка пальця з позначеними маркерами

Автоматизоване розпізнавання відбитків пальців має суттєві переваги перед традиційними дактилоскопічними методами. Використання електронних безконтактних сканерів дозволяє отримати стабільне та чітке зображення,

виключаючи вплив таких факторів, як зсув або поворот пальця, зміна тиску чи якість шкіри [53]. На відміну від фарбових методів, цифрова обробка значно підвищує швидкість ідентифікації та зменшує ризик отримання спотворених зображень.

Одним із ключових факторів успішної ідентифікації є якість сканованого відбитка, оскільки саме від неї залежить вибір алгоритму формування біометричного шаблону. Чим точніше система розпізнає унікальні особливості папілярного малюнка, тим вищою буде надійність ідентифікації та захист від несанкціонованого доступу.

Метод сітківки ока, у якому сітківка ока – є внутрішньою оболонкою ока, яка відіграє ключову роль у сприйнятті світла та передачі візуальної інформації до мозку. Вона містить фоторецепторні клітини, що реагують на електромагнітне випромінювання у видимому спектрі, перетворюючи його в нервові сигнали [54]. Крім функції зорового аналізатора, сітківка має унікальну структуру судинного малюнка, яка відрізняється у кожної людини та залишається незмінною протягом життя.

Автентифікація за сітківкою здійснюється шляхом аналізу малюнка кровоносних судин, які утворюють хлоридальну систему у глибоких шарах сітківки [55]. Для цього використовується спеціальна інфрачервона камера, яка сканує внутрішню поверхню ока та створює цифровий шаблон судинної сітки. Цей шаблон порівнюється з еталонними зразками у базі даних, що дозволяє точно ідентифікувати особу.

Розпізнавання за сітківкою є одним із найбільш надійних методів біометричної ідентифікації, оскільки судинний малюнок ока має високу унікальність та практично не змінюється з часом. Використання інфрачервоного сканування забезпечує точність ідентифікації, що робить цей метод ефективним для систем з підвищеними вимогами до безпеки.

Метод райдужної оболонки, у якому радужна оболонка – це м’язовий шар у передній частині ока, розташований між рогівкою та зіницею, що виконує функцію

природного регулятора світлового потоку (рис. 2.3). Вона контролює кількість світла, яке потрапляє на сітківку, змінюючи розмір зіниці залежно від освітлення.

Рисунок 2.3 – Око людини

Структурно райдужна оболонка утворена трабекулярною мережею, що містить унікальні рельєфні особливості, такі як борозни, кільця, судини, зморшки, пігментні плями та гребінчасті стяжки. Ці особливості формують неповторний візерунок, який залишається стабільним протягом усього життя людини [56]. Хоча колір райдужної оболонки може змінюватися в ранньому дитинстві, її детальна структура залишається сталою. Винятком є лише випадки хірургічного втручання або серйозного травмування ока, які можуть змінити розташування судинного малюнка.

Процес розпізнавання особи за цим методом складається з трьох основних етапів [57]:

− захоплення зображення – використовується спеціалізована інфрачервона CCD-камера, яка забезпечує високу якість сканування. Для отримання чіткого зображення райдужної оболонки зазвичай робиться кілька знімків;

− сегментація зображення – визначаються межі райдужної оболонки, усуваються зайві елементи, такі як відблиски та повіки, що можуть впливати на точність аналізу;

− формування шаблону – отримане зображення кодується в цифровий формат для подальшого порівняння.

У процесі автентифікації сформований шаблон зіставляється із записаними в базі зразками. Для визначення ступеня схожості використовується метрика Хеммінга, яка оцінює відмінність між двома райдужними оболонками.

Ідентифікація за райдужною оболонкою є одним із найточніших методів біометричного розпізнавання, оскільки ця структура має високий рівень унікальності та стабільності. Завдяки використанню інфрачервоного сканування цей метод дозволяє отримати точні результати навіть за несприятливих умов освітлення, що робить його ефективним у системах контролю доступу з підвищеними вимогами до безпеки.

Розпізнавання обличчя – це один із найпоширеніших методів біометричної ідентифікації, який базується на аналізі унікальних геометричних характеристик обличчя людини. Ця технологія широко використовується у системах контролю доступу, що забезпечує високий рівень безпеки без необхідності використання фізичних носіїв, таких як картки чи паролі. Алгоритми розпізнавання дозволяють ідентифікувати або верифікувати особу за допомогою зображення її обличчя, що робить цей метод зручним для використання в реальному часі [58]. Завдяки розвитку нейронних мереж та комп’ютерного зору точність розпізнавання значно зросла, дозволяючи використовувати цей підхід навіть у складних умовах освітлення та змінених ракурсах. Така технологія впроваджується в корпоративних системах контролю доступу, мобільних пристроях, аеропортах і банківських установах, де необхідна швидка та надійна перевірка особи.

Для точної ідентифікації система аналізує ключові точки на обличчі, що дозволяє створити унікальну математичну модель для кожного користувача [59]. На рис. 2.4 зображено характерні точки, які використовуються в алгоритмах розпізнавання, включаючи відстань між очима, розташування брів, контури носа, губ і щелепи.

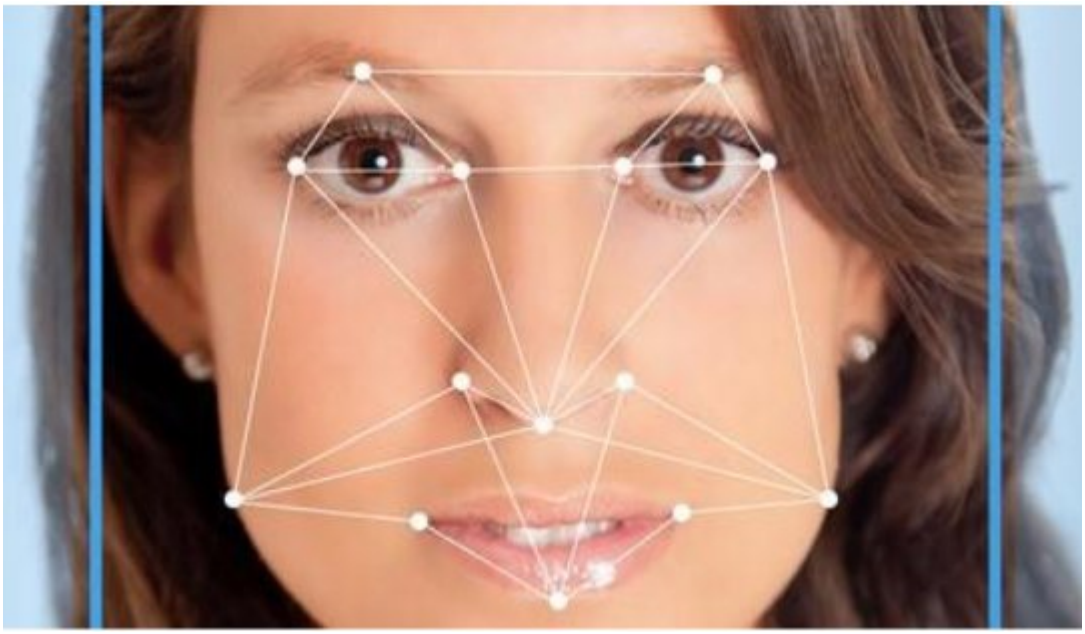


Рисунок 2.4 – Характерні точки для розпізнавання обличчя

Обробка обличчя здійснюється в кілька етапів: спочатку система виявляє обличчя на зображенні, після чого визначає ключові точки, нормалізує масштаб та орієнтацію зображення, а потім формує вектор ознак, який використовується для порівняння з наявними зразками. Використання нейронних мереж дозволяє зменшити похибки при розпізнаванні, навіть якщо користувач змінює міміку або його обличчя частково закрите.

Метод розпізнавання обличчя є ефективним рішенням для біометричної ідентифікації, яке поєднує високу точність, зручність та швидкість роботи. Завдяки використанню ключових точок та алгоритмів глибокого навчання, ця технологія забезпечує стабільну ідентифікацію навіть у несприятливих умовах. Розпізнавання обличчя продовжує активно розвиватися, стаючи невід’ємною частиною сучасних систем контролю доступу та безпеки.

Більшість біометричних систем працюють за єдиним алгоритмом, який передбачає збереження та подальше порівняння унікальних характеристик користувача. На першому етапі система проводить реєстрацію, під час якої знімається та обробляється біометрична ознака, наприклад, відбиток пальця, контури обличчя або голосовий зразок. Деякі системи можуть запитувати кілька варіантів зразка, щоб сформувати найбільш точне представлення біометричних

даних. Отримана інформація кодується в цифровий формат та зберігається у базі даних для подальшої перевірки.

Після реєстрації система може потребувати додаткового підтвердження особи. Це може бути введення PIN-коду, використання смарт-картки, яка містить біометричний шаблон, або автентифікація через інший фактор безпеки. Такий підхід забезпечує двофакторний захист, що значно підвищує рівень безпеки системи.

Процес ідентифікації у біометричних системах включає чотири основні етапи:

− запис – система реєструє фізіологічний або поведінковий параметр користувача;

− обробка – з отриманого зображення або сигналу виділяються ключові унікальні характеристики, які формують біометричний шаблон;

− порівняння – отримані дані зіставляються з уже наявними зразками у базі;

− прийняття рішення – система визначає, чи співпадає наданий зразок із збереженим у базі, та видає відповідний результат.

Завдяки сучасним алгоритмам обробки та нейронним мережам, БС забезпечують високу точність розпізнавання та можуть застосовуватися для контролю доступу, верифікації особи та забезпечення інформаційної безпеки.

Сучасні БС ідентифікації зберігають унікальні цифрові шаблони, які прив’язані до конкретних користувачів, що мають право доступу. У процесі розпізнавання пристрій, наприклад, сканер відбитків пальців, система розпізнавання обличчя або аналізатор голосу, зчитує біометричні дані особи. Отримана інформація обробляється та порівнюється із записаними у реєстраційній базі шаблонами, що дозволяє ідентифікувати або верифікувати користувача. Схематично цей процес зображено на рис. 2.5.

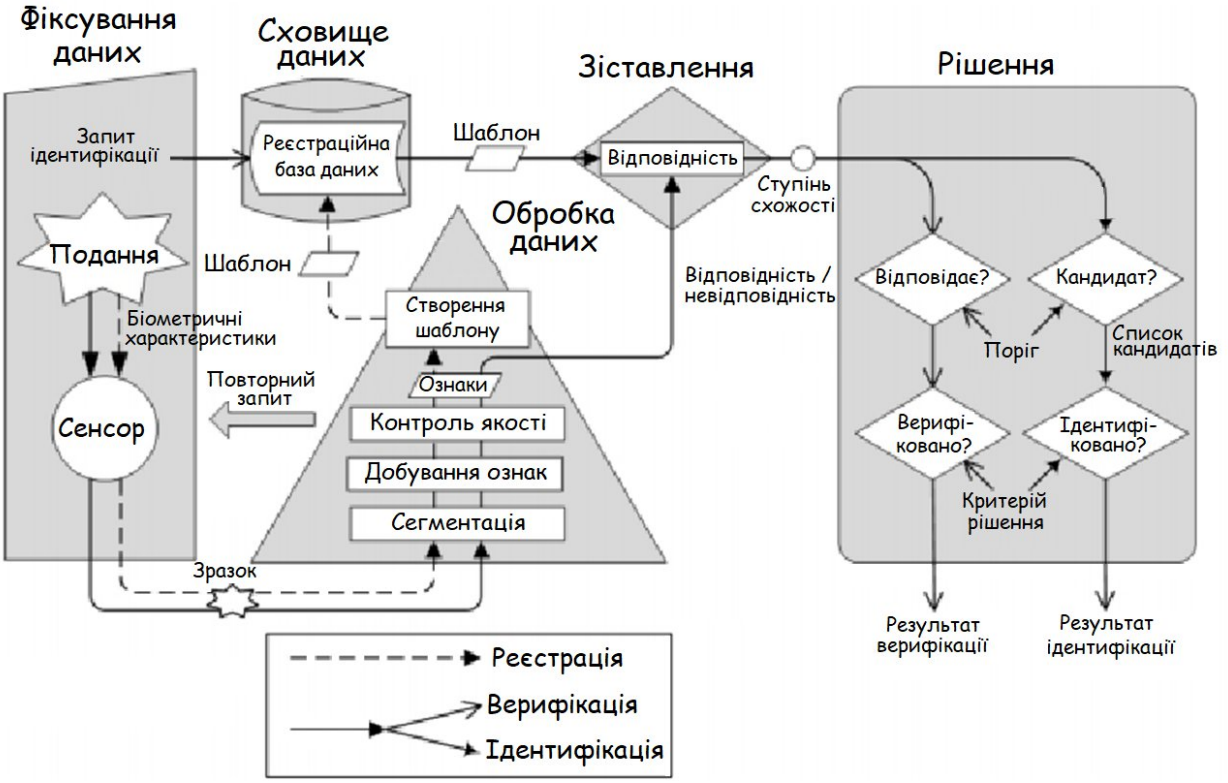


Рисунок 2.5 – Функціональна схема роботи біометричної системи

Як показано на рисунку, основою процесу є формування біометричного шаблону, який містить лише ключові характеристики об'єкта ідентифікації. Для зменшення розміру файлів та підвищення безпеки система не зберігає повні біометричні зображення, а лише оброблені математичні моделі. Це унеможливлює відновлення вихідних даних із шаблону, що підвищує конфіденційність та захист особистої інформації користувачів.

Робота біометричної системи включає три основні рівні обробки даних:

− фіксація – пристрій реєструє необроблені біометричні дані (наприклад, зображення обличчя, голосову хвилю або відбиток пальця);

− проміжна обробка – система виділяє характерні ознаки з отриманої інформації, перетворюючи її у придатний для порівняння формат. На цьому етапі дані ще не є повністю оптимізованими для збереження у шаблоні;

− фінальна обробка – сформований біометричний шаблон порівнюється із записаними у базі даними. Саме на цьому рівні система приймає рішення про збіг або невідповідність користувача.

Такий підхід дозволяє створити ефективну та безпечну систему ідентифікації, що може застосовуватися у системах контролю доступу, фінансових транзакціях та інших сферах, де необхідний високий рівень автентифікації користувачів.

Алгоритм обробки біометричних даних проходить через кілька етапів, що зображено на рис. 2.6.

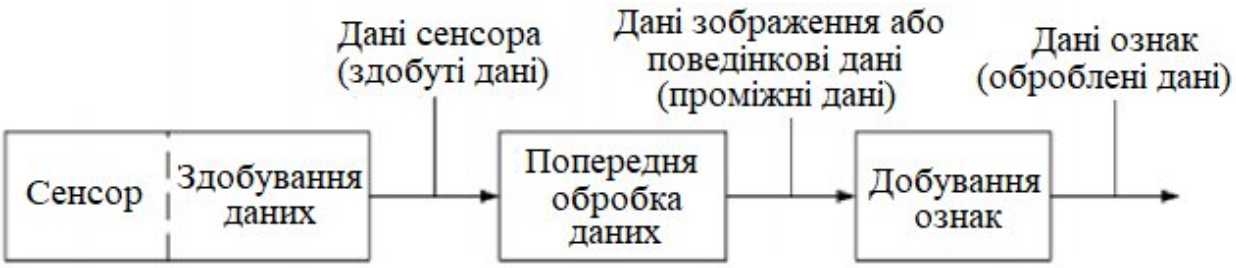


Рисунок 2.6 – Обробка біометричних даних

На початковому етапі сенсор реєструє біометричні характеристики користувача, отримуючи сирі дані. Далі система виконує попередню обробку, у ході якої зображення або поведінкові особливості переводяться у більш структурований формат. Потім відбувається виділення ключових ознак, які дозволяють сформувати унікальний біометричний шаблон.

Важливо зазначити, що при реєстрації нового користувача система створює унікальний шаблон виключно на основі розпізнаних біометричних ознак, без збереження повних зображень або сирих даних. Це забезпечує конфіденційність та ефективність подальшої перевірки особи у процесі ідентифікації чи верифікації.

2.2 Опис моделей нейронних мереж для біометричної ідентифікації

На сьогодні існує декілька типів нейронних мереж, що застосовуються для обробки та класифікації зображень. Однією з найпоширеніших архітектур є багатошаровий перцептрон, який здатний аналізувати вхідні зображення та відносити їх до певних категорій після попереднього навчання . Завдяки своїй

гнучкості та ефективності цей тип мереж широко використовується у задачах розпізнавання біометричних характеристик, зокрема ідентифікації обличчя.

Одним із найефективніших підходів у цій галузі є згорткова нейронна мережа (ЗНМ), яка демонструє високу точність при розпізнаванні облич [60]. Основними особливостями цієї архітектури є локальні рецепторні поля, спільні ваги та ієрархічна структура з просторовими семплінгами, що дозволяє мережі ефективно адаптуватися до змін у вхідних даних [61]. Завдяки такій організації ЗНМ має високу стійкість до змін масштабу, поворотів, зміни ракурсу та інших спотворень, які можуть впливати на якість розпізнавання. Наприклад, тестування ЗНМ на наборі даних ORL, що містить зображення облич із незначними змінами освітлення та масштабу, показало 96% точності розпізнавання.

Серед ключових переваг нейронних мереж для розпізнавання обличь можна виділити [62]:

− високу точність розпізнавання, навіть у випадках часткового перекриття або змін освітлення;

− стійкість до поворотів та змін ракурсу, що робить метод ефективним для реальних умов застосування.

Однак, використання нейронних мереж також має певні недоліки:

− необхідність повного перенавчання при додаванні нових осіб у базу, що значно збільшує час обробки даних;

− складність налаштування архітектури, оскільки вибір оптимальної кількості нейронів, шарів та алгоритмів оптимізації є нетривіальним завданням.

Незважаючи на ці виклики, нейронні мережі залишаються найперспективнішим підходом у розпізнаванні обличь, забезпечуючи високу точність і надійність у сфері біометричної ідентифікації.

Існує велика кількість архітектур штучних нейронних мереж (ШНМ), що використовуються для аналізу зображень, проте найбільш ефективними у розпізнаванні обличчя є ЗНМ. Вони відрізняються між собою кількістю шарів, особливостями навчання та швидкістю обробки даних, що безпосередньо впливає на точність ідентифікації. Залежно від архітектури, кожна модель демонструє різні

показники продуктивності, зокрема час розпізнавання та стійкість до змін вхідних даних. Нижче розглянуто найпоширеніші архітектури ЗНМ, які застосовуються для біометричної ідентифікації обличчя.

2.2.1 Архітектура LeNet-5

LeNet-5 – це одна з перших ЗНМ, яка була розроблена для розпізнавання рукописних цифр, але її принципи знайшли застосування в біометричних системах ідентифікації, зокрема розпізнаванні обличчя. Вона складається з багатьох шарів, що автоматично виділяють ключові особливості зображення, усуваючи потребу у ручному виборі ознак [63]. Завдяки каскадному застосуванню згорткових операцій і підсемплінгу мережа виявляє стабільні просторові патерни, що дозволяє їй бути стійкою до варіацій у зображенні, таких як зміни масштабу чи зсуви [64]. Саме ця особливість робить її ефективною для розпізнавання обличчя, де важливо враховувати зміни в позах та виразах обличчя.

На рис. 2.7 наведена архітектура LeNet-5, яка складається з чергування згорткових шарів і шарів підсемплінгу, що поступово зменшують розмірність даних, зберігаючи при цьому найважливіші характеристики.

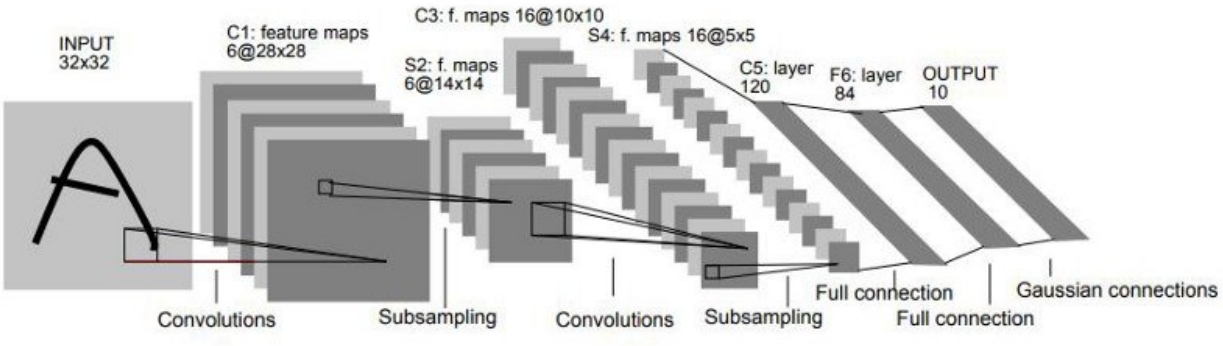


Рисунок 2.7 – Архітектура LeNet5 [65]

Вхідне зображення розміром 32×32 пікселі проходить через перший згортковий шар (C1), який генерує 6 карт ознак. Після цього шар підсемплінгу (S2) зменшує їхній розмір, що зменшує надмірність інформації та підвищує

інваріантність до змін у вхідному зображенні. Наступні шари C3 та S4 повторюють цей процес, але вже з 16 картами ознак, що дозволяє мережі розпізнавати більш складні структури. Далі йдуть повнозв'язні шари (C5, F6), які обробляють отриману інформацію для остаточної класифікації, після чого на виході формується вектор з 10 можливих класів.

Дана структура дозволяє мережі ефективно аналізувати зображення та виділяти найважливіші характеристики, що є ключовим аспектом для застосування у системах розпізнавання обличчя. Висока інваріантність до змін положення та масштабу дозволяє LeNet-5 працювати навіть за умов змін освітлення чи часткового перекриття обличчя [66]. Завдяки своїй ефективності та низьким обчислювальним витратам, ця архітектура стала основою для багатьох сучасних моделей, що використовуються у біометричних системах контролю доступу.

2.2.2 AlexNet

AlexNet – це одна з перших глибоких ЗНМ, яка здійснила прорив у сфері комп'ютерного зору. Вона була розроблена для розпізнавання зображень і показала високу ефективність у завданнях класифікації [67]. Завдяки своїй здатності автоматично виділяти ключові ознаки об'єктів на зображеннях, AlexNet знайшла застосування у біометричних системах, зокрема в розпізнаванні обличчя [68]. Глибока структура мережі дозволяє їй розпізнавати складні патерни та забезпечує стійкість до змін ракурсу, освітлення та шумів [69]. Використання цієї архітектури у біометричних системах контролю доступу дає змогу покращити точність ідентифікації та мінімізувати кількість помилкових спрацьовувань.

На рис. 2.8 представлена архітектура AlexNet, яка складається із п'яти згорткових шарів, трьох шарів субдискретизації (Pooling) та двох повнозв'язних шарів перед вихідним шаром Softmax.

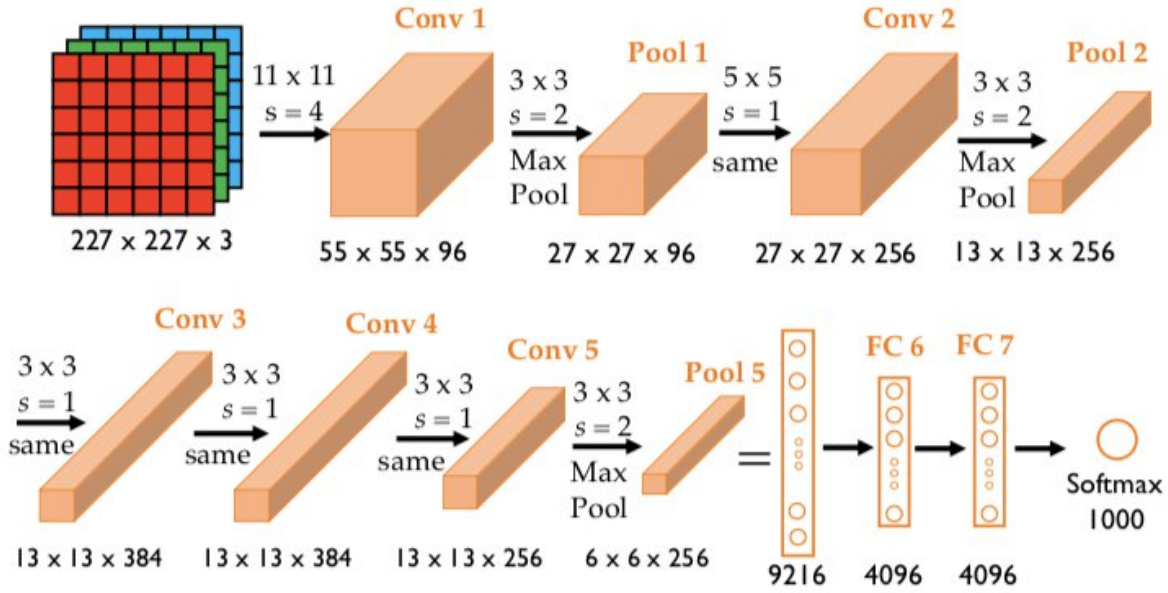


Рисунок 2.8 – Архітектура LeNet5 [70]

Вхідні зображення розміром 227×227 пікселів проходять через перший згортковий шар, що використовує ядро 11×11 і крок згортки 4, створюючи 96 карт ознак. Після цього застосовується операція максимального підсемплінгу, яка зменшує розмірність карти ознак, усуваючи надлишкову інформацію. Далі проходять наступні згорткові шари, що використовують менші ядра (3×3), але більшу кількість карт ознак (до 384 на третьому шарі), що дозволяє мережі витягувати більш складні та детальні ознаки зображення. В останніх шарах використовується ще один рівень Pooling, після чого нейронна мережа передає отриману інформацію у два повнозв’язні шари (4096 нейронів у кожному), які виконують класифікацію.

НМ AlexNet забезпечує високу точність у розпізнаванні обличчя завдяки глибокій архітектурі та ефективному виділенню ознак. Її використання у біометричних системах дозволяє знизити рівень помилкової ідентифікації та покращити розпізнавання осіб у складних умовах, таких як часткове перекриття обличчя чи погане освітлення [71].

Саме ця модель стала основою для подальшого розвитку згорткових нейронних мереж, що використовуються у сучасних системах контролю доступу та безпеки.

2.2.3 ResNet

ResNet (Residual Neural Network) – це глибока ЗНМ, що використовується для аналізу зображень та вирішення складних завдань комп’ютерного зору. Завдяки своїй архітектурі вона дозволяє уникнути проблеми зникнення градієнта, що часто виникає під час навчання дуже глибоких нейронних мереж [72]. ResNet знаходить широке застосування у системах розпізнавання обличчя, оскільки її архітектура забезпечує ефективне вилучення ознак та збереження суттєвих деталей навіть у складних умовах, таких як зміни освітлення чи часткове перекриття обличчя [73]. Її використання дозволяє покращити точність біометричних систем та зробити ідентифікацію осіб більш надійною.

На рис. 2.9 представлена архітектура ResNet, яка базується на концепції залишкових (residual) блоків. Головна ідея цього підходу – використання прямих з'єднань (skip connections), що дозволяють передавати інформацію між шарами без значного спотворення. Це дає можливість глибокій мережі навчатися ефективніше, зменшуючи ризик деградації продуктивності при збільшенні кількості шарів. Вхідне зображення розміром 224×224 пікселів проходить через початковий згортковий шар 7×7, після чого застосовується нормалізація пакетів (Batch Normalization) та функція активації. Далі йдуть шари підсемплінгу (MaxPooling), що допомагають зменшити розмірність даних, зберігаючи при цьому найважливіші ознаки. Основна частина ResNet складається з комбінації згорткових і залишкових блоків, які можуть мати різну кількість рівнів залежно від глибини моделі.

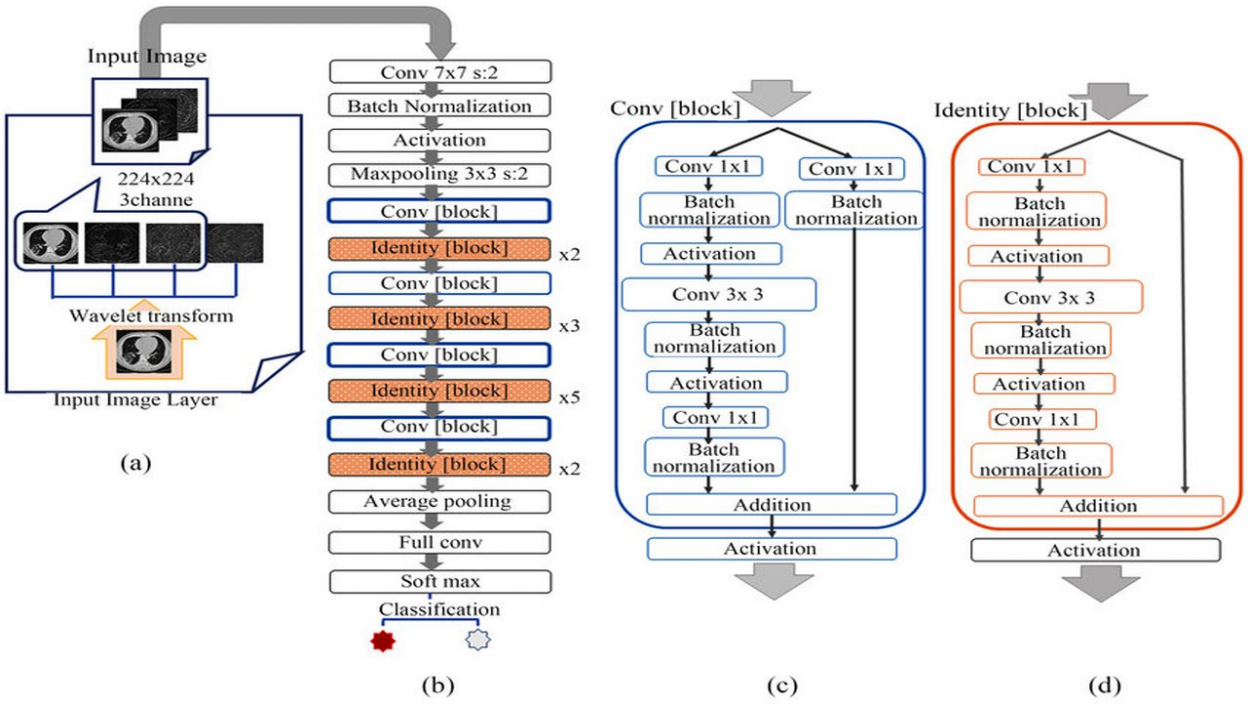


Рисунок 2.9 – Архітектура ResNet [74]

У залишкових блоках передбачено ідентичні (Identity) шляхи, що передають інформацію вперед, уникаючи втрати важливих характеристик. Завершується мережа шаром середнього згорткового підсумовування (Average Pooling), повнозв'язним шаром і виходом Softmax для класифікації.

Архітектура ResNet дозволяє будувати дуже глибокі нейронні мережі, які зберігають стабільність під час навчання і демонструють високу точність у розпізнаванні обличь [75]. Завдяки своїй структурі вона є однією з найефективніших моделей, що застосовується у сучасних біометричних системах, забезпечуючи стійкість до спотворень, змін ракурсу та освітлення. Ця технологія відіграє ключову роль у розробці інтелектуальних систем безпеки, які потребують високого рівня точності та надійності.

2.2.4 YOLOv11

YOLOv11 (You Only Look Once) – це сучасна глибока нейронна мережа для детекції та розпізнавання об'єктів на зображеннях у реальному часі. Завдяки

високій швидкості та точності ця модель знайшла застосування у системах біометричної ідентифікації, включаючи розпізнавання обличчя для контролю доступу [76]. Головна особливість YOLOv11 – це одноетапний підхід до виявлення об’єктів, який дозволяє мережі швидко аналізувати вхідне зображення та визначати положення об’єктів без необхідності багаторазового сканування зони інтересу [77]. Це робить YOLOv11 особливо корисною для застосувань, що потребують оперативної обробки даних, наприклад, відеоспостереження чи автоматизованих систем безпеки.

На рисунку 2.10 зображена архітектура YOLOv11, яка складається з трьох основних частин: Backbone (основний блок), Neck (проміжний блок) та Head (вихідний блок).

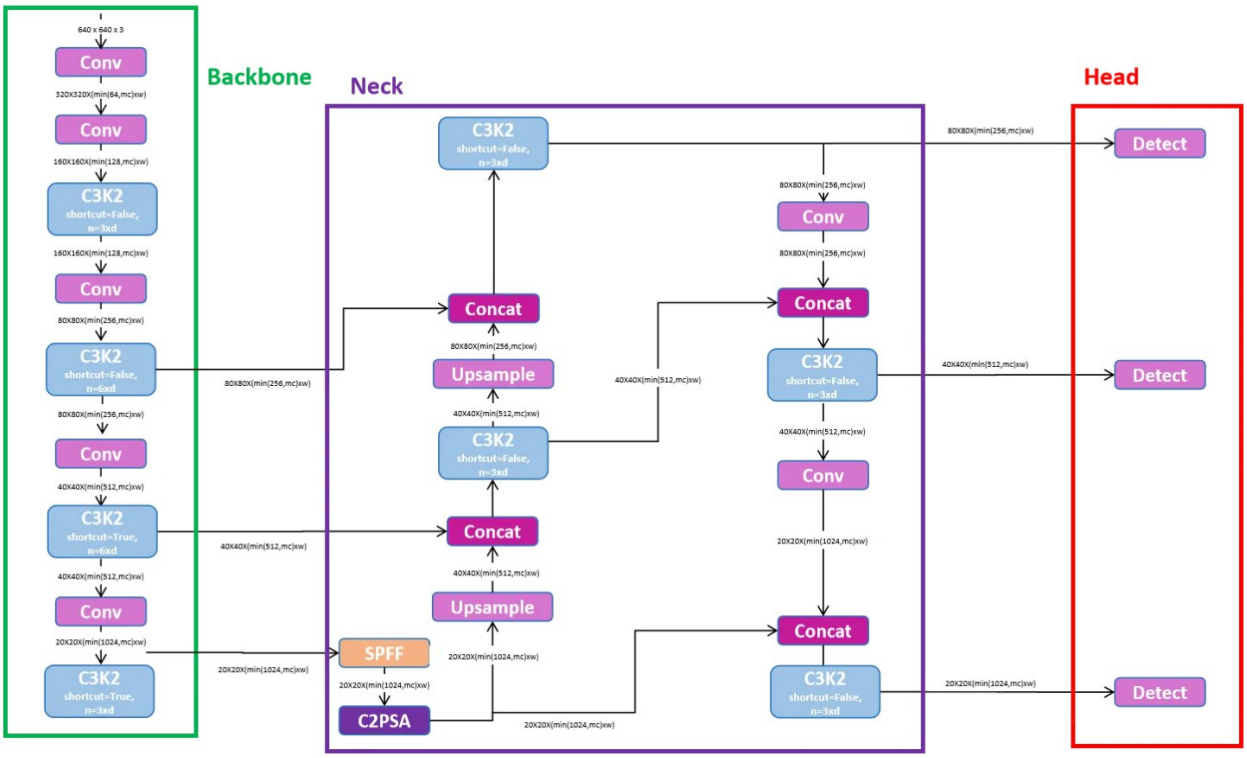


Рисунок 2.10 – Архітектура YOLOv11 [78]

Backbone відповідає за екстракцію ознак із зображення. Він містить послідовність згорткових шарів, що поступово зменшують розмірність вхідного зображення, зберігаючи при цьому ключові особливості. Neck включає додаткові рівні, такі як Upsample та Concat, які використовують механізми підвищення

деталізації та комбінування ознак з різних рівнів. Завдяки цьому модель отримує багатошарове представлення об'єктів, що допомагає точніше визначати обличчя навіть у складних умовах. Head – це заключний рівень, який відповідає за детекцію об'єктів та їх класифікацію, використовуючи спеціалізовані блоки для обробки фінальних ознак.

Архітектура YOLOv11 дозволяє значно підвищити ефективність розпізнавання обличчя у реальному часі, забезпечуючи мінімальні затримки при аналізі відеопотоків [79]. Її використання у біометричних системах дозволяє не лише ідентифікувати особу, а й контролювати доступ до об’єктів із високими вимогами до безпеки. Завдяки оптимізованій структурі YOLOv11 є однією з найефективніших моделей для інтеграції у сучасні системи контролю та відеоспостереження.

2.3 Підготовка даних до навчання нейронної мережі

Для навчання моделі розпізнавання обличь було використано датасет Face Recognition Dataset, який містить різноманітні зображення осіб, представлені у різних умовах освітлення, ракурсах та з різними виразами обличчя [80]. Така різноманітність даних дозволяє створити стійку до змін модель, яка ефективно ідентифікує особу навіть у складних умовах, наприклад, при частковому перекритті обличчя або зміні освітлення. Основною метою використання цього набору даних є створення високоточного алгоритму біометричної ідентифікації, який може бути застосований у системах безпеки, відеоспостереження та контролю доступу.

Дані організовані у вигляді категоризованих зображень, де кожна особа представлена у вигляді окремої папки з відповідними мітками. Це дозволяє нейронній мережі запам’ятовувати та аналізувати унікальні риси кожного користувача, що значно покращує точність класифікації. Крім того, велика кількість варіацій обличь у датасеті допомагає моделі узагальнювати отримані знання, що підвищує її ефективність у реальних умовах експлуатації. Використання зображень із різними кутами огляду та освітленням дозволяє системі краще

адаптуватися до змінних зовнішніх факторів, що є критичним для біометричних систем контролю доступу.

На рис. 2.11 зображено приклад тренувальних даних, які використовуються для навчання нейронної мережі. Видно, що кожен набір зображень відноситься до конкретної особи, представлений у різних умовах, що дає змогу моделі визначати стабільні та унікальні риси обличчя незалежно від зовнішніх факторів.

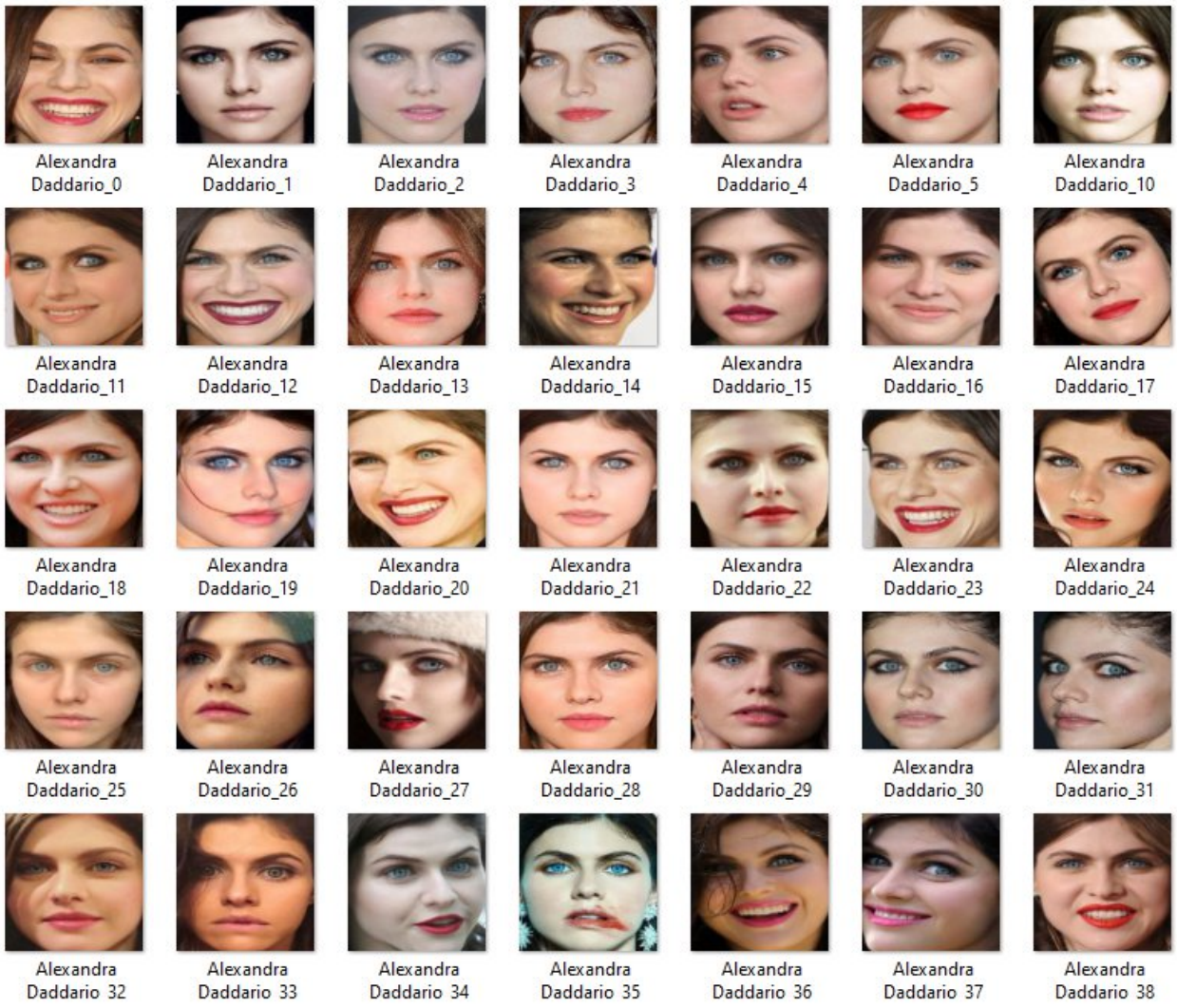


Рисунок 2.11 – Приклад навчальних зображень з Face Recognition Dataset

Зображення представлені у форматі, який містить ідентифікаційні мітки для кожного об’єкта. Це дає можливість мережі ефективно класифікувати осіб та коригувати ваги моделі під час навчання. Завдяки такій структурі підготовлені дані допомагають покращити стабільність розпізнавання, зменшуючи кількість

помилкових позитивних або негативних ідентифікацій. Таким чином, використання різнорідного датасету з високою варіативністю забезпечує гнучкість і надійність моделі у розпізнаванні осіб у реальних сценаріях.

2.4 Налаштування моделей нейронних мереж для тренування

У процесі розроблення методу та програмно-технічного засобу для СП на основі біометричних даних постало завдання побудови класу моделей, здатних автоматизовано виконувати розпізнавання обличчя за наявними зображеннями. Серед усіх підходів було розглянуто як згорткові нейронні мережі, так і класичні методи машинного навчання на основі лінійних класифікаторів.

Для побудови конвеєра було використано середовище ML.NET, яке дає змогу кроково перетворювати сирі дані (зображення) на придатні для моделювання ознаки, а потім – застосувати відповідний класифікатор. На вхід надходять зображення, зібрані з робочих директорій. Кожне зображення було попередньо зменшено до фіксованого розміру, після чого кожен піксель трансформовано у три складові (R, G, B), нормовані за діапазоном.

Наступним кроком виконується перетворення текстових міток, що позначають приналежність до певної особи, у формат так званого «ключа» (Key). Це вимога, притаманна більшості мультикласових тренерів у ML.NET. Далі бажано додатково відмасштабувати (нормалізувати) ознаки, щоб уникнути надмірних коливань під час обчислень [81]. Усі сформовані вектори пікселів потрапляють до стовпця, прийнятого за стандарт у ML.NET – «Features».

Завершальною операцією в конвеєрі є безпосередньо навчання, що складається з оголошеного тренера та налаштування його гіперпараметрів. Після проходження необхідної кількості ітерацій результатом стає модель, у якій під час передбачення автоматично виконується зворотне перетворення ключа в текстову назву класу (тобто особи).

Початково розглядалися моделі на основі глибинних згорткових мереж, але вони внутрішньо використовують середовище TensorFlow, яке потребує

процесорів із підтримкою інструкцій AVX. В умовах сумісності з устарілими або спеціалізованими CPU постало завдання перейти на методи, що не покладаються на бібліотеки TensorFlow.

Серед різноманітних підходів у ML.NET було обрано метод максимальної ентропії на основі обмеженого спуску за градієнтом для багатокласової класифікації, який реалізує логістичну регресію для мультикласових задач із застосуванням алгоритму L-BFGS (Бройдена – Флетчера – Голдфарба – Шанно). Ця модель не тільки не вимагає глибинних архітектур, а й забезпечує помірну ефективність на зображеннях малої розмірності або якщо вже виконано суттєве попереднє вилучення ознак.

У межах проєкту для тренера було застосовано такі настройки (гіперпараметри):

− кількість ітерацій (MaximumNumberOfIterations). Визначає, скільки разів алгоритм «пройдеться» по даних, уточнюючи вагові коефіцієнти. Збільшення цього числа може підвищувати точність, проте занадто велике значення призводить до тривалішої оптимізації, а іноді й до перенавчання. В експериментальних випробуваннях використовували значення в межах 100, коригуючи його залежно від обсягу даних та спостережуваної зміни похибки.

− параметри регуляризації. L1-регуляризація схильна «занулювати» ваги, роблячи модель більш розрідженою (sparse). Це корисно, коли потрібно відсіяти несуттєві ознаки, а також зменшити ризик перенавчання. L2-регуляризація «розгладжує» ваги, не даючи їм набувати надто великих значень. Її підвищення підсилює ефект боротьби з перенавчанням, але надто велике – може знизити точність, недорозвинувши модель;

− розмір історії (HistorySize). Під час використання L-BFGS алгоритм зберігає обмежену кількість попередніх градієнтів для наближення гессіану. Типовим значенням є 20, і воно зазвичай збалансовує швидкість і використання пам’яті;

− перетворення міток у ключ. Обов’язкове, щоби тренер розпізнавав цільовий стовпець як мультиклас. Після завершення навчання мітки знову конвертуються назад у читабельні назви (MapKeyToValue);

− додаткова нормалізація пікселів. Хоча в більшості експериментів зображення вже зводяться до діапазону [0..1] під час формування масиву пікселів, також застосовувався алгоритм мін-макс нормалізації, який додатково усуває вибіркові зсуви й масштабування. У деяких випадках це допомагає швидше сходитися моделі.

У ході низки експериментів було з’ясовано, що правильне підбирання L_1/L_2-параметрів дає змогу досягти кращої узагальнювальної здатності на обмежених наборах даних, особливо в поєднанні з додатковою нормалізацією пікселів. Окрім цього, збільшення кількості ітерацій позитивно впливає на точність, але непропорційно підвищує час навчання. Якщо процесор не підтримує сучасних інструкцій, LbfgsMaximumEntropy стає гнучкою альтернативою глибинній мережі, дозволяючи виконувати розпізнавання обличчя на менш продуктивних системах.

Отже, у межах даного налаштування моделей було зосереджено на параметрах класичного мультклассового тренера з лінійною функцією, за рахунок чого вдалося уникнути апаратних обмежень TensorFlow і забезпечити прийнятну якість розпізнавання в умовах типових для системи контролю доступу.

2.6 Висновки

У рамках даного розділу проведено аналіз використання технологій розпізнавання зображень для біометричних систем пропуску, що дозволило визначити основні методи ідентифікації осіб та підходи до їх реалізації. Детально розглянуто біометричні характеристики, такі як геометрія кисті, відбитки пальців, сканування сітківки та райдужної оболонки ока, розпізнавання обличчя, аналіз почерку, клавіатурний почерк та голосова ідентифікація. Було розроблено функціональну схему роботи біометричної системи та алгоритм обробки

біометричних даних, що є ключовими елементами побудови надійної системи контролю доступу.

Проаналізовано архітектури сучасних нейронних мереж, що застосовуються для біометричної ідентифікації, зокрема LeNet-5, AlexNet, ResNet та YOLOv11. Розгляд цих моделей дозволив оцінити їхні можливості щодо точності та швидкості розпізнавання, що є критично важливими параметрами для інтеграції у систему пропуску. Виконано детальний опис підготовки даних для навчання нейронної мережі, що включало завантаження та аналіз датасету Face Recognition Dataset із платформи Kaggle. Описано приклади тренувальних даних та особливості їхньої структури, що дозволило підготувати якісний набір для навчання моделі.

Значну увагу приділено налаштуванню параметрів нейронних мереж, необхідних для коректного навчання. Визначено оптимальну кількість ітерацій, налаштовано параметри регуляризації, включаючи L_1-регуляризацію, яка сприяє зануленню ваг, та L_2-регуляризацію, що допомагає уникнути переобчислення. Виконано перетворення міток у ключові значення та застосовано додаткову нормалізацію пікселів, що покращує стабільність навчання та підвищує точність моделі.

Отримані результати дозволяють перейти до наступного етапу дослідження, зокрема розробки методу та алгоритму використання біометричних даних для систем пропуску. Визначені в цьому розділі підходи до побудови моделей, підготовки та обробки даних стануть основою для реалізації практичної частини, що забезпечить ефективність та надійність запропонованої системи контролю доступу.

3 МЕТОД ТА АЛГОРИТМ ВИКОРИСТАННЯ БІОМЕТРИЧНИХ ДАНИХ ДЛЯ СИСТЕМ ПРОПУСКУ

3.1 Метод та алгоритм навчання моделі для розпізнавання обличчя людини

Для побудови системи розпізнавання обличчя для контролю доступу було обрано метод максимальної ентропії, який реалізує логістичну регресію для багатокласової класифікації з використанням алгоритму L-BFGS (Бройдена – Флетчера – Голдфарба – Шанно). Даний підхід забезпечує ефективну обробку та класифікацію вхідних зображень без використання глибоких нейронних мереж, що дозволяє уникнути залежності від апаратного забезпечення, сумісного з TensorFlow.

Метод максимальної ентропії базується на ідеї, що серед усіх можливих розподілів ймовірностей слід обирати той, який має найвищу ентропію, тобто є найменш детермінованим, за умови дотримання накладених обмежень. У контексті класифікації це означає, що модель прагне зробити передбачення, максимально рівномірно розподілене між можливими класами, поки не буде отримано достатньо інформації для конкретного висновку.

Метод максимальної ентропії використовує умовний розподіл ймовірностей для передбачення класу об'єкта y за набором ознак x. Ймовірність належності до класу y визначається за допомогою логістичної (*softmax*) функції:

$$P(x,w) = \frac{e^{w_y \cdot x}}{\Sigma_{y'} e^{w_{y'} \cdot x}}, \tag{3.1}$$

де:

- x – вхідний вектор ознак (перетворене зображення обличчя);
- w_y – ваговий вектор, що відповідає класу y;

– $\sum_y$, $e^{w_y \cdot x}$ – нормувальний коефіцієнт, що забезпечує коректне представлення ймовірності.

Функція втрат у даному методі виражається через перехресну ентропію:

$$L(w) = -\sum_{i=1}^{N} \sum_{y \in Y} 1(y_i = y) \log \log P(x,w) \quad (3.2)$$

де:

– N – кількість навчальних прикладів;

– Y – набір усіх можливих класів (осіб, які потрібно розпізнати);

– $1(y_i = y)$ – індикаторна функція, що приймає значення 1, якщо y_i є правильним класом для x_i, і 0 в іншому випадку.

Для мінімізації цієї функції втрат використовується алгоритм L-BFGS, який дозволяє знайти оптимальні значення ваг w шляхом ітеративного оновлення параметрів моделі.

L-BFGS є методом обмеженої пам’яті для квазі-Ньютона оптимізації, що дозволяє ефективно працювати з великими розмірностями параметрів без значного використання оперативної пам’яті.

Головні особливості алгоритму L-BFGS:

– використовує градієнтні оновлення, але без явного обчислення повної гессіан-матриці (матриці другої похідної функції втрат);

– використовує наближене збереження обмеженої кількості останніх оновлень градієнта, що дозволяє ефективно оцінити напрямок оновлення ваг;

– підходить для великих наборів параметрів, що важливо для класифікації високорозмірних векторів ознак, отриманих із зображень.

Метод оновлює параметри згідно з рівнянням:

$$w_{t+1} = w_t - aH^{-1} \nabla L(w_t) \quad (3.3)$$

де:

- α – коефіцієнт навчання;
- H^{-1} – наближена обернена гессіан-матриця;
- $\nabla L(w_t)$ – градієнт функції втрат.

L-BFGS застосовується для розв’язання опуклих оптимізаційних задач, таких як логістична регресія, і забезпечує швидку збіжність навіть у випадках великої розмірності ознак.

Переваги методу максимальної ентропії із застосуванням обмеженого спуску за градієнтом:

- незалежність від глибоких нейронних мереж. Метод не потребує використання CNN або TensorFlow, що дозволяє застосовувати його на системах зі старішими процесорами без підтримки інструкцій AVX;
- помірна обчислювальна складність. Завдяки оптимізації методом L-BFGS навчання проходить значно швидше, ніж у випадку глибинного навчання, при збереженні високої точності;
- гнучкість та інтерпретованість. Оскільки метод базується на логістичній регресії, результати передбачення легко інтерпретувати, а регуляризаційні параметри дозволяють уникнути перенавчання;
- збереження інформації про ознаки. Оскільки алгоритм використовує безпосередньо вхідний вектор ознак, він не вимагає складних перетворень або попереднього вилучення характеристик, що спрощує інтеграцію в реальні системи.

Отже, метод максимальної ентропії із застосуванням обмеженого спуску за градієнтом був обраний для задачі розпізнавання обличчя у системі контролю доступу завдяки своїй незалежності від вимогливих до ресурсів архітектур, швидкій збіжності та високій точності класифікації. Використання логістичної регресії у багатокласовій класифікації дозволило забезпечити стабільну роботу навіть на пристроях із обмеженими обчислювальними можливостями. Алгоритм L-BFGS дозволяє ефективно оптимізувати вагові коефіцієнти, що робить його придатним для застосування в реальних умовах біометричних систем пропуску.

З урахуванням викладених переваг, доцільним є представлення загальної структури процесу ідентифікації особи, що реалізується на основі методу

максимальної ентропії (рис. 3.1). Схема дозволяє наочно відобразити послідовність етапів – від моменту фіксації зображення до прийняття остаточного рішення щодо особи, яка проходить ідентифікацію.

Рисунок 3.1 – Структурна схема методу ідентифікації особи

На схемі зображено основні функціональні етапи процесу біометричної ідентифікації. Початковим елементом є сенсор, який фіксує зображення обличчя користувача та передає його на блок попередньої обробки. В процесі обробки виконується виявлення області обличчя, нормалізація, масштабування та перетворення зображення у вектор ознак. Отриманий вектор надходить до модуля класифікації, який реалізує логістичну регресію з функцією максимальної ентропії. Ймовірності належності до кожного з класів (зареєстрованих користувачів) обчислюються за допомогою softmax-функції, що дозволяє визначити ступінь схожості між вхідним зразком та кожним шаблоном у базі. Далі на основі заданого порогу приймається рішення про відповідність: якщо ймовірність перевищує критичне значення, система підтверджує або відхиляє ідентифікацію. У разі успішного збігу модель повертає результат ідентифікації або верифікації, який

надалі використовується для дозволу або блокування доступу. Такий підхід забезпечує високу швидкість прийняття рішення при збереженні прийнятного рівня точності навіть у ресурсно обмежених умовах.

Для функціонування системи розпізнавання обличчя необхідно реалізувати чітко визначений процес навчання моделі, який включає підготовку, обробку даних та оцінку якості класифікації. Від правильності виконання кожного етапу залежить точність моделі та її здатність до узагальнення. На рис. 3.2 зображено детальний алгоритм навчання моделі розпізнавання обличчя, що відображає послідовність обробки вхідних даних, процес навчання та інтеграцію отриманої моделі у систему.

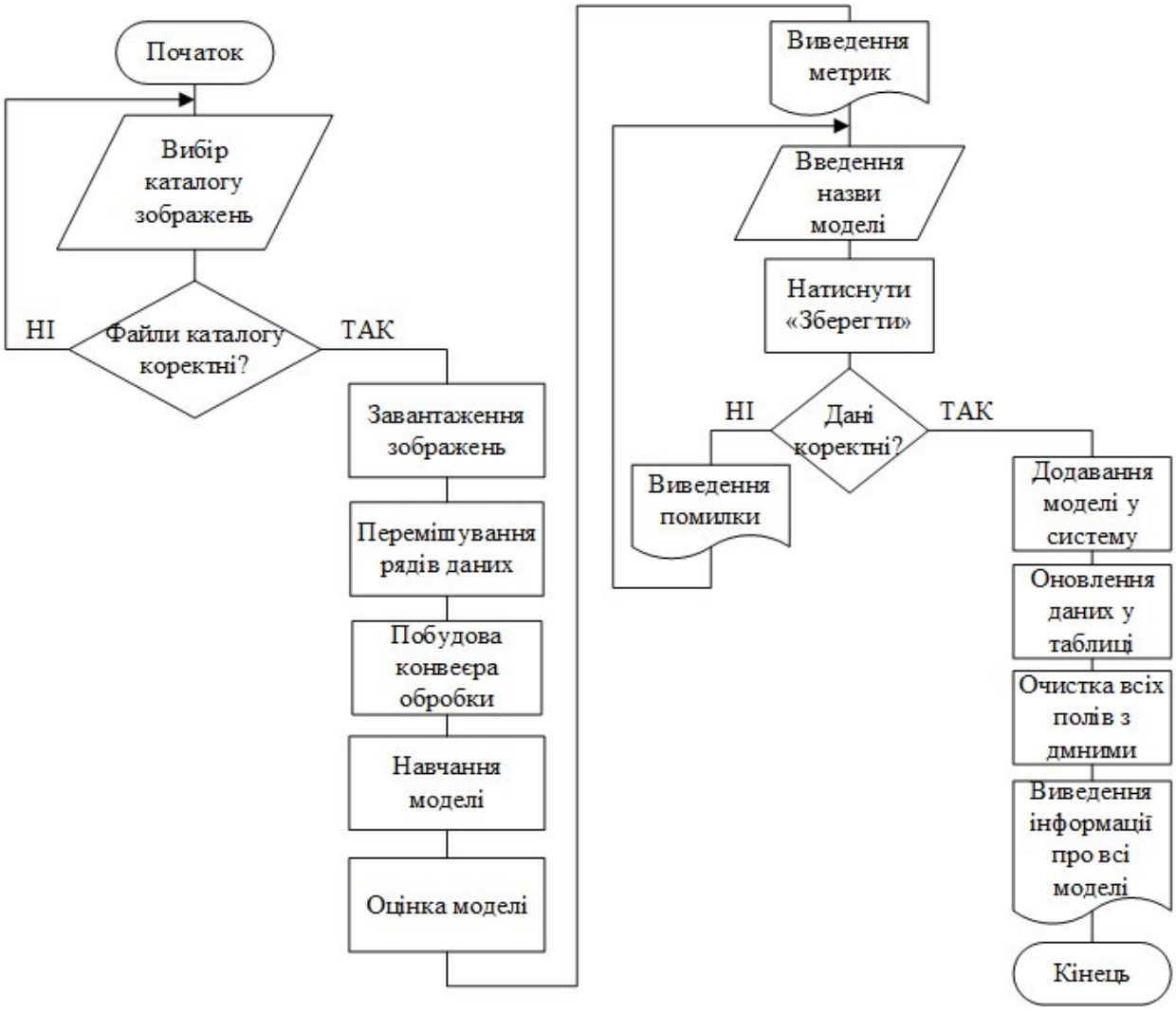


Рисунок 3.2 – Алгоритм навчання моделі для розпізнавання обличчя людини

Навчання починається з вибору каталогу, що містить зображення обличь. На цьому етапі система перевіряє коректність файлів, і якщо структура даних не відповідає заданим вимогам, користувач отримує повідомлення про помилку. Якщо ж перевірка проходить успішно, система завантажує зображення та перемішує ряди даних для уникнення впливу порядку входження прикладів на якість навчання. Далі відбувається побудова конвеєра обробки, який включає нормалізацію піксельних значень, кодування міток класів та формування ознак. Після підготовки даних запускається процес тренування моделі, що виконується за допомогою алгоритму обмеженого спуску за градієнтом. Коли навчання завершено, проводиться оцінка моделі, що включає розрахунок метрик, таких як точність передбачень та функція втрат.

Після оцінки моделі система виводить отримані метрики для аналізу якості розпізнавання. Далі користувач вводить назву моделі та натискає кнопку «Зберегти». Якщо введені дані коректні, модель додається у систему, дані в таблицях оновлюються, а попередні значення очищуються. Також здійснюється виведення оновленої інформації про всі моделі, які є в системі. У разі некоректного введення система повідомляє про помилку та пропонує внести виправлення. Така послідовність дій дозволяє автоматизувати процес навчання та інтеграції моделей розпізнавання обличчя у систему пропуску, забезпечуючи їхню точність і стабільність роботи.

3.2 Метод обробки похибок при розпізнаванні рис обличчя

У біометричній системі на основі аналізу зображення обличчя користувачеві надається дозвіл на вхід або ж відмовляється у доступі. Під час цього процесу можуть виникнути наступні фундаментальні помилки:

- хибний допуск (False Acceptance): система приймає сторонню особу, яка не має прав на доступ;
- хибне відхилення (False Rejection): система відмовляє у доступі легітимному користувачеві, що справді має право доступу.

З точки зору експлуатації, метою є мінімізувати обидва типи помилок. Проте практично існує компроміс: зниження ймовірності хибного допуску зазвичай супроводжується підвищенням імовірності хибного відхилення і навпаки. Тому метод обробки похибок повинен гарантувати збалансовану роботу системи з урахуванням вимог безпеки та зручності.

False Acceptance Rate (FAR) відображає частку випадків, коли система помилково видала дозвіл на вхід стороннім особам порівняно зі всіма відмовами. Формалізовано це подається у вигляді:

$$FAR = \frac{Number\ of\ False\ Acceptances}{Total\ Number\ of\ Rejections}, \quad (3.4)$$

де:

− Number of False Acceptances – кількість помилкових позитивних спрацювань (наприклад, система визнала неавторизовану людину як законного користувача);

− Total Number of Rejections – загальна кількість випадків, коли біометричний алгоритм «відхилив» доступ.

Чим менше це відношення, тим рідше система ризикує пропустити сторонніх, що посилює безпеку.

Зворотньою проблемою є False Reject Rate (FRR), який розкриває, скільки відмов система видала несправедливо щодо власних зареєстрованих користувачів. Алгебраїчно його описують так:

$$FRR = \frac{Number\ of\ False\ Rejections}{Total\ Number\ of\ Acceptances}, \quad (3.5)$$

де:

− Number of False Rejections – показує, скільки разів законні користувачі були «відкинуті» системою;

− Total Number of Acceptances – загальна кількість успішних входів, включно з усіма випадками, де користувач отримав доступ (як справжні, так і випадково пропущені імпостери, але зазвичай акцент роблять на відношенні саме до прийняття законних користувачів).

Показник FRR відображає рівень незручностей і можливих збоїв у роботі системи, спричинених відмовами справжнім власникам доступу. Зниження FRR підвищує довіру користувачів і зменшує ймовірність збоїв у робочому процесі.

Кожна біометрична система задає поріг (τ), згідно з яким визначається прийняття чи відхилення користувача. Знижений поріг сприяє тому, що більше осіб отримає доступ – це зменшує FRR, проте водночас підвищує ризик хибних допусків (FAR зростає). І навпаки, підвищений поріг робить систему вкрай суворою: значно зменшується FAR, але часто страждає FRR.

Тому «метод обробки похибок» передбачає механізми аналізу поточної статистики та адаптивного керування порогом: наприклад, на основі максимізації коректності для певної вибірки або мінімізації сумарної вартості помилок, якщо кожен із типів помилки має різну «ціну».

Пропонований метод обробки похибок включається як складова модуля моніторингу якості в системі пропуску. Алгоритмічно процес працює наступним чином:

− у режимі реального часу контролюються запити на вхід та випадки відмови;

− на основі обчислених показників FAR та FRR проводиться аналіз, чи не виходять вони за допустимі межі;

− у разі потреби здійснюється автоматична чи напівавтоматична перебудова порога та повторна оцінка.

Даний підхід убезпечує від поступового накопичення погрішностей, що можуть спричиняти критичні наслідки для пропускної системи.

Визначені ключові показники похибок: FAR відповідає за рівень захищеності від стороннього проникнення, а FRR характеризує рівень довіри з боку легальних користувачів і загальну зручність використання. Реалізовано підхід до адаптивного

керування, що передбачає гнучке налаштування порога τ, завдяки чому система може динамічно реагувати на поточні умови та підтримувати оптимальний компроміс між безпекою та зручністю. Забезпечується стале функціонування СП шляхом постійного збору статистики, який надає змогу вчасно виявити тенденції до зростання певного типу помилок, вжити коригувальних заходів та оптимізувати біометричну систему загалом.

Таким чином, запропонований метод обробки похибок при розпізнаванні рис обличчя слугує інтегральною частиною програмно-технічного комплексу доступу, утримуючи високий рівень безпеки та водночас гарантує комфорт користувачів завдяки контрольованому показнику хибних відмов.

3.3 Математична модель

Нехай кожне вхідне зображення має роздільну здатність 224×224 і 3 колірні канали (RGB). Тоді, позначивши індекс пікселя за двома координатами (i,j) і каналом c, можна подати зображення у вигляді векторизованого тензора:

$$x = (x_1, x_2, \ldots, x_d) \in R^d \tag{3.6}$$

де d=224×224×3. Елемент x_k відповідає інтенсивності (нормалізованому значенню) каналу c для пікселя з координатами (i,j).

Оскільки вихідні кольорові значення R,G,B звичайно знаходяться в діапазоні [0,255], проводять їх масштабування до [0,1]. Формально для кожного пікселя (до векторизації) можна записати:

$$x_{\text{НОРМ}}(i,j,c) = \frac{x_{\text{ПОЧ}}(i,j,c)}{255}. \tag{3.7}$$

Після цього зображення вважається попередньо підготовленим, а результуючий вектор має вигляд:

$$\hat{x} = (\hat{x}_1, \hat{x}_2, \ldots, \hat{x}_d). \quad (3.8)$$

Припустимо, що є набір $\{(\hat{x}_n, y_n)\}_{n=1}^{N}$, де $\hat{x}_n$ – нормалізований вектор пікселів n-го зображення, а y_n – відповідна мітка класу (наприклад, ідентифікатор особи). Для забезпечення коректної оцінки якості частину даних виділяють у тестову вибірку. Розбиття можна формально подати так:

$$D = D_{train} \cup D_{test},\ D_{train} \cap D_{test} = \varnothing, \quad (3.9)$$

де D_{train} – навчальна множина, а D_{test} – тестова множина (наприклад, у співвідношенні 80% до 20%).

З метою зручності реалізації алгоритмів багатокласової класифікації кожну початкову мітку y перетворюють у ключ-ідентифікатор (LabelAsKey):

$$y \mapsto k_y \in \{1,2, \ldots, K\}, \quad (3.10)$$

де K – загальна кількість різних осіб (класів) у системі, а k_y – ціле число, яке відповідає оригінальній мітці y.

Для навчання моделі використовується вектор $\hat{x}$ без додаткових перетворень або з мінімальними змінами (наприклад, масштабуванням). У найпростішому випадку вектор ознак f є копією нормалізованого вектора пікселів:

$$f = (f_1, f_2, \ldots f_d) = \hat{x}. \quad (3.11)$$

Якщо додатково застосовувати нормалізацію ℓ_2 або MinMax, формально це можна описати як:

$$f_i = \frac{\hat{x}_i - min\ (\hat{x})}{ma(\hat{x}) - min\ (\hat{x})} (\text{для кожного } i). \quad (3.12)$$

Нехай θ є вектором параметрів класифікатора. Тоді імовірність належності зразка f до класу k можна записати як:

$$p(f,\theta) = \frac{exp\left(\theta_k^T f\right)}{\sum_{j=1}^{K} exp\left(\theta_j^T f\right)}, \tag{3.13}$$

де θ_k – підмножина параметрів, що відповідає класу k. У коді цей блок реалізовано через алгоритм максимальної ентропії на основі обмеженого спуску за градієнтом для багатокласової класифікації, який використовує узагальнення моделі логістичної регресії для кількох класів.

Для кожної пари (f_n, k_{y_n}) у навчальній вибірці функція втрат має вигляд:

$$l\left(\theta; f_n, k_{y_n}\right) = -\sum_{k=1}^{K} \left(k = k_{y_n}\right) ln\, ln\, p\,(k|f_n,\theta), \tag{3.14}$$

де $1(\cdot)$ – індикаторна функція (дорівнює 1, якщо $= k_{y_n}$, і 0 – в іншому випадку).

Наступним кроком регуляризація додається до функції втрат за формулою:

$$R(\theta) = a\sum_{k-1}^{K} \|\theta_k\|_1 + \beta\sum_{k-1}^{K} \|\theta_k\|_2^2, \tag{3.15}$$

де $\|\cdot\|_1$ – норма L_1, $\|\cdot\|_2^2$ – норма L_2 у квадраті, а α і β – вагові коефіцієнти регуляризації (α=0.1, β=0.3).

Загальна цільова функція з урахуванням регуляризації (сума втрат для всієї навчальної множини) приймає вигляд:

$$L(\theta) = \frac{1}{N}\sum_{n=1}^{N} l\left(\theta, f_n, k_{y_n}\right) + R(\theta). \tag{3.16}$$

У процесі навчання знаходимо:

$$\theta^* = argL(\theta)\ . \tag{3.17}$$

Тренер тренування моделі застосовує квазі-ньютонівський метод (L-BFGS), що послідовно оновлює оцінки θ на підставі градієнтів функції $L(\theta)$. Формально ітераційний процес можна подати як:

$$\theta^{(t+1)} = \theta^{(t)} - \eta^{(t)} B_t^{-1} \nabla_\theta L\left(\theta^{(t)}\right), \tag{3.18}$$

де B_t – наближена матриця Гессіана (оновлювана на кожній ітерації), а $\eta^{(t)}$ – крок (learning rate).

Отримавши оптимальні параметри θ^*, для нового вектора ознак f_{new} вираховуємо імовірності:

$$p\left(f_{new}, \theta^*\right) = \frac{exp\left(\left(\theta_k^*\right)^T f_{new}\right)}{\sum_{j=1}^{K} exp\left(\left(\theta_j^*\right)^T f_{new}\right)}. \tag{3.19}$$

Для прийняття рішення обираємо клас $\hat{k}$ із максимальною імовірністю:

$$\hat{k} = argp\left(f_{new}, \theta^*\right)\ . \tag{3.20}$$

А числові значення *Score* відповідають вектору імовірностей $[p\left(f_{new}, \theta^*\right), \ldots, p\left(f_{new}, \theta^*\right)]$.

Для оцінювання ефективності навчених моделей розпізнавання зображень обличчя використовуються метрики: *MicroAccuracy*, *MacroAccuracy*, *LogLoss*. Ці метрики обчислюються після завершення тренування та дають кількісну оцінку якості прийняття рішень (класифікації).

Метрика *MicroAccuracy* (мікро-точність) оцінює загальну частку правильно класифікованих прикладів з усієї вибірки без розрізнення на класи. У такий спосіб

кожен випадок має однакову вагу у формуванні кінцевого відсотка коректних прогнозів.

Нехай:

− TP_i – кількість прикладів, які фактично належать класу *i* і були правильно віднесені до класу *i* (*True Positive*);

− FP_i – кількість прикладів, що належать іншим класам, але помилково зараховані до класу ii (*False Positive*);

− FN_i – кількість прикладів, що належать класу ii, проте були помилково віднесені до інших (*False Negative*).

Якщо в системі *C* класів, для мікро-підходу підсумовуємо ці величини по всім класам:

$$TP_{(all)} = \sum_{i=1}^{C} TP_i, FP_{(all)} = \sum_{i=1}^{C} FP_i, FN_{(all)} = \sum_{i=1}^{C} FN_i. \quad (3.21)$$

Тоді *MicroAccuracy* має вигляд:

$$MicroAccuracy = \frac{TP_{(all)}}{TP_{(all)} + FP_{(all)}}, \quad (3.22)$$

що еквівалентно відношенню кількості правильно класифікованих прикладів до загальної кількості передбачень. Таким чином, для мікро-точності головне – урахувати всі зразки як «однорідний набір», без зосередження на конкретних класах.

На відміну від мікро-підходу, який агрегує всю інформацію, *MacroAccuracy* (макро-точність) ґрунтується на почерговому обчисленні точності для кожного класу з подальшим усередненням. Це дозволяє однаково зважувати кожен клас, навіть якщо їхні розміри нерівномірні (наприклад, один клас має значно більше прикладів, ніж інший).

Для класу і точність ($Accuracy_i$) визначається через суму правильних спрацювань (TP) та правильних відхилень (TN) від загальної кількості прикладів, деяким чином пов'язаних із цим класом:

$$Accuracy_i = \frac{TP_i+TN_i}{TP_i+TN_i+FP_i+FN_i}, \quad (3.23)$$

де, TN_i – кількість прикладів, що не належать класу i і не були до нього віднесені (тобто система не помилилась з відсутністю приналежності до i).

Загальна метрика *MacroAccuracy* обчислюється як середнє арифметичне по всіх C класах:

$$MacroAccuracy = \frac{1}{C}\sum_{i=1}^{C} Accuracy_i. \quad (3.24)$$

Оскільки кожен клас робить однаковий внесок у формування фінального значення, MacroAccuracy стає особливо корисною, коли структура вибірки є незбалансованою (класів із різною кількістю прикладів).

Логарифмічна втрата (*LogLoss*) оцінює не лише те, чи правильно система класифікувала приклад, а й те, наскільки «впевнено» вона це зробила. Ця метрика вимагає, аби модель повертала повноцінний вектор імовірностей належності до кожного з C можливих класів.

Нехай $p_{i,n}$ – імовірність, із якою модель «вважає», що приклад із номером n належить класу i. У задачі багатокласової класифікації виконується:

$$\sum_{i=1}^{N} p_{i,n} = 1, \quad (3.25)$$

а для коректної оцінки потрібні цільові змінні $y_{i,n}$, що дорівнюють 1, якщо приклад n справді належить класу i, і 0 – в іншому випадку.

Загальний вираз *LogLoss* для N прикладів можна подати так:

$$LogLoss = \frac{1}{N}\sum_{n=1}^{N} \sum_{i=1}^{C} y_{i,n} \cdot ln\, ln\,(p_{i,n}). \quad (3.26)$$

Якщо модель приписує високі імовірності справжнім класам, значення $ln\, ln\,(p_{i,n})$ буде від'ємним, але близьким до нуля, отже внесок до суми втрат стане невеликим. Натомість, якщо модель «помиляється» і ставить неадекватно низьку імовірність правильному класу, $ln\, ln\,(p_{i,n})$ набуде великих від'ємних значень, унаслідок чого *LogLoss* суттєво зростає.

Використання *MicroAccuracy*, *MacroAccuracy* та *LogLoss* у комплексі дає змогу всебічно проаналізувати поведінку алгоритму розпізнавання: від суто правильно-неправильної оцінки до збалансованості по класах і, зрештою, розгляду рівня впевненості в прогнозах. Це забезпечує повну характеристику якості системи, що вкрай важливо для надійного використання біометричних даних у реальних умовах.

Застосування даної моделі дає змогу розпізнавати користувачів за їхніми біометричними даними та формувати показник упевненості, що використовується в системі контролю доступу.

3.4 Висновки

У рамках даного розділу проведено аналіз методу та алгоритму навчання моделі для розпізнавання обличчя людини, що використовується у системах контролю доступу. Обґрунтовано вибір методу максимальної ентропії, який реалізує логістичну регресію для багатокласової класифікації та оптимізується за допомогою алгоритму L-BFGS.

Окрему увагу приділено методу обробки похибок, який використовується для підвищення точності розпізнавання. Проаналізовано математичний апарат для оцінки FAR та FRR, що дозволяє зменшити ймовірність помилкового допуску або відхилення коректних користувачів. Запропоновані підходи до оптимізації системи

дозволяють досягти збалансованої роботи моделі, що важливо для реального використання в системах безпеки.

Було розроблено математичну модель, яка визначає основні формальні залежності та етапи тренування системи розпізнавання обличчя. Вона враховує особливості формування простору ознак, оптимізацію параметрів навчання, обробку вхідних даних та їх класифікацію. Запропонована модель є основою для реалізації алгоритмів у програмному забезпеченні СП та дозволяє забезпечити точну роботу системи незалежно від змін зовнішніх умов.

Результати цього розділу створюють базу для наступного етапу дослідження, де буде здійснено практичну реалізацію розробленого програмно-технічного засобу. Запропоновані методи, алгоритми та математична модель дозволять провести експериментальну перевірку роботи системи та оцінити її ефективність у реальних умовах.

4 РЕЗУЛЬТАТИ РОБОТИ ПРОГРАМНО-ТЕХНІЧНОГО ЗАСОБУ ДЛЯ СИСТЕМИ ПРОПУСКУ НА ОСНОВІ БІОМЕТРИЧНИХ ДАНИХ

4.1 Проектування архітектури системи пропуску на основі біометричних даних

У сучасних системах контролю доступу на основі біометричних даних ключовим аспектом є архітектура програмного забезпечення, що забезпечує стабільну роботу, масштабованість та безпеку обробки даних. Архітектурне рішення впливає на швидкість обробки запитів, можливість інтеграції з іншими інформаційними системами та адаптацію до зростаючих навантажень. Враховуючи специфіку роботи біометричних систем, що включає зчитування, обробку та порівняння біометричних шаблонів, було обрано трьохрівневу архітектуру для побудови СП.

Трьохрівнева архітектура дозволяє чітко розподілити функціональні компоненти системи між окремими рівнями: клієнтським, серверним (прикладним) та рівнем бази даних. Такий підхід забезпечує гнучкість, надійність та безпеку, що є критично важливими факторами для біометричних систем [82]. Клієнтський рівень відповідає за взаємодію з користувачем, включаючи сканування обличчя та подачу запитів до сервера. Серверний рівень виконує обробку запитів, включаючи попередню обробку зображень, вилучення ознак та порівняння їх із наявними шаблонами. Рівень бази даних зберігає біометричні шаблони та записи про користувачів, забезпечуючи швидкий доступ до ідентифікаційної інформації.

Обґрунтування вибору трьохрівневої архітектури базується на її перевагах у масштабованості, продуктивності та безпеці. Завдяки розподілу обчислювального навантаження між рівнями система здатна ефективно обробляти великий потік запитів без перевантаження окремих її компонентів. Крім того, така структура дозволяє гнучко адаптувати систему до змін, наприклад, додавати нові методи біометричної автентифікації або інтегрувати систему з іншими службами безпеки. Окреме збереження біометричних шаблонів на рівні бази даних підвищує захищеність даних, оскільки доступ до них обмежується лише серверною частиною, що мінімізує ризики компрометації інформації.

На початковому етапі розроблено структуровану модель рівня даних. На рис. 4.1 зображено діаграму класів цього рівня, яка демонструє структуру класів, їх властивості та методи, що використовуються для управління інформацією у системі.

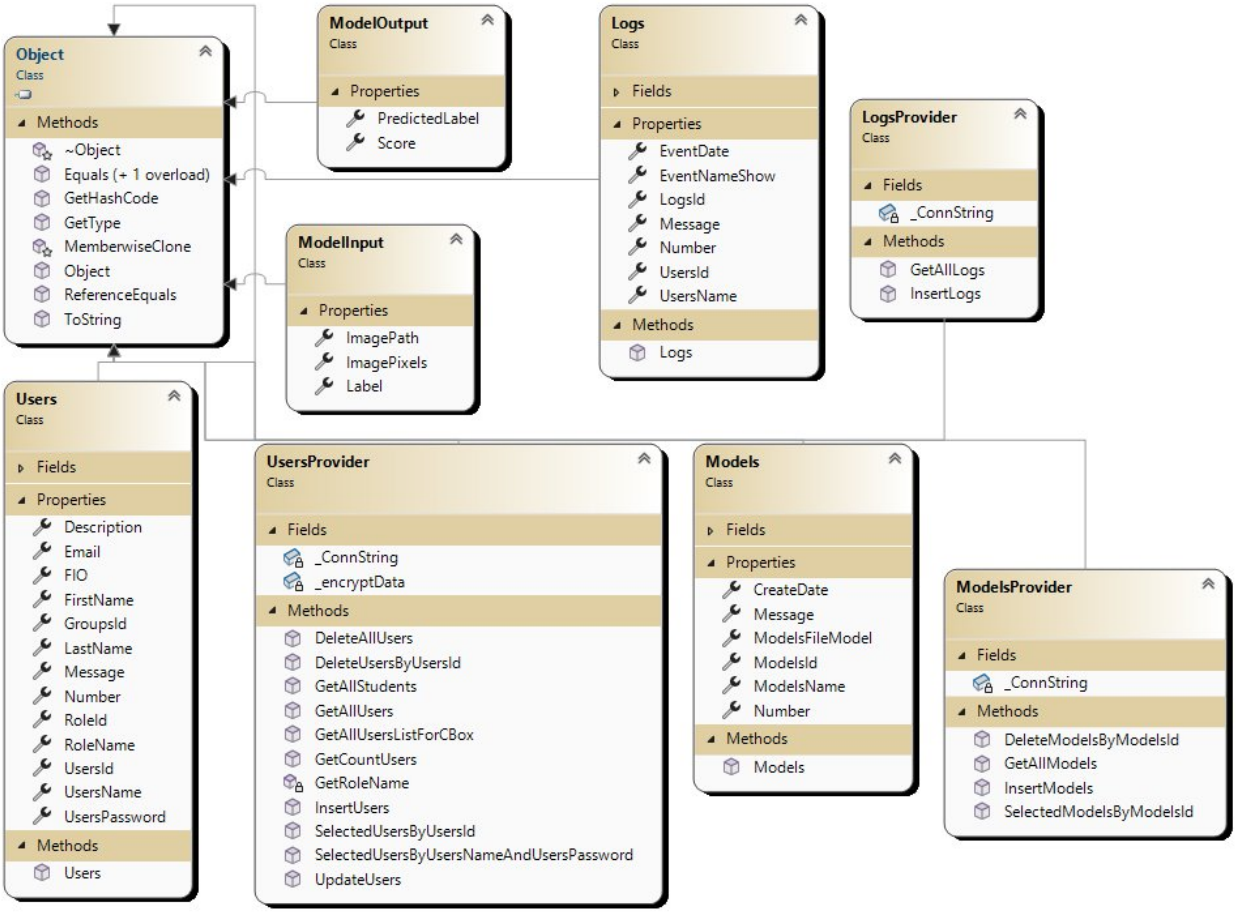


Рисунок 4.1 – Діаграма класів рівня даних системи

Діаграма класів рівня даних складається з:

– класу Users, який містить інформацію про користувачів системи, включаючи їхні імена, прізвища, логіни, паролі, рівень доступу та супутні атрибути. Він є основним класом для управління обліковими записами користувачів та взаємодіє з класом UsersProvider, який забезпечує методи для виконання CRUD-операцій (створення, читання, оновлення та видалення даних про користувачів);

− класу Models, що зберігає відомості про завантажені та використані моделі машинного навчання для розпізнавання облич. Він містить властивості, що визначають назву моделі, дату її створення та унікальні ідентифікатори. Доступ до даних цього класу забезпечується через клас ModelsProvider, який містить методи для отримання, оновлення та видалення моделей;

− класу Logs, який використовується для збереження інформації про події у системі, зокрема дати входу користувачів, результатів розпізнавання облич та інших важливих операцій. Він містить атрибути, що дозволяють відстежувати історію дій у системі, а його управління здійснюється через клас LogsProvider, що включає методи отримання та додавання нових записів;

− класу ModelInput, який відповідає за структуру вхідних даних для моделі машинного навчання. Він містить властивості, що визначають шлях до зображення, вектор ознак та мітку класу;

− класу ModelOutput, що містить результати роботи моделі, включаючи передбачений клас (PredictedLabel) та оцінку достовірності передбачення (Score).

Розроблена структура рівня даних дозволяє ефективно керувати користувачами, моделями розпізнавання та історією подій, забезпечуючи масштабованість, безпеку та узгодженість даних. Завдяки чітко визначеним методам взаємодії між класами система може легко інтегрувати нові функціональні можливості та підтримувати актуальність збереженої інформації.

Для забезпечення взаємодії користувачів із системою пропуску на основі біометричних даних розроблено структуровану модель рівня користувацького інтерфейсу. Вона включає набір форм, що відповідають за обробку різних функціональних процесів, таких як автентифікація користувачів, управління моделями розпізнавання, перегляд системних логів та персоналізація налаштувань. Завдяки такій структурі система отримує інтуїтивно зрозумілий та зручний інтерфейс, що дозволяє операторам швидко та ефективно керувати доступом.

На рис. 4.2 зображено діаграму класів рівня користувацького інтерфейсу, яка демонструє архітектуру GUI, її основні компоненти та методи, що забезпечують роботу інтерфейсу.

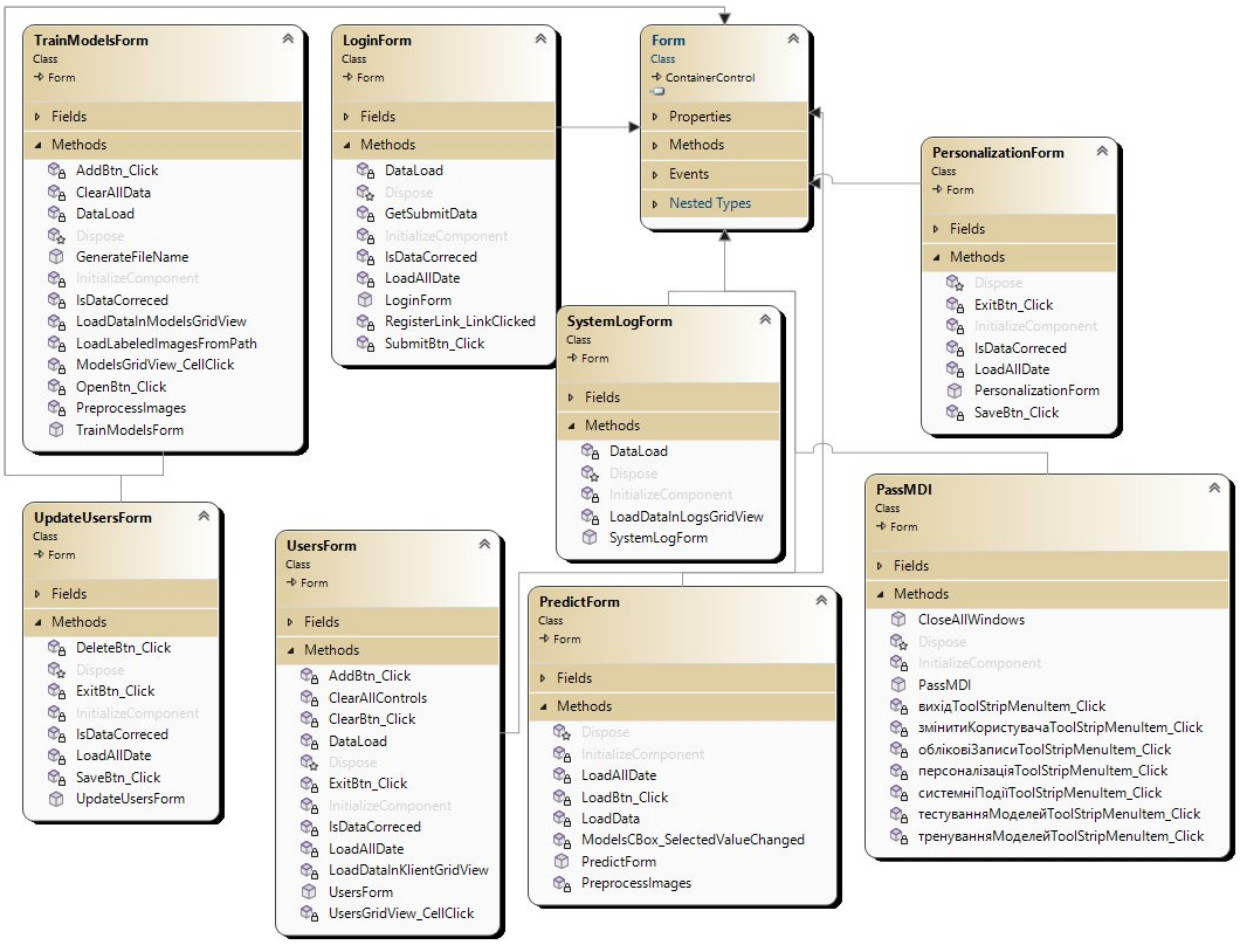


Рисунок 4.2 – Діаграма класів рівня користувацького інтерфейсу

Діаграма класів рівня користувацького інтерфейсу складається з:

– класу LoginForm, який відповідає за автентифікацію користувачів у системі. Він містить методи для завантаження даних, перевірки коректності введеної інформації та обробки натискання кнопки входу. Також реалізована можливість відновлення доступу через реєстраційну систему;

– класу PassMDI, що забезпечує головне меню системи, з якого користувач може переходити до різних модулів, таких як тестування моделей, системні логи, управління користувачами та персоналізація. Він містить методи для обробки взаємодії з меню та закриття всіх відкритих вікон системи;

– класу TrainModelsForm, який відповідає за навчання моделей розпізнавання облич. У ньому реалізовані методи для завантаження даних,

попередньої обробки зображень, запуску процесу тренування та перевірки коректності введених параметрів;

− класу PredictForm, що забезпечує процес тестування моделей розпізнавання. Він містить методи для вибору моделі, виконання розпізнавання та відображення отриманих результатів;

− класу UsersForm, який дозволяє керувати обліковими записами користувачів. Він містить методи для додавання, редагування, видалення та перегляду списку користувачів. Інформація завантажується у вигляді таблиці з можливістю фільтрації та сортування;

− класу UpdateUsersForm, що використовується для редагування інформації про користувачів. Він містить методи для завантаження даних, внесення змін та перевірки коректності введених значень;

− класу SystemLogForm, який дозволяє переглядати історію подій у системі. Він містить методи для завантаження логів, відображення деталей кожної події та виконання пошуку за ключовими параметрами;

− класу PersonalizationForm, що відповідає за налаштування інтерфейсу користувача. У ньому реалізовані методи для збереження змін у конфігурації, коригування параметрів безпеки та управління персональними налаштуваннями.

Розроблена архітектура користувацького рівня дозволяє ефективно управляти всіма аспектами роботи СП. Чітке розмежування функцій між класами забезпечує гнучкість, масштабованість та зручність у користуванні, що є важливим критерієм для впровадження біометричних систем безпеки.

У процесі розробки СП на основі біометричних даних важливу роль відіграє рівень бізнес-логіки, який забезпечує обробку введених користувачем даних, управління правами доступу, перевірку коректності інформації та шифрування критично важливих даних. Завдяки чітко визначеній архітектурі, система може функціонувати ефективно та безпечно, дотримуючись принципів масштабованості та захищеності інформації.

На рис. 4.3 зображено діаграму класів рівня бізнес-логіки системи, яка містить класи, що відповідають за перевірку, шифрування, управління ролями та інші ключові функції.

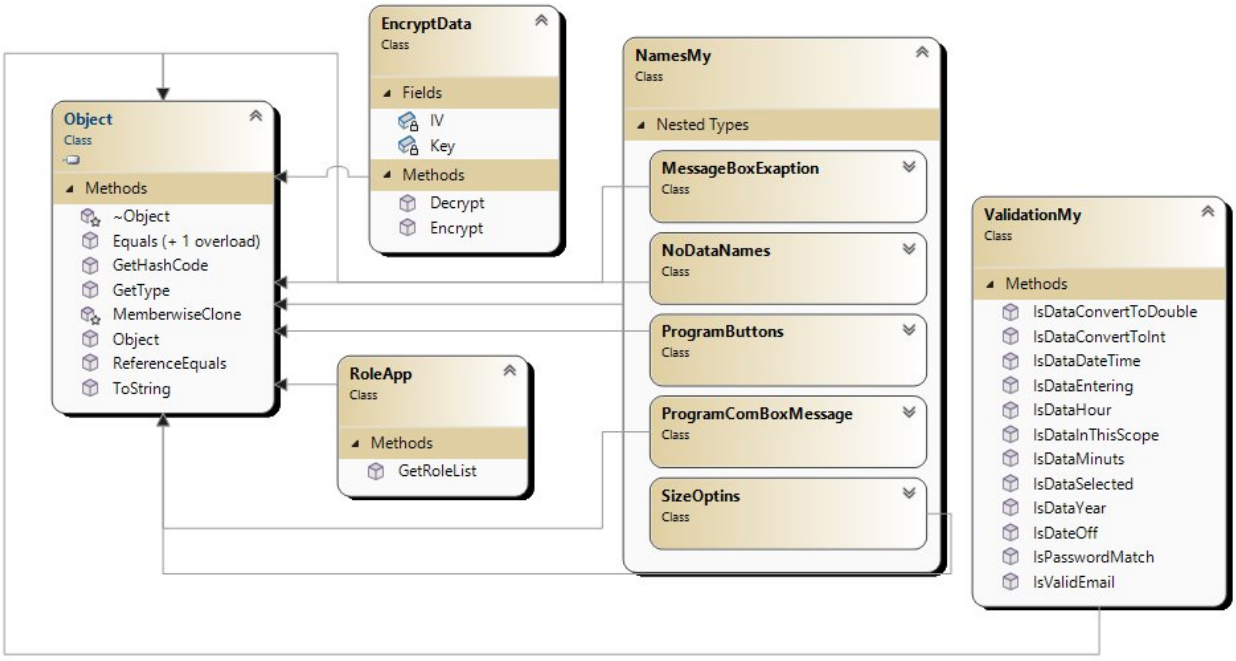


Рисунок 4.3 – Діаграма класів рівня бізнес-логіки

Діаграма класів рівня бізнес-логіки складається з:

− класу EncryptData, який реалізує механізми шифрування та розшифрування конфіденційних даних. Він містить поля IV (ініціалізаційний вектор) та Key (ключ шифрування), а також методи Encrypt та Decrypt, які використовуються для забезпечення безпеки збережених та переданих даних у системі;

− класу ValidationMy, що відповідає за перевірку коректності введених користувачем даних. Він містить методи для перевірки типів даних (IsDataConvertToDouble, IsDataConvertToInt, IsDataDateTime), перевірки правильності введених дат (IsDataHour, IsDataMinutes, IsDataYear) та валідації введених паролів (IsPasswordMatch) і електронних адрес (IsValidEmail);

− класу RoleApp, який використовується для управління ролями користувачів у системі. Він містить метод GetRoleList, що дозволяє отримати список доступних ролей та їхніх привілеїв у межах роботи системи;

− класу NamesMy, який містить вкладені класи для управління повідомленнями, кнопками інтерфейсу та іншими елементами системи. Вкладені класи включають MessageBoxExaption (робота із винятковими повідомленнями), NoDataNames (робота із порожніми даними), ProgramButtons (управління кнопками), ProgramComBoxMessage (робота з комбінованими повідомленнями), SizeOptins (керування розмірами елементів).

Розроблена структура рівня бізнес-логіки дозволяє ефективно обробляти вхідні дані, контролювати доступ до критично важливих функцій, перевіряти валідність інформації та захищати конфіденційні дані від несанкціонованого доступу. Така архітектура забезпечує стабільність, безпеку та гнучкість системи, що є ключовими вимогами для реалізації біометричного контролю доступу.

Для успішної ідентифікації осіб у системі пропуску було реалізовано алгоритм навчання моделі розпізнавання обличчя, який використовує методи машинного навчання для аналізу та класифікації зображень. Основний підхід базується на попередній обробці даних, формуванні ознак та використанні алгоритму максимальної ентропії на основі обмеженого спуску за градієнтом. У ході навчання система аналізує велику кількість зображень облич, що належать різним особам, і формує математичне представлення унікальних рис кожного користувача. Це дозволяє досягти високої точності розпізнавання навіть за змінних умов освітлення, положення голови чи інших факторів. Для підвищення ефективності було оптимізовано параметри регуляризації та впроваджено додаткові етапи нормалізації даних.

Класи ModelInput та ModelOutput, які використовуються для представлення вхідних та вихідних даних у процесі навчання та розпізнавання можна побачити у додатку А.

Клас ModelInput відповідає за збереження інформації про вхідні дані для моделі. Він містить шлях до зображення, що використовується у навчанні, та мітку,

яка визначає відповідний клас особи. Крім того, у цьому класі передбачено масив числових значень, що представляє пікселі зображення після його перетворення у відповідний формат. Кожен елемент масиву відповідає інтенсивності кольорових каналів (Red, Green, Blue), що дозволяє забезпечити коректне представлення вхідних зображень у вигляді числового вектора.

Клас ModelOutput призначений для збереження результатів розпізнавання. У ньому міститься передбачена мітка, що відповідає ідентифікованій особі, а також масив значень, що містить ймовірності належності до різних класів. Ці значення дозволяють оцінити впевненість моделі у своєму прогнозі, що є важливим аспектом для аналізу точності розпізнавання. Результати класифікації можуть бути використані для прийняття рішення щодо надання або обмеження доступу в системі пропуску.

У процесі взаємодії з користувачем передбачено можливість вибору папки для завантаження зображень, що використовуватимуться у системі (рис. 4.4). При натисканні кнопки відкривається діалогове вікно, яке дозволяє користувачеві вибрати каталог зі зображеннями. Якщо вибір підтверджено, то система очищує попередні дані у відповідному текстовому полі, щоб забезпечити коректне відображення нової інформації. Обраний шлях до каталогу зберігається у відповідному текстовому полі інтерфейсу, що дозволяє користувачеві переглядати активний каталог. Реалізація асинхронного виклику забезпечує зручну роботу з інтерфейсом, не блокуючи основний потік виконання програми, що сприяє плавній роботі системи.

У реалізованій функції передбачено запуск процесу тренування моделі у фоновому режимі, що дозволяє уникнути блокування основного потоку програми та забезпечити плавну роботу користувацького інтерфейсу

Основним кроком є ініціалізація MLContext, що створює середовище для машинного навчання, необхідне для налаштування та тренування моделі. Після цього в інтерфейсі користувача виводиться повідомлення про успішне створення контексту навчання, що допомагає у відстеженні поточних етапів обробки даних. Використання методу Invoke гарантує, що оновлення інтерфейсу відбувається

коректно, оскільки робота з елементами управління здійснюється у відповідному потоці. Такий підхід забезпечує стабільне виконання задач машинного навчання, не перериваючи взаємодію користувача із системою.

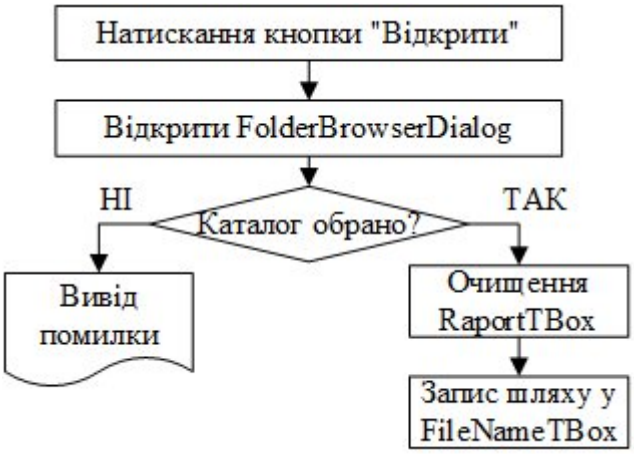


Рисунок 4.4 – Вибір папки для завантаження зображень

У процесі налаштування тренування моделі ключовим кроком є збереження шляху до папки, яка містить зображення для навчання (рис. 4.5).

```
1. Відкрити діалог вибору папки
2. Якщо користувач підтвердив вибір:
   2.1. Зберегти обраний шлях до змінної imagePath
   2.2. Виконати дію в основному потоці:
        - Додати текст до текстового поля звіту:
          "Обрано каталог зі зображеннями: {imagePath}"
```

(1. Відкрити діалог вибору папки
2. Якщо користувач підтвердив вибір:
2.1. Зберегти обраний шлях до змінної imagePath
2.2. Виконати дію в основному потоці:
- Додати текст до текстового поля звіту:
"Обрано каталог зі зображеннями: {imagePath}")

Рисунок 4.5 – Зберігання шляху до обраної папки

Після вибору каталогу його шлях передається у змінну, що забезпечує доступ до всіх файлів у цій директорії під час подальшої обробки.

Для зручності користувача інформація про вибраний каталог виводиться в інтерфейс у текстовому полі звіту, що дозволяє слідкувати за етапами виконання.

Оскільки оновлення інтерфейсу має виконуватися в головному потоці програми, використовується механізм Invoke, що гарантує коректне оновлення графічного інтерфейсу без порушення роботи асинхронного процесу навчання. Це дозволяє програмі залишатися чутливою до дій користувача, не перериваючи виконання основних алгоритмів.

На наступному етапі відбувається завантаження вихідних даних, які містять інформацію про зображення, включаючи їхні шляхи та відповідні мітки (код представлено у додатку А).

Для цього використовується спеціальний метод, який обробляє вибраний каталог і формує набір даних, що необхідний для подальшого навчання моделі. Після успішного завантаження інформація про виконання цього етапу виводиться у текстове поле звіту, що дає можливість користувачеві слідкувати за процесом. Щоб коректно оновлювати графічний інтерфейс у багатопотоковому режимі, використовується механізм Invoke, який забезпечує безперебійну взаємодію між фоновими обчисленнями та елементами інтерфейсу. Завдяки цьому користувач отримує зворотний зв’язок щодо поточного стану виконання програми без втрати її чутливості до подальших дій.

Також проводиться обробка зображень, яка включає їх перетворення у числовий формат, необхідний для подальшого навчання моделі представлений у додатку А. Кожне зображення приводиться до стандартного розміру 224×224 пікселів і конвертується у тензор, що містить тривимірний масив числових значень, що представляють кольорові канали RGB. Це забезпечує уніфікованість вхідних даних і дозволяє моделі ефективно навчатися, використовуючи однакові параметри для всіх зображень. Після завершення цього процесу система генерує повідомлення про успішне перетворення, яке додається до звіту. Використання Invoke дозволяє оновити графічний інтерфейс у безпечному режимі, зберігаючи стабільність роботи програми під час асинхронного виконання операцій.

Дані зображень конвертуються у формат IDataView, який є основною структурою для зберігання та обробки даних у середовищі ML.NET (додаток А).

Завдяки цьому формується єдиний набір даних, який може використовуватися для подальшого навчання та тестування моделі. Конверсія дозволяє ефективно організувати обробку зображень, оптимізуючи їх для використання у конвеєрі машинного навчання. Після створення IDataView система генерує повідомлення про успішне завершення цього етапу, яке додається у текстове поле звіту. Для забезпечення коректного оновлення інтерфейсу під час асинхронного виконання використовується механізм Invoke, що дозволяє інтегрувати повідомлення у графічний інтерфейс без порушення роботи основного потоку.

Представлені у форматі IDataView, розподіляються на тренувальну та тестову вибірки для подальшого навчання та оцінювання моделі. Для цього використовується функція TrainTestSplit, яка розділяє вхідний набір даних, залишаючи 80% для навчання моделі та 20% для тестування її точності.

Встановлення початкового значення seed забезпечує відтворюваність розподілу даних при повторних запусках, що важливо для стабільного порівняння результатів. Після успішного поділу система додає у звіт повідомлення про завершення цього етапу. Завдяки використанню Invoke, графічний інтерфейс оновлюється коректно, забезпечуючи безперебійну взаємодію користувача з програмою під час виконання навчального процесу.

Після цього будується конвеєр обробки даних (pipeline), який містить послідовність трансформацій та тренувальний алгоритм для навчання моделі. Спочатку виконується кодування міток класів, що дозволяє використовувати їх у процесі машинного навчання. Далі застосовується нормалізація MinMax, що приводить значення пікселів зображень до єдиного діапазону, що покращує стабільність та швидкість навчання. Після цього дані підготовлюються до тренування, копіюючи значення нормалізованих ознак у колонку Features, яка слугуватиме вхідними даними для класифікатора.

Як основний алгоритм навчання використовується LbfgsMaximumEntropy, який реалізує багатокласову логістичну регресію з регуляризацією, що допомагає запобігти перенавчанню моделі. Встановлюються параметри L1- та L2-регуляризації, які контролюють узагальнюючу здатність моделі, а також максимальна кількість ітерацій та розмір історії, що визначає стабільність та ефективність оптимізації. Завершальним етапом є зворотне перетворення прогнозованих значень у вихідні мітки, що робить результати більш інтерпретованими для користувача. Побудований конвеєр забезпечує ефективне навчання та підготовку моделі до прогнозування на основі оброблених даних.

На наступному етапі відбувається навчання моделі на основі підготовлених тренувальних даних.

Запускається таймер для вимірювання тривалості процесу навчання, що дозволяє оцінити продуктивність алгоритму та ефективність використання ресурсів. Конвеєр обробки даних, що містить попередньо налаштовані трансформації та класифікаційний алгоритм, застосовується до тренувальної вибірки, після чого отримується готова модель, яка зможе виконувати прогнозування. По завершенню процесу таймер зупиняється, а отриманий час навчання виводиться у текстове поле звіту, що дозволяє користувачеві контролювати тривалість цього етапу. Оновлення інтерфейсу виконується через Invoke, що забезпечує коректне відображення результатів у головному потоці програми без блокування інших процесів. Після навчання виконується оцінка продуктивності навченого класифікатора.

На початку модель застосовується до тестових даних, що дозволяє отримати передбачені значення для кожного зображення у вибірці. Потім проводиться аналіз результатів класифікації, використовуючи вбудовані засоби оцінки якості у ML.NET. Метод Evaluate розраховує основні метрики, такі як точність класифікації, рівень правильних передбачень та загальну ефективність моделі. Порівнюючи отримані передбачення з фактичними мітками класів, система визначає рівень відповідності моделі тестовим даним, що дозволяє оцінити її здатність до узагальнення та ефективність у реальних сценаріях використання.

Після завершення оцінювання моделі розраховані метрики точності виводяться у графічний інтерфейс для аналізу.

Значення MicroAccuracy показує загальну точність моделі, враховуючи всі правильно класифіковані приклади без розрізнення класів. MacroAccuracy обчислює середню точність для кожного класу окремо, що дозволяє оцінити баланс моделі при класифікації різних категорій. LogLoss характеризує невизначеність передбачень, відображаючи, наскільки модель впевнена у своїх прогнозах, причому менше значення свідчить про кращу якість класифікації. Використання Invoke гарантує, що оновлення текстового поля звіту виконується у головному потоці програми, що забезпечує коректне відображення результатів без порушення роботи інтерфейсу.

Алгоритм зберігання моделі машинного навчання в рамках розробленого застосунку є завершальним етапом навчального процесу, що реалізується після успішного створення та валідації моделі. Його структурну реалізацію наведено на рис. 4.6, де відображено послідовність дій, спрямованих на коректне збереження артефактів моделі та фіксацію факту виконаної операції у системі.

Рисунок 4.6 – Алгоритм зберігання моделі

На початку здійснюється перевірка правильності введених користувачем даних, що гарантує достовірність параметрів, пов’язаних із майбутньою моделлю. Після цього формується шлях до файлу збереження, який зазвичай містить унікальне ім’я для уникнення конфліктів та збереження історії. Далі визначається локальний каталог проєкту – тобто абсолютний шлях до директорії, у якій буде фізично збережено модель. Після підготовки усіх шляхових параметрів здійснюється вставка відповідного запису до бази даних, що забезпечує ідентифікацію та подальше керування моделлю в рамках системи.

Наступним кроком є безпосереднє збереження навченої моделі машинного навчання за допомогою засобів ML.NET. В результаті формується архівований файл моделі, готовий до використання для подальшого прогнозування. Після збереження здійснюється очищення полів введення на формі, що дозволяє користувачу почати нову сесію без залишкових даних. Завершується алгоритм реєстрацією події у системному журналі для забезпечення прозорості дій користувача та відстежуваності історії виконаних операцій. Останнім етапом є виведення повідомлення, яке інформує користувача про успішне завершення процесу збереження моделі. Така послідовність дій забезпечує як функціональну надійність, так і зручність взаємодії з користувачем у рамках реалізованої інформаційної системи.

Алгоритм вибору моделі, відображений на рис. 4.7, описує послідовність дій, що виконуються у системі після зміни вибору у комбінованому списку моделей користувачем. Цей процес є критично важливим для забезпечення коректного завантаження навченої моделі машинного навчання з метою подальшого прогнозування.

Першим етапом є перевірка стану завантаження тем, що виступає механізмом захисту від передчасного спрацювання події зміни вибору у випадках, коли список ще не було ініціалізовано повністю. У разі позитивного результату виконується отримання обраної моделі за її ідентифікатором – значенням, що зчитується з елемента інтерфейсу користувача. На наступному кроці система здійснює

перевірку валідності вибору, тобто підтверджує, що користувач дійсно обрав коректну модель (ідентифікатор не є нульовим або помилковим).

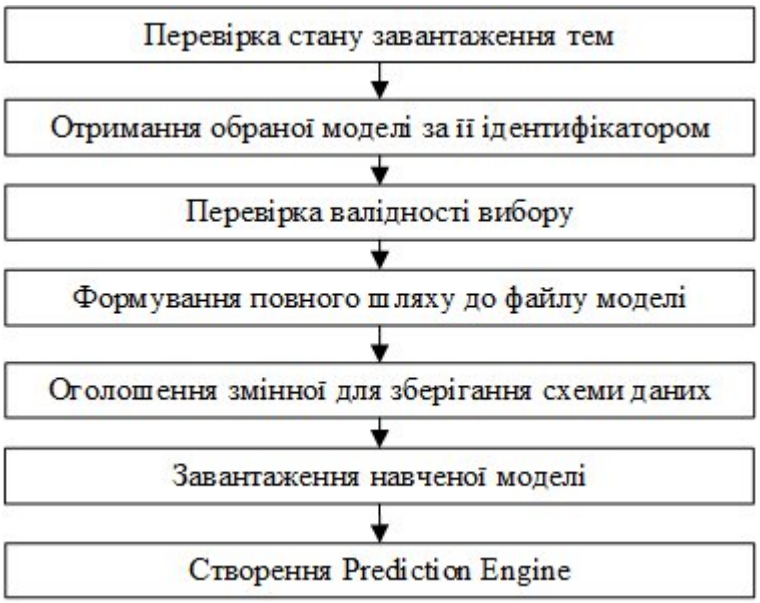


Рисунок 4.7 – Алгоритм вибору моделі

У разі успішного проходження попередніх перевірок формується повний шлях до файлу моделі. Цей шлях складається з кореневого каталогу проєкту та відносного шляху, збереженого у базі даних разом із моделлю. Після цього оголошується змінна для збереження схеми даних – структури, що визначає формальні характеристики вхідного набору, на якому була навчена модель. Це дозволяє коректно інтерпретувати вхідні значення при виконанні прогнозу.

Далі виконується завантаження навченої моделі з файлу, яка відновлюється у вигляді трансформера (ITransformer) за допомогою засобів ML.NET. Завершується алгоритм створенням об’єкта типу Prediction Engine, який надає інтерфейс для прогнозування нових даних у режимі реального часу. Таким чином, кожна зміна вибору моделі активує повноцінний процес підготовки системи до подальшої аналітики на основі раніше збереженої моделі машинного навчання.

Алгоритм підготовки зображень перед передачею у модель, представлений на рис. 4.8, реалізує повноцінний цикл перетворення вхідних графічних даних у числову форму, придатну для подачі на вхід моделі машинного навчання. Цей

процес є критично важливим при роботі з моделями, орієнтованими на обробку зображень, зокрема згортковими нейронними мережами.

Рисунок 4.8 – Алгоритм підготовки зображень перед передачею у модель

На початку виконується ініціалізація колекції для збереження результатів обробки, яка надалі міститиме підготовлені об'єкти. Далі система переходить до ітераційної обробки всіх вхідних елементів – кожен з яких є посиланням на зображення у файловій системі. Для кожного такого елементу завантажується зображення з диска, що є вихідним представленням візуальної інформації.

Після завантаження зображення виконується його масштабування до фіксованого розміру, як правило, 224×224 пікселі, що відповідає вимогам більшості попередньо навчених моделей. Далі ініціалізується масив пікселів, у якому будуть

зберігатись числові значення кольорових каналів кожного пікселя. Зображення проходить поелементний перебір по координатах, тобто кожен піксель аналізується окремо.

Для кожного пікселя виконується отримання його кольорового значення, представленого каналами R, G та B. На основі координат розраховується індекс у масиві, який забезпечує правильне збереження трьох послідовних значень для кожного пікселя. Далі відбувається нормалізація кольорових значень у діапазон [0; 1], що є стандартною практикою для забезпечення стабільності навчання та прогнозування.

Після нормалізації виконується формування нового об'єкта типу ModelInput, у якому зберігаються шлях до зображення, його мітка (за потреби) та нормалізований масив пікселів. Цей об'єкт додається до вихідного списку результатів. Завершується алгоритм поверненням повної колекції підготовлених елементів, які можуть бути використані для передбачення у моделі або подальшої обробки. Такий структурований підхід забезпечує повторюваність, масштабованість та відповідність до вимог сучасних систем комп’ютерного зору.

Алгоритм події завантаження зображення, зображений на рис. 4.9, реалізує ключову логіку взаємодії користувача з графічним інтерфейсом системи у момент, коли необхідно обрати файл для аналізу. Цей процес ініціюється натисканням кнопки завантаження, що активує низку перевірок та операцій, спрямованих на підготовку зображення до подальшої обробки і прогнозування.

На початку відбувається перевірка вибраної моделі – система переконується в тому, що користувач перед тим обрав конкретну модель машинного навчання зі списку доступних. Цей крок запобігає ситуації, коли прогнозування відбувається без визначеної аналітичної основи. Далі очищується вміст текстового поля звіту, у якому могли залишитися результати попередніх операцій, що забезпечує чистий контекст для нових повідомлень.

Наступним етапом є ініціалізація діалогового вікна вибору файлу, обмеженого фільтром типів даних, які відповідають графічному формату. Система переходить у стан очікування дій користувача – якщо файл було обрано, вона

продовжує виконання. Після підтвердження вибору виконується налаштування режиму відображення зображення, як правило, з параметром масштабування, що дозволяє коректно показати зображення у межах елементу керування.

Рисунок 4.9 – Алгоритм події завантаження зображення

Вибране зображення відображається на формі, що забезпечує користувачу миттєвий візуальний зворотний зв’язок про результат операції. У той же момент вміст зображення зчитується з файлової системи у вигляді масиву байтів, що надалі використовується у процедурі прогнозування або архівації. Завершальним кроком є збереження шляху до вибраного зображення у відповідній змінній, що дозволяє однозначно ідентифікувати файл у подальших етапах обробки. Таким чином, даний алгоритм забезпечує надійне та послідовне завантаження зображення в контексті інтелектуальної підсистеми.

Схема попередньої обробки зображення з ініціалізацією таймера, відображений на рис. 4.10, демонструє логіку виконання ключового етапу у процесі машинного аналізу зображень, коли необхідно не лише підготувати графічні дані для подальшого прогнозування, але й точно зафіксувати тривалість цієї операції.

Такий підхід дозволяє оцінити продуктивність системи при роботі з різними вхідними даними.

Рисунок 4.10 – Попередня обробка зображення з ініціалізацією таймера

Процес розпочинається з ініціації обробки зображення, тобто з моменту, коли користувач викликає відповідну функцію (наприклад, шляхом натискання кнопки «Прогнозувати»). Одразу після цього виконується ініціалізація таймера, що передбачає створення об'єкта для фіксації тривалості виконання обчислень. Це є підготовчим етапом до заміру продуктивності системи.

На наступному кроці таймер активується – починається фактичне вимірювання часу. Після запуску таймера формується список зображень, який зазвичай містить один або кілька об'єктів типу ModelInput, де вказано шлях до зображення. Цей список є підготовкою до подальшої передачі в модуль попередньої обробки.

Далі виконується попередня обробка зображення, яка включає масштабування, нормалізацію кольорових каналів та перетворення пікселів у формат, сумісний із моделлю. У результаті формується структура, придатна для подальшого прогнозування. Завершальним кроком алгоритму є отримання

обробленого результату, тобто підготовленого об'єкта, що вже містить числове представлення зображення, яке може бути безпосередньо передане до моделі машинного навчання для аналізу. Такий алгоритм дозволяє точно відслідковувати тривалість попередньої обробки та забезпечує узгоджене середовище для аналізу зображень у реальному часі.

Алгоритм прогнозування та зчитування часу виконання, представлений на рис. 4.11, є завершальним етапом обчислювального конвеєра, що реалізується після попередньої обробки вхідного зображення. Цей процес не лише виконує власне передбачення класу, а й забезпечує точний контроль за продуктивністю системи за допомогою вбудованого механізму таймеру.

```
1. Виконати прогноз за допомогою моделі:
   1.1. Передати підготовлені вхідні дані preprocessedInput у метод Predict(...)
   1.2. Зберегти результат у змінну result
2. Зупинити таймер, який вимірює тривалість операції
3. Зчитати значення часу виконання:
   3.1. Отримати тривалість з об’єкта stopwatch
   3.2. Зберегти її у змінну ts типу TimeSpan
```

(1. Виконати прогноз за допомогою моделі:
1.1. Передати підготовлені вхідні дані preprocessedInput у метод Predict(...)
1.2. Зберегти результат у змінну result
2. Зупинити таймер, який вимірює тривалість операції
3. Зчитати значення часу виконання:
3.1. Отримати тривалість з об’єкта stopwatch
3.2. Зберегти її у змінну ts типу TimeSpan)

Рисунок 4.11 – Виконання прогнозування та зупинка вимірювання часу

У першій фазі реалізується передача попередньо підготовлених вхідних даних до навченої моделі. Для цього об'єкт preprocessedInput, який містить нормалізоване зображення у форматі, придатному для інтерпретації моделлю, передається до методу Predict інструменту прогнозування. Результат цієї операції зберігається у відповідну змінну result, що виступає носієм усієї інформації про прогноз, зокрема мітку класу та впевненість моделі у зробленому рішенні.

Після отримання результату негайно зупиняється таймер, який до цього моменту здійснював вимірювання загальної тривалості обчислень. Це дозволяє точно зафіксувати час, необхідний системі для обробки одного запиту. Відтак відбувається звернення до об'єкта stopwatch, із якого зчитується значення накопиченого часу. Це значення зберігається у змінну ts типу TimeSpan, що забезпечує можливість подальшого форматування та виведення у звіт або лог. Такий алгоритм дозволяє не лише отримати результат прогнозування, а й оцінити ефективність його виконання з точки зору часових характеристик.

Псевдокод виведення часу, витраченого на обробку та розпізнавання зображення, наведений на рис. 4.12, деталізує завершальний етап прогнозного процесу, пов’язаний із формуванням і відображенням у звіті інформації про тривалість виконання. Така операція дозволяє не лише завершити сеанс розпізнавання, а й надати користувачеві зворотний зв’язок щодо продуктивності системи.

```
1. Сформувати рядок з інформацією про час розпізнавання:
   1.1. Взяти значення ts.Hours, ts.Minutes, ts.Seconds, ts.Milliseconds
   1.2. Відформатувати їх у вигляді: ГГ:ХХ:СС.МС
   1.3. Зберегти результат у змінну elapsedTime
2. Додати до текстового поля звіту новий рядок:
   2.1 "Час розпізнавання: {elapsedTime} мс."
```

(1. Сформувати рядок з інформацією про час розпізнавання:
1.1. Взяти значення ts.Hours, ts.Minutes, ts.Seconds, ts.Milliseconds
1.2. Відформатувати їх у вигляді: ГГ:ХХ:СС.МС
1.3. Зберегти результат у змінну elapsedTime
2. Додати до текстового поля звіту новий рядок:
2.1 "Час розпізнавання: {elapsedTime} мс.")

Рисунок 4.12 – Виведення часу, витраченого на обробку та розпізнавання

На початковому етапі здійснюється формування рядка з інформацією про час розпізнавання. Для цього з об’єкта ts, який має тип TimeSpan, вилучаються значення годин, хвилин, секунд і мілісекунд. Отримані дані підлягають

форматуванню у зручний для сприйняття вигляд – зазвичай у формі ГГ:ХХ:СС.МС, де кожен компонент представлено двозначним числом. Результат форматування записується до змінної elapsedTime, яка є проміжною і використовується виключно для відображення.

Після формування форматованого рядка виконується додавання нового рядка до текстового поля звіту. У поле виводиться повідомлення, що містить сформований текст із зазначенням точного часу, витраченого на виконання прогнозу. Таким чином, користувач не лише бачить результат розпізнавання, але й отримує оцінку швидкодії системи, що може бути використано для подальшого аналізу або порівняння моделей.

Схема аналізу рівня впевненості моделі, представлена на рис. 4.13, ілюструє логіку прийняття рішення щодо достовірності результату прогнозування, отриманого внаслідок обробки зображення. Такий підхід є характерним для систем машинного навчання, де необхідно відфільтровувати випадки, коли модель не продемонструвала достатньої впевненості у своєму рішенні.

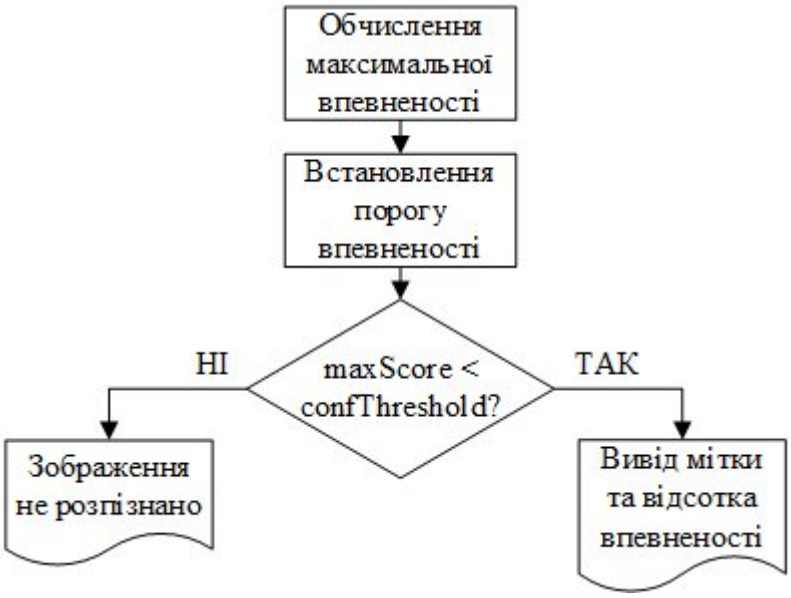


Рисунок 4.13 – Аналіз рівня впевненості моделі

На початку реалізується обчислення максимального значення впевненості серед усіх класів, які модель розглядає як можливі варіанти відповіді. Це значення

визначається як найбільший елемент у масиві оцінок і зберігається у змінну maxScore. Далі встановлюється порогове значення впевненості, яке визначає мінімально допустимий рівень достовірності для виведення прогнозу. Це значення, зазвичай емпірично обране, зберігається у змінній confThreshold і вважається критерієм для фільтрації результатів.

Після визначення обох величин виконується умовна перевірка: чи є отримане значення maxScore меншим за встановлений поріг confThreshold. Якщо ця умова справджується, тобто модель не досягла бажаного рівня впевненості, користувачу виводиться повідомлення про те, що зображення не було розпізнано. Такий варіант відповіді дозволяє уникнути помилкових результатів і гарантує надійність системи.

У протилежному випадку, коли модель виявляє впевненість, що перевищує порогове значення, система переходить до виведення результату. У цьому випадку користувачеві демонструється мітка розпізнаного класу та відсоткове значення впевненості, яке надає уявлення про силу аргументації моделі.

На рис. 4.14 представлена загальна архітектура СП на основі біометричних даних.

Архітектура складається з декількох основних компонентів, які забезпечують ефективне функціонування системи. Працівник взаємодіє з біометричним сканером, що зчитує унікальні характеристики обличчя та передає отримані дані у базу біометричних даних. Далі ці дані обробляються за допомогою моделі штучного інтелекту, яка порівнює отриману інформацію із збереженими шаблонами та ухвалює рішення щодо ідентифікації. Сервер контролю доступу виступає центральним вузлом обробки та прийняття рішень, взаємодіючи з базою даних, журналом аудиту та системою сповіщень. Для збереження історії дій система містить резервне сховище даних, що дозволяє вести архівні записи та забезпечує відновлення інформації у разі збою. Адміністратор отримує доступ до інформації через спеціалізований інтерфейс, що дозволяє керувати системою, перевіряти журнали подій та контролювати рівень доступу користувачів. Завдяки такій структурі система забезпечує високий рівень безпеки та зручність управління, інтегруючи в собі сучасні технології біометричної ідентифікації.

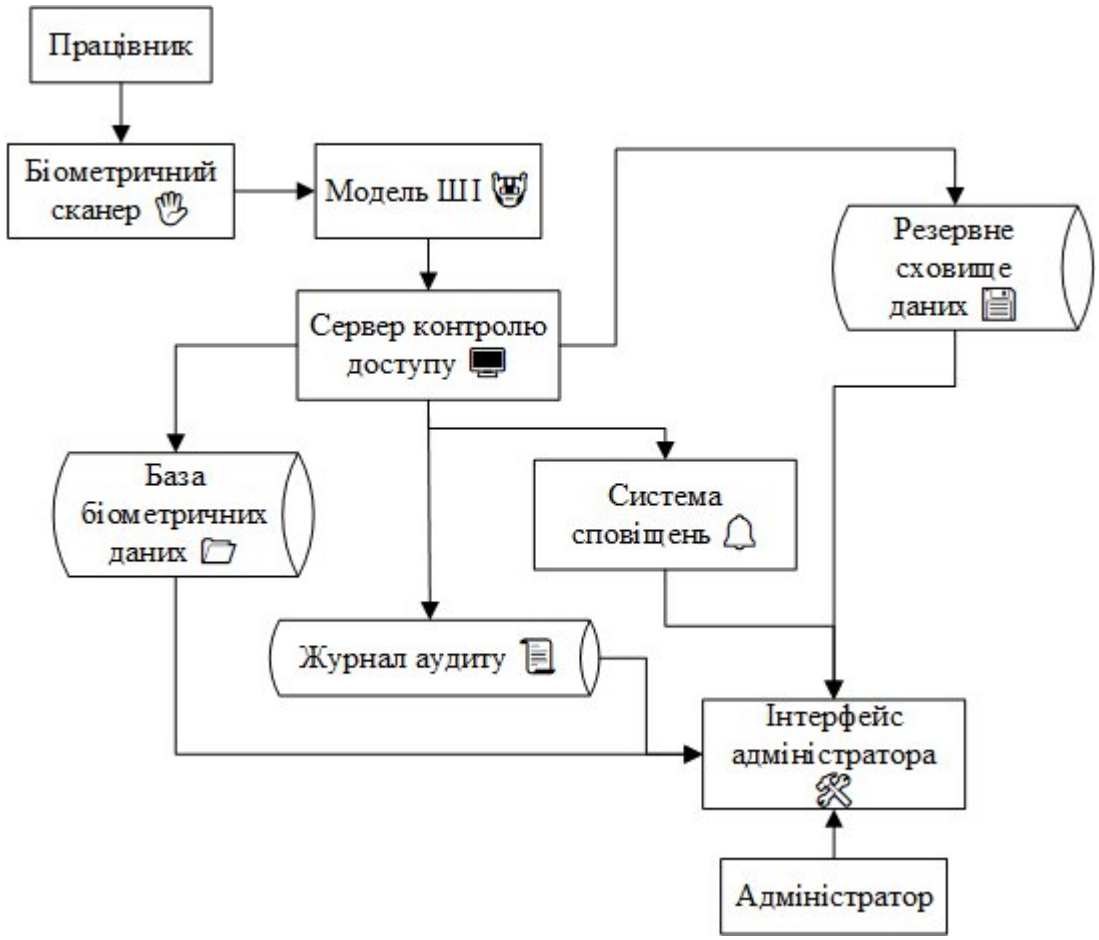


Рисунок 4.14 – Загальна архітектура системи

4.2 Аналіз результатів експериментів, та оцінка точності роботи системи

У процесі створення системи розпізнавання обличчя для контролю доступу критично важливим етапом є аналіз отриманих результатів та оцінка точності моделі. Висока точність класифікації дозволяє забезпечити надійну ідентифікацію осіб та мінімізувати ризики хибнопозитивних і хибнонегативних результатів. Основними метриками, що використовуються для оцінювання ефективності моделі, є MicroAccuracy, MacroAccuracy та LogLoss, які дозволяють комплексно оцінити якість роботи системи. У процесі тестування важливо враховувати як загальну точність класифікації, так і поведінку моделі у випадках різних варіацій освітлення, положення голови та інших факторів, що можуть впливати на розпізнавання.

На рис. 4.15 зображено процес навчання моделі, який включає кілька ключових етапів. Спочатку здійснюється завантаження каталогу зображень, що використовуються для навчання, після чого відбувається обробка даних та перетворення їх у формат, придатний для роботи з алгоритмами машинного навчання. Зображення конвертуються у числові вектори (тензори), що містять інформацію про колірні значення пікселів, після чого формується IDataView для подальшого навчання. Наступним кроком є розподіл вибірки на тренувальний та тестовий набори у співвідношенні 80/20, що дозволяє оцінити здатність моделі до узагальнення. На основі підготовлених даних створюється конвеєр навчання, який включає нормалізацію ознак. Завдяки цьому модель поступово коригує свої вагові коефіцієнти, адаптуючись до особливостей набору даних. Завершальним етапом є безпосереднє навчання моделі, після якого виконується оцінка її ефективності за основними метриками, включаючи точність класифікації та логарифмічні втрати.

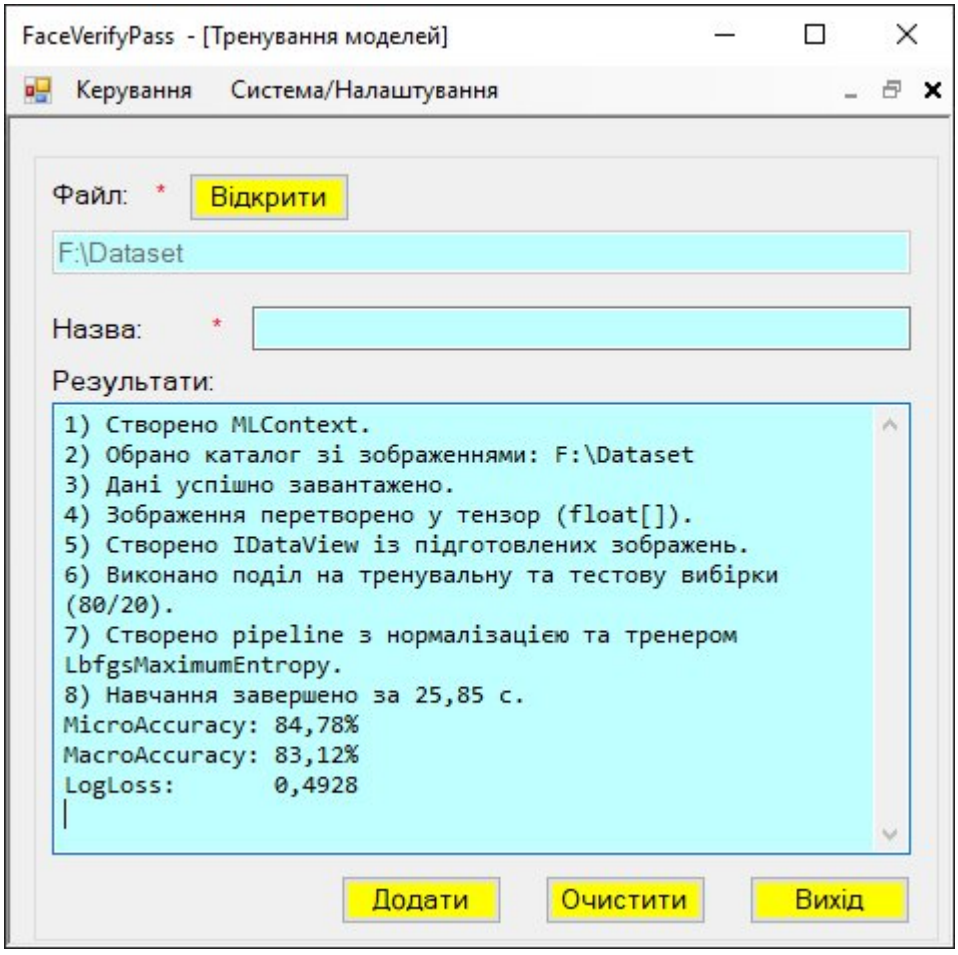


Рисунок 4.15 – Виведення інформації про результат навчання моделі

По завершенню навчання було виконано оцінку моделі, що дозволило отримати наступні метрики:

- LogLoss: 0,4928 – ця метрика оцінює, наскільки точними є передбачення моделі, зокрема, її здатність правильно прогнозувати ймовірності приналежності зображень до відповідних класів. Нижче значення log-loss вказує на більш точні передбачення. У даному випадку значення 0,4928 є досить прийнятним, що свідчить про хорошу здатність моделі коректно класифікувати зображення;
- MacroAccuracy: 83,12% – цей показник визначає середню точність класифікації, розраховуючи її окремо для кожного класу, а потім усереднюючи результати. Значення 83,12% свідчить про стабільність роботи моделі, але також вказує на можливі розбіжності в точності розпізнавання окремих класів. Це може бути пов’язано з різною кількістю зображень у класах або відмінностями у варіаціях зображень, що використовувалися для навчання;
- MicroAccuracy: 84,78% – ця метрика оцінює загальну кількість правильно класифікованих зразків у співвідношенні до загальної кількості передбачень. Високе значення мікро точності (84,78%) означає, що модель добре справляється із загальною задачею класифікації, враховуючи всі зразки рівномірно, незалежно від кількості зображень у кожному класі.

Загальний аналіз отриманих метрик свідчить про високий рівень ефективності моделі в розпізнаванні осіб на основі біометричних даних. Незважаючи на відносно високу точність, можлива подальша оптимізація, спрямована на покращення розпізнавання окремих класів та зниження log-loss, що дозволить ще більше підвищити точність передбачень у реальних умовах використання системи.

Після завершення процесу навчання моделі розпізнавання облич було проведено серію тестувань для перевірки її здатності правильно ідентифікувати особи на нових зображеннях. Основною метою цього етапу було визначити, наскільки модель адаптована до розпізнавання облич у різних умовах, зокрема при зміні ракурсу, освітлення або виразу обличчя.

Перевірка здійснювалася на тестовому наборі даних, який не входив до тренувальної вибірки, що дозволило отримати неупереджену оцінку її здатності до узагальнення. Особливу увагу приділяли ситуаціям, коли частина обличчя була закрита або в кадрі знаходилися сторонні об’єкти, що могло вплинути на точність класифікації. Аналіз отриманих результатів дозволив оцінити рівень загальної точності моделі та виявити можливі помилки, що виникають при розпізнаванні, тим самим визначивши її ефективність і стійкість до різних факторів у реальних умовах експлуатації.

На рис. 4.16 представлено результати першого сценарію тестування моделі розпізнавання обличчя.

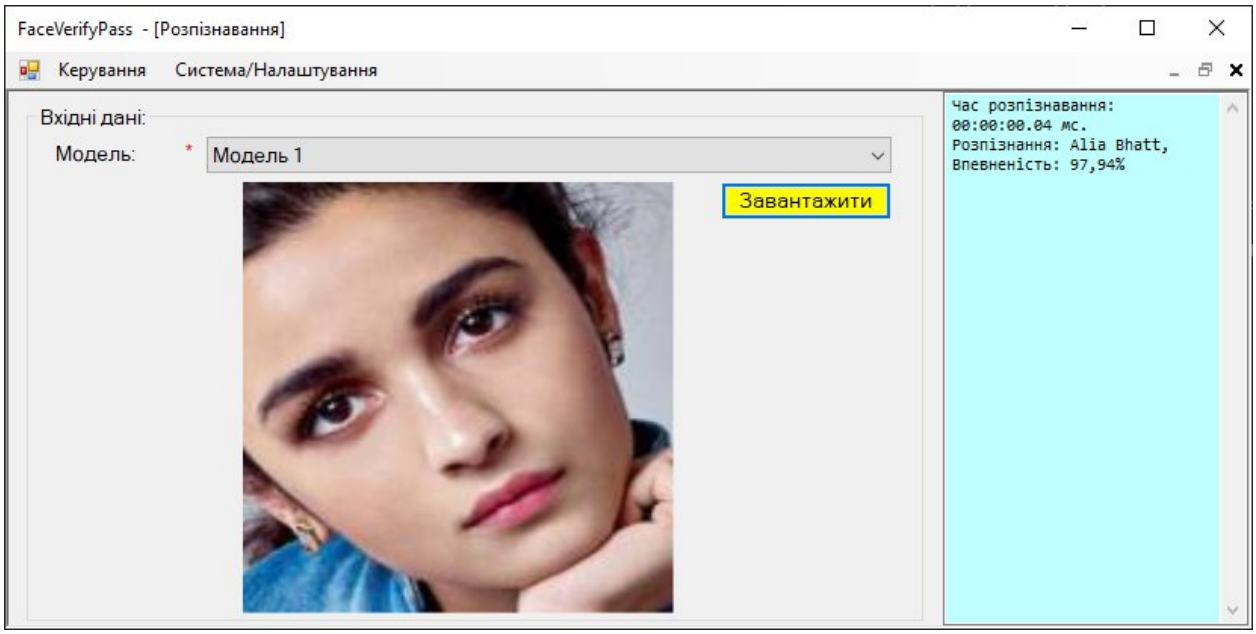


Рисунок 4.16 – Результат тестування 1-го сценарію

У цьому тестовому випадку модель успішно ідентифікувала особу, визначивши її як Alia Bhatt із високим рівнем впевненості – 97,94%. Отриманий показник свідчить про точність роботи алгоритму та його здатність до ефективного розпізнавання навіть у разі складних умов, таких як зміна ракурсу або освітлення. Час розпізнавання склав 00:00:00.04 мс, що вказує на високу швидкодію системи та її придатність для роботи в реальному часі.

Аналіз результатів показує, що система демонструє високу продуктивність та стабільність у процесі ідентифікації осіб. Швидкий час обробки та висока впевненість у результатах підтверджують, що модель добре оптимізована для використання в сценаріях автоматизованого контролю доступу.

На рис. 4.17 представлено результати другого сценарію тестування моделі розпізнавання обличчя.

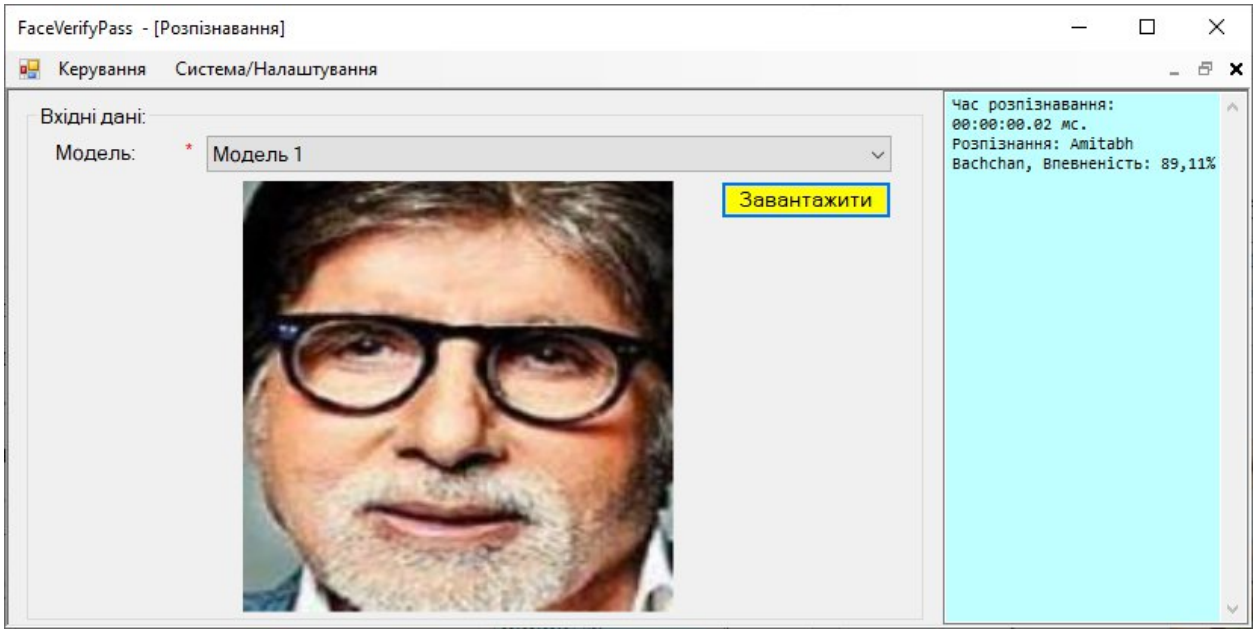


Рисунок 4.17 – Результат тестування 2-го сценарію

У цьому випадку модель успішно визначила особу як Amitabh Bachchan з рівнем впевненості 89,11%. Такий показник демонструє високу точність роботи алгоритму, хоча впевненість у розпізнаванні трохи нижча, ніж у попередньому сценарії. Це може бути пов’язано з особливостями зображення, такими як наявність аксесуарів (окулярів) або зміни в освітленні. Час розпізнавання склав 00:00:00.02 мс, що підтверджує швидкодію системи та її ефективність для застосування в режимі реального часу.

Аналіз результатів показує, що модель здатна правильно розпізнавати особи навіть за наявності можливих перешкод, таких як окуляри. Незважаючи на невелике зниження впевненості, система демонструє високу продуктивність, що робить її надійним рішенням для автоматизованих систем контролю доступу.

Рис. 4.18 ілюструє результати тестування моделі розпізнавання обличчя на зображенні, яке не входило до тренувального набору даних. У цьому випадку модель не змогла здійснити ідентифікацію особи, відобразивши повідомлення «Зображення не розпізнано». Такий результат очікуваний, оскільки система була навчена лише на певному наборі облич, і нова особа не мала відповідного класу в базі. Це свідчить про те, що алгоритм коректно виконує свою функцію, не приписуючи незнайоме обличчя до наявних класів, що є важливим аспектом для системи контролю доступу. Час розпізнавання склав 00:00:00.02 мс, що підтверджує високу продуктивність системи навіть у випадку, коли ідентифікація не відбулася.

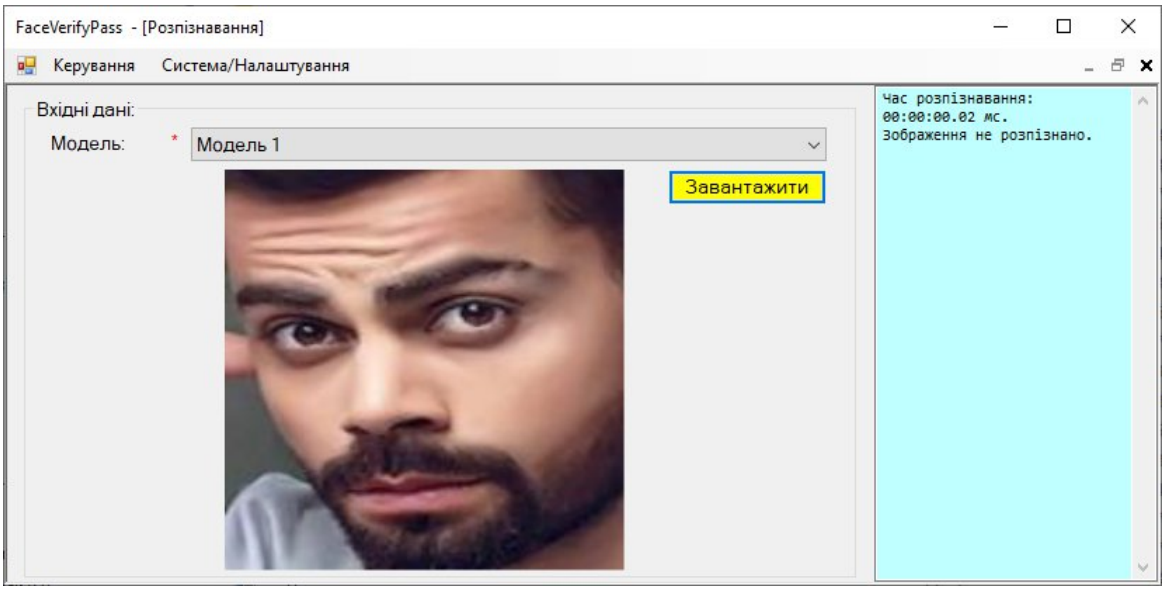


Рисунок 4.18 – Результат тестування 3-го сценарію

Цей сценарій демонструє обмеження моделі, зокрема необхідність розширення навчального датасету для охоплення більшої кількості осіб або використання методів додаткового донавчання системи. Крім того, результат свідчить про те, що модель використовує достатньо високий поріг впевненості для ідентифікації, що мінімізує ймовірність помилкових спрацьовувань, але водночас може призводити до відмови в розпізнаванні осіб, які раніше не зустрічалися в навчальному наборі. Це підтверджує необхідність подальшого вдосконалення

моделі, зокрема шляхом динамічного оновлення даних та адаптації алгоритму до нових користувачів системи.

Після завершення тестування розробленої системи розпізнавання обличь було проведено її порівняння з іншими відомими рішеннями, такими як DeepFace та Amazon Rekognition. DeepFace є відкритим фреймворком для розпізнавання облич, який базується на глибоких нейронних мережах і підтримує кілька моделей для виявлення та класифікації осіб. Amazon Rekognition, у свою чергу, є комерційним сервісом, що надає можливості розпізнавання обличь у хмарному середовищі та забезпечує високу точність за рахунок використання масштабованих обчислювальних ресурсів.

Основною метою порівняння було оцінити ефективність розробленої системи за ключовими параметрами: точністю розпізнавання, швидкістю обробки, вимогами до апаратних ресурсів та здатністю працювати у локальному середовищі. Аналіз дозволив визначити, у яких аспектах розроблене рішення є конкурентоспроможним, а також які його потенційні обмеження. Такий підхід дає змогу оцінити доцільність використання системи в різних умовах, зокрема у корпоративних чи державних установах, де необхідно забезпечити безперервну роботу без залежності від хмарних сервісів.

У табл. 4.1 представлено порівняння стійкості різних систем до змін зовнішніх умов, таких як освітлення, наявність аксесуарів (окуляри, головні убори) та варіації виразів обличчя. Розроблена система показала високу стійкість до змін освітлення та змогла правильно розпізнавати особи навіть за наявності окулярів. Однак її чутливість до зміни ракурсу є дещо вищою, ніж у Amazon Rekognition, який використовує складніші алгоритми узагальнення. DeepFace, у свою чергу, показав середню стійкість до аксесуарів, що може спричиняти помилки в розпізнаванні.

Таблиця 4.1 – Порівняння стійкості до змін зовнішніх умов

Система	Зміни освітлення	Наявність аксесуарів	Зміни виразу обличчя	Чутливість до шуму
Розроблена система	Висока	Впізнає з окулярами	Стійка до змін	Середня
DeepFace	Середня	Чутлива до аксесуарів	Чутлива до змін	Висока
Amazon Rekognition	Висока	Впізнає з окулярами	Стійка до змін	Низька

У табл. 4.2 представлено порівняння швидкості обробки зображень. Розроблена система забезпечує швидкість 4 мс на кадр, що дозволяє обробляти близько 250 кадрів за секунду. Це суттєво перевищує показники DeepFace, який працює повільніше через складність використовуваних моделей. Amazon Rekognition також демонструє хорошу швидкість, однак його залежність від інтернет-з'єднання може спричиняти додаткові затримки.

Таблиця 4.2 – Порівняння швидкості обробки

Система	Час розпізнавання (мс)	К-сть оброблених кадрів в секунду
Розроблена система	4	250
DeepFace	15	67
Amazon Rekognition	6	167

Окрему увагу слід приділити апаратним вимогам, наведеним у табл. 4.3. Розроблена система може функціонувати на середніх апаратних ресурсах, потребуючи лише 2-ядерного процесора та 4 ГБ оперативної пам'яті. Це робить її придатною для використання у локальних системах контролю доступу, де немає можливості застосовувати сервери з потужними графічними процесорами.

DeepFace має схожі вимоги, але через складнішу архітектуру може потребувати більше ресурсів при обробці великих наборів даних. Amazon Rekognition не вимагає специфічних локальних ресурсів, але залежить від хмарних обчислень, що може бути недоліком у середовищах з обмеженим доступом до мережі.

Таблиця 4.3 – Порівняння вимог до апаратних ресурсів

Система	Мінімальні апаратні вимоги	Рекомендовані апаратні вимоги	Підтримувані платформи
Розроблена система	CPU: 2 ядра, RAM: 4 ГБ	CPU: 4 ядра, RAM: 8 ГБ	Windows, Linux
DeepFace	CPU: 2 ядра, RAM: 4 ГБ	CPU: 4 ядра, RAM: 8 ГБ	Windows, Linux, MacOS
Amazon Rekognition	Хмарне рішення	Хмарне рішення	Windows, Linux, MacOS, Android, iOS

Загальний аналіз порівняння показав, що розроблена система є ефективним рішенням для локальних впроваджень, оскільки не потребує потужного апаратного забезпечення та може працювати автономно. Вона демонструє високу швидкість обробки зображень, що робить її придатною для використання у реальних сценаріях з високими вимогами до продуктивності. Хоча Amazon Rekognition забезпечує вищу точність і має кращу стійкість до шуму, він залежить від хмарної інфраструктури, що може бути неприйнятним у деяких сценаріях. Враховуючи це, розроблена система є оптимальним варіантом для застосувань, де необхідне швидке та автономне розпізнавання обличчя без залежності від зовнішніх сервісів.

Хоча розроблена система продемонструвала високу точність і швидкість розпізнавання, завжди існують можливості для її подальшого вдосконалення. Оптимізація моделей машинного навчання та покращення процесів обробки даних можуть значно підвищити точність і надійність ідентифікації осіб у системах

контролю доступу. Нижче розглянуто основні напрями, які можуть бути застосовані для покращення роботи системи.

Одним із ключових способів підвищення точності моделі є збільшення обсягу тренувального набору даних. Використання більшої кількості зображень із різними варіаціями освітлення, ракурсів та виразів обличчя сприятиме кращому узагальненню моделі. Окрім цього, застосування методів аугментації, таких як обертання, зміна контрасту, масштабування та дзеркальне відображення, дозволить штучно розширити набір даних без необхідності додаткового збору нових зображень.

Також перспективним напрямом є використання більш складних нейронних архітектур. Хоча поточна модель базується на методі максимальної ентропії, застосування глибших згорткових нейронних мереж, таких як EfficientNet або Vision Transformers, може значно покращити розпізнавання складних патернів обличчя. Ці моделі відзначаються високою точністю при мінімальних обчислювальних витратах, що дозволяє використовувати їх навіть на апаратних ресурсах середнього рівня.

Важливим аспектом є також покращення процесу попередньої обробки зображень. Використання алгоритмів вирівнювання обличчя на основі ключових точок (очі, ніс, рот) може суттєво покращити стандартизацію вхідних зображень, що зменшить вплив різних ракурсів на точність розпізнавання. Крім того, застосування фільтрації шуму та підвищення контрасту дозволить покращити якість вхідних даних, особливо у випадках роботи з камерами відеоспостереження або в умовах слабкого освітлення.

Розширення можливостей моделі може також включати впровадження методів багатокласової класифікації для розпізнавання кількох осіб одночасно. Це особливо важливо для систем контролю доступу в місцях із великим потоком людей, де потрібно швидко й ефективно обробляти кілька осіб у кадрі. Використання ансамблевих методів, які поєднують кілька моделей для підвищення точності класифікації, також може бути ефективним рішенням.

Для забезпечення довготривалої ефективності моделі необхідно передбачити механізм регулярного оновлення та повторного навчання. Оскільки система може стикатися з новими типами зображень або змінами у зовнішності користувачів (наприклад, зміна зачіски, наявність бороди, окулярів тощо), періодичне оновлення тренувального набору дозволить моделі адаптуватися до нових умов і підтримувати високу точність розпізнавання.

Отже, підвищення ефективності системи розпізнавання облич може бути досягнуто шляхом розширення тренувального набору, використання сучасних архітектур нейронних мереж, покращення попередньої обробки зображень та впровадження механізмів регулярного оновлення. Комплексний підхід до цих аспектів дозволить зробити систему більш надійною, точною та адаптивною до реальних умов експлуатації.

4.3 Висновки

У рамках даного розділу було здійснено проектування та реалізацію програмно-технічного засобу для СП на основі біометричних даних, що дозволило створити ефективну трьохрівневу архітектуру з чітким розподілом функцій між рівнями. Детально розглянуто всі складові системи, включаючи рівень даних, рівень бізнес-логіки та рівень користувацького інтерфейсу. Для кожного з рівнів побудовано відповідні діаграми класів та проаналізовано їх функціональні можливості. Окрім цього, було розроблено алгоритми навчання моделей та прогнозування, що забезпечують високу швидкість та точність обробки зображень.

У процесі експериментального дослідження проведено тренування моделі, яке дозволило отримати наступні показники якості: MicroAccuracy – 84,78%, MacroAccuracy – 83,12%, LogLoss – 0,4928. Це підтверджує ефективність розробленого алгоритму у вирішенні завдань розпізнавання облич для біометричної ідентифікації в системах контролю доступу. Виконане тестування дозволило оцінити стійкість системи до змін зовнішніх умов, зокрема до змін освітлення, наявності аксесуарів та варіацій у виразах обличчя. У порівнянні з

аналогічними рішеннями, такими як DeepFace та Amazon Rekognition, розроблена система продемонструвала конкурентну точність розпізнавання, водночас забезпечуючи суттєву перевагу в швидкості обробки, що становить 4 мс на кадр, що значно випереджає DeepFace (15 мс) і перевершує Amazon Rekognition (6 мс).

Проведене порівняння також показало, що розроблена система має оптимальні вимоги до апаратних ресурсів, дозволяючи її використання навіть на пристроях із мінімальними характеристиками (2-ядерний процесор, 4 ГБ оперативної пам’яті), на відміну від Amazon Rekognition, який вимагає хмарних обчислень. Це забезпечує можливість локального впровадження системи без необхідності постійного підключення до мережі та використання сторонніх сервісів, що є важливим фактором для підприємств та установ із підвищеними вимогами до конфіденційності даних.

Загальний аналіз отриманих результатів підтверджує, що розроблена система є ефективним рішенням для автоматизації пропускного контролю на основі біометричної ідентифікації. Вона забезпечує високу швидкість обробки, достатню точність розпізнавання, а також можливість роботи в локальному середовищі з помірними апаратними вимогами.

ВИСНОВКИ

У ході виконання кваліфікаційної роботи було проведено комплексне дослідження та реалізацію методу і програмно-технічного засобу для СП на основі біометричних даних. На початковому етапі було здійснено аналіз існуючих систем контролю доступу, серед яких розглянуто механічні, електронні та біометричні рішення. Було встановлено, що саме БС забезпечують найвищий рівень надійності, швидкості та зручності використання, що підтверджено відповідними характеристиками. Проведений огляд сучасних технологій продемонстрував активне впровадження біометричних методів у світовій практиці, тоді як в Україні їх застосування поки що має певні обмеження. Визначено, що для розробки програмного забезпечення доцільно використовувати мову C# у середовищі Visual Studio 2022, що зумовлено стабільністю екосистеми, підтримкою машинного навчання через ML.NET та можливістю гнучкої інтеграції з базами даних і системами контролю доступу.

У другому розділі розглянуто принципи використання біометричних даних у системах контролю доступу, зокрема геометрію руки, відбитки пальців, сканування сітківки, розпізнавання обличчя та голосову ідентифікацію. Проведено детальний аналіз архітектур нейронних мереж, таких як LeNet-5, AlexNet, ResNet та YOLOv11, що застосовуються для задач розпізнавання зображень. Для навчання моделі було використано датасет Face Recognition Dataset із платформи Kaggle, що містить зображення осіб у різних умовах освітлення, з різними ракурсами та виразами обличчя. Проведено підготовку даних, зокрема нормалізацію зображень, їхнє приведення до єдиного розміру та структурування вибірки для тренування.

У третьому розділі обґрунтовано вибір методу навчання моделі для розпізнавання обличчя. Було вирішено використовувати метод максимальної ентропії, що реалізує логістичну регресію для багатокласової класифікації з використанням алгоритму L-BFGS. Обраний підхід дозволяє отримати високу точність при помірних обчислювальних витратах, що є важливим для систем реального часу. Розроблено та представлено алгоритм навчання моделі, що

включає етапи підготовки даних, їхнього перетворення у відповідний формат, тренування та оцінки моделі. Також було розглянуто методи обробки похибок розпізнавання, включаючи аналіз показників False Acceptance Rate та False Reject Rate, що дозволяє оптимізувати баланс між точністю і швидкістю роботи системи.

У четвертому розділі проведено детальний аналіз роботи системи, включаючи оцінку її продуктивності, порівняння з існуючими рішеннями та визначення ефективності реалізованого підходу. Навчання моделі дало такі результати: MicroAccuracy – 84,78%, MacroAccuracy – 83,12%, LogLoss – 0,4928, що свідчить про високу якість класифікації та здатність системи точно розпізнавати обличчя. Система продемонструвала хорошу стійкість до змін умов середовища, таких як варіації освітлення, присутність аксесуарів та зміни міміки, що підтверджує її практичну придатність. У процесі порівняльного аналізу з відомими аналогами, зокрема DeepFace та Amazon Rekognition, виявлено, що запропоноване рішення має суттєві переваги у швидкості обробки, забезпечуючи розпізнавання за 4 мс, що перевищує показники DeepFace (15 мс) і є кращим за Amazon Rekognition (6 мс). Окрім того, система не потребує значних обчислювальних ресурсів, що дає змогу її впровадження навіть у пристроях із мінімальними апаратними характеристиками, забезпечуючи автономну роботу без необхідності підключення до хмарних сервісів.

Розроблена система повністю відповідає сучасним вимогам до біометричних технологій і забезпечує ефективне рішення для автоматизованого контролю доступу. Вона поєднує точність, швидкодію та оптимальну ресурсоефективність, що робить її придатною для використання в різних середовищах, включаючи корпоративні, державні та приватні об'єкти. Алгоритм, що лежить в основі системи, забезпечує високу якість розпізнавання навіть у складних умовах, а гнучкість підходу дозволяє адаптувати систему під конкретні потреби користувачів. Отримані результати свідчать про ефективність запропонованого методу, що підтверджується його конкурентоспроможністю у порівнянні з іншими популярними рішеннями.

Подальші дослідження можуть бути спрямовані на розширення навчального набору даних для покращення узагальнюючої здатності моделі та підвищення її адаптивності до нових сценаріїв використання. Також доцільним є впровадження вдосконалених методів обробки зображень, таких як попереднє вирівнювання облич і зменшення шумів, що дозволить знизити ймовірність помилкових спрацьовувань. Крім того, інтеграція системи з іншими методами аутентифікації та використання гібридних рішень може підвищити загальну безпеку та функціональність. Перспективним напрямом розвитку є також оптимізація моделі для роботи в режимі реального часу та покращення її продуктивності на пристроях із обмеженими ресурсами.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАНЬ

1. Markert P., Bailey D., Golla M., Dürmuth M., Aviv A. On the security of smartphone unlock pins. *ACM Transactions on Privacy and Security (TOPS)*. 2021. Vol. 24. No 4. pp. 1-36.

2. Liang Y., Samtani S., Guo B., & Yu Z. Behavioral biometrics for continuous authentication in the internet-of-things era: An artificial intelligence perspective. *IEEE Internet of Things Journal*. 2020. Vol. 7. No 9. pp. 128-143.

3. Albalawi S., Alshahrani L., Albalawi N., Kilabi R., & Alhakamy A. A comprehensive overview on biometric authentication systems using artificial intelligence techniques. *International Journal of Advanced Computer Science and Applications*. 2022. Vol. 13. No 4. pp. 1-11.

4. Голуб'як І. В., Косаревич Р. Я. Методи розпізнавання обличчя. *Проблеми інформаційних технологій*. 2017. № 22. С. 158 – 164.

5. Ковальчук Ю. А., Головко В. В. Використання системи розпізнавання обличчя для ідентифікації особи. *Технічні науки та технології*. 2019. № 1. Вип. 15. С. 201–208.

6. Tavenard, R., Faouzi J., Vandewiele G., Divo F., Androz G., Holtz C., Woods E. Tslearn. A machine learning toolkit for time series data. *Journal of machine learning research*. 2020. Vol. 21. No 118. pp. 1-6.

7. Microsoft. ML.NET: Machine Learning Made for .NET. URL: https://dotnet.microsoft.com/en-us/apps/ai/ml-dotnet (дата звернення: 01.03.2025).

8. Iqbal U., Mir A. H. Efficient and dynamic access control mechanism for secure data acquisition in IoT environment. *Int. J. Com. Dig. Sys*. 2021. Vol. 10. No 1. pp. 31-46.

9. Khaleel M., Yaghoubi E., Yaghoubi E., & Jahromi M. The role of mechanical energy storage systems based on artificial intelligence techniques in future sustainable energy systems. *Int. J. Electr. Eng. and Sustain*. 2023. 1-31 p.

10. Masoumzadeh A., Dercksen A. BlueSky: physical access control: Characteristics, challenges, and research Opportunities. *In Proceedings of the 27th ACM on Symposium on Access Control Models and Technologie*s. 2022. pp. 163-172.

11. Al-Natour S., Cavusoglu H., Benbasat I., & Aleem U. An empirical investigation of the antecedents and consequences of privacy uncertainty in the context of mobile apps. *Information Systems Research*. 2020. Vol. 31. No 4. pp. 1037-1063.

12. Ajavon A., & Srivaramangai P. Electronic Health Record's, Data Security and Access Control in Medical Centers in Ghana. *International Journal of Information Technology and Management Information Systems (IJITMIS)*. 2023. Vol. 14. No 1. pp. 21-32.

13. Sulaiman, R., Isah A., & Shin H. Exploring Near-Field Communication Technology for Academia Identification Card System. *The Journal of Contents Computing*. 2023. Vol. 5. No 1. pp. 599-607.

14. Ragothaman K., Wang Y., Rimal B., & Lawrence M. Access control for IoT: A survey of existing research, dynamic policies and future directions. Sensors. 2023. Vol. 23. No 4. 805p.

15. Lee H., Park S. H., Yoo J. H., Jung S. H., & Huh J. H. Face recognition at a distance for a stand-alone access control system. *Sensors*. 2020. Vol. 20. No 3. 785 p.

16. Awad A., Babu A., Barka E., & Shuaib K. AI-powered biometrics for Internet of Things security: A review and future vision. *Journal of Information Security and Applications*. 2024. Vol. 82. 128 p.

17. Eddine B., & Zohra, C. Real Time Face Recognition System using Raspberry Pi4. *In 2024 3rd International Conference on Advanced Electrical Engineering (ICAEE)*. 2024. pp. 1-7.

18. Minaee S., Abdolrashidi A., Su H., Bennamoun M., & Zhang D. Biometrics recognition using deep learning: *A survey. Artificial Intelligence Review*. 2023. Vol. 56. No 8. pp. 647-695.

19. Adjabi I., Ouahabi A., Benzaoui A., & Taleb-Ahmed A. Past, present, and future of face recognition: *A review. Electronics*. 2020. Vol. 9. No 8. 1188 p.

20. Kurylo M., Klochko A., Klietsova N., & Bolotina A. The use of biometric technologies for bank transaction security management against the background of the international experience: *Evidence from Ukraine. Banks and Bank Systems*. 2021. Vol. 16. No 2. 47 p.

21. Wu X., Zhou Z., & Chen S. A mixed-methods investigation of the factors affecting the use of facial recognition as a threatening *AI application. Internet Research*. 2024. Vol. 34. No 5. pp. 872-897.

22. RFID HOTEL and ID&C enhances strategic collaboration with Vingcard for RFID credentials in US and Canada. URL: https://www.hospitalitynet.org/news/4122542.html (дата звернення: 01.03.2025).

23. An J., Yuen C., Dai L., Di Renzo M., Debbah M., & Hanzo L. Near-field communications: Research advances, potential, and challenges. *IEEE Wireless Communications*. 2024. Vol. 31. No 3. pp. 100-107.

24. Hyatt Becomes First Hotel Brand to Offer Digital Room Keys in Apple Wallet on iPhone and Apple Watch. URL: https://hoteltechnologynews.com/2021/12/hyatt-becomes-first-hotel-brand-to-offer-digital-room-keys-in-apple-wallet-on-iphone-and-apple-watch/ (дата звернення: 01.03.2025).

25. Trabelsi Z., & Parambil M. A bibliometric study on recent trends in artificial intelligence-based suspicious activity recognition. *Security Journal*. 2024. Vol. 37. No 2. pp. 399-424.

26. Motorola Solutions -turvajärjestelmien teoriapohja. URL: https://www.theseus.fi/bitstream/handle/10024/818061/Hirvonen_Jere.pdf?sequence=2 (дата звернення: 01.03.2025).

27. Тітова В., Кльоц Ю., Мостовий С., Колісник В. Система контролю та управління доступом на основі RFID-технологій. *Вимірювальна та обчислювальна техніка в технологічних процесах*. 2023. № 4. С. 44–48.

28. Бобало Ю. Я., Горбатий І. В., Кіселичник М. Д. Інформаційна безпека : навч. посібник. – *Львів : Видавництво Львівської політехніки*. 2019. 580 с.

29.BAS-IP Intercom. URL: https://www.bas-ip.com.ua/catalog/soft/bas-ip-intercom/ (дата звернення: 01.03.2025).

30. SmartPlus - мобільний додаток для домофонів Akuvox. URL: https://tiandy.com.ua/smartplus-mobilniy-dodatok-dlya-domofoniv-akuvox/ (дата звернення: 01.03.2025).

31. Smith M., & Miller S. The ethical application of biometric facial recognition technology. *Ai & Society*. 2022. Vol. 37. No 1. pp.167-175.

32. Тарасевич Т. Ю. Правове регулювання біометричної ідентифікації особи: національні тенденції та зарубіжний досвід. *Часопис Київського університету права*. 2021. № 2. С. 280–284.

33. Kyiv Smart Card – як це працює? URL: https://finance.ua/ua/saving/kyivsmartcard (дата звернення: 01.03.2025).

34. Оплата за допомогою селфі: чи замінить біометрія гаманці та банківські картки. URL: https://new.minfin.com.ua/ua/privatbank/fin-projects/facepay (дата звернення: 01.03.2025).

35. Система FacePay24 від ПриватБанку. URL: https://www.pravda.com.ua/archives/date_13092019/ (дата звернення: 01.03.2025).

36. Xanthidis D., Manolas C., Xanthidou O. K., & Wang H. I. Handbook of Computer Programming with Python. *CRC Press*. 2022. 418 p.

37. Simwinga A., & Phiri J. Integration of Facial Recognition. *In Proceedings of Ninth International Congress on Information and Communication Technology: ICICT*. 2024. Vol. 8. No 95. 112 p.

38. Beckert B., Bubel R., Drodt D., Hähnle R., Lanzinger F., Pfeifer W., ... & Weigl A. The Java Verification Tool KeY: *A Tutorial. In International Symposium on Formal Methods*. 2024. pp. 597-623.

39. Коноваленко І.В., Марущак П.О. Платформа .NET та мова програмування C# 8.0: навчальний посібник. *Тернопіль: ФОП Паляниця В. А*. 2020. 320 с.

40. Verma R. Extending Visual Studio. *In Visual Studio Extensibility Development: Extending Visual Studio IDE for Productivity, Quality, Tooling, Analysis, and Artificial Intelligence*. 2023. pp. 73-113.

41. Baptista G., & Abbruzzese F. Software Architecture with C# 10 and. NET 6: Develop software solutions using microservices, DevOps, EF Core, and design patterns for Azure. *Packt Publishing Ltd*. 2022. 513 p.

42. Visual Studio Code Reviews & Product Details. URL: https://www.g2.com/products/visual-studio-code/reviews (дата звернення: 01.03.2025).

43. Ingebrigtsen E. Metaprogramming in C#: Automate your. NET development and simplify overcomplicated code. *Packt Publishing Ltd*. 2023. 892 p.

44. Rider for C# - The Best Visual Studio Alternative IDE . URL: https://developer.okta.com/blog/2020/11/30/rider-csharp-visual-studio-alternative (дата звернення: 01.03.2025).

45. Hong, K. Developing A 2d Platform Endless Runner Game With Unity Engine. 2023. 312 p.

46. Haider S., Rehman Y., AliS. U. Enhanced multimodal biometric recognition based upon intrinsic hand biometrics. *Electronics*. 2020. Vol. 9. No 11. 814 p.

47. Štitilis D., Laurinaitis M., & Verenius E. The use of biometric technologies in ensuring critical infrastructure security: the context of protecting personal data. *Entrepreneurship and sustainability issues*. 2023. Vol. 10. No 3. 133 p.

48. Khan S., Parkinson S., Grant L., Liu N., Mcguire S. Biometric systems utilising health data from wearable devices: Applications and future challenges in computer security. *ACM Computing Surveys (CSUR)*. 2020. Vol. 53. No 4. pp. 1-29.

49. Новіцький Г.М. Методи біометричної ідентифікації. *Збірник наукових праць Вінницького національного технічного університету*. 2019. № 1. С. 1–4.

50. Fernández N., Godinho R., García M., Garate M. Exploring the utility of Geometric Morphometrics to analyse prehistoric hand stencils. *Scientific Reports*. 2024. Vol. 14. No 1. 518 p.

51. North-Samardzic A. Biometric technology and ethics: Beyond security applications. *Journal of Business Ethics*. 2020. Vol. 167. No 3. pp. 433-450.

52. Tu Y., Yao Z., Xu J., Liu Y., & Zhang Z. Fingerprint restoration using cubic Bezier curve. *BMC bioinformatics*. 2020. Vol. 21. pp. 1-19.

53. Dabous, S. A., & Feroz, S. Condition monitoring of bridges with non-contact testing technologies. *Automation in Construction*. 2020. 116 p.

54. Черкащенко А., Гадецька З. Біометричні методи захисту інформації. *Матеріали Міжнародної науково-практичної інтернет-конференції «Тенденції та перспективи розвитку науки і освіти в умовах глобалізації». Переяслав*. 2020. Вип. 65. С. 321–324.

55. Kumar K., & Singh N. Analysis of retinal blood vessel segmentation techniques: a systematic survey. *Multimedia Tools and Applications*. 2023. Vol. 82. No 5. pp. 679-773.

56. Луценко М.С., Кузнєцов О.О., Горбенко Ю.І., Пушкарьов А.І., Уварова А.О. Генерація ключів з біометричних образів райдужної оболонки ока. 2018. Т. 17. № 3. С. 104–110.

57. Nguyen K., Proença H., & Alonso-Fernandez F. Deep learning for iris recognition: A survey. *ACM Computing Surveys*. 2024. Vol. 56. No 9. pp. 1-35.

58. Krishnendu K. Analysis of Recent Trends in Face Recognition Systems. *CoRR*. 2023. 12 p.

59. Kim B., Seo S. Intelligent digital human agent service with deep learning based-face recognition. *IEEE Access*. 2022. pp. 794-805.

60. Ануфрієв П.О. Удосконалення технології розпізнавання облич за допомогою згорткових нейронних мереж. *Наукові праці ДонНТУ. Серія «Інформатика, кібернетика та обчислювальна техніка»*. 2021. № 2. Вип.33. С. 70–77.

61. Taherkhani A., Cosma G., McGinnity T. AdaBoost-CNN: An adaptive boosting algorithm for convolutional neural networks to classify multi-class imbalanced datasets using transfer learning. *Neurocomputing*. 2020. Vol. 404. pp. 351-366.

62. Кравченко С.М., Предчук Т.В. Згорткова нейронна мережа для розпізнавання об'єктів. *Матеріали конференції «Сучасні інформаційні системи та технології». Житомир: Державний університет «Житомирська політехніка»*. 2024. С. 45–48.

63. Mazumdar E., Ratliff L., & Sastry S. On gradient-based learning in continuous games. *SIAM Journal on Mathematics of Data Science*. 2020. Vol. 2. No 1. pp. 103-131.

64. Zhang J., Yu X., Lei X., Wu C. A novel deep LeNet-5 convolutional neural network model for image recognition. *Computer Science and Information Systems*. 2022. Vol. 19. No 3. pp. 1463-1480.

65. LeNet-5 - A Classic CNN Architecture. URL: https://www.datasciencecentral.com/lenet-5-a-classic-cnn-architecture/ (дата звернення: 04.03.2025).

66. Rouhsedaghat M., Wang Y., Ge X., Hu S., You S., & Kuo C. Facehop: A light-weight low-resolution face gender classification method. *In Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event. Springer International Publishing*. 2021. pp. 169-183.

67. Зелінський Ю. П., Кравченко С. М. Розпізнавання емоційних виразів обличчя людини за допомогою згорткових нейронних мереж. *Вчені записки ТНУ імені В. І. Вернадського. Серія: Технічні науки*. 2021. Т. 32. № 5. С. 88–92.

68. Alay N., Al-Baity H. Deep learning approach for multimodal biometric recognition system based on fusion of iris, face, and finger vein traits. *Sensors*. 2020. Vol. 19. 523 p.

69. Ariff N., Ismail A., Aziz N., Hussin A. Analysis of optimizers on AlexNet Architecture for face biometric authentication system. *In 2022 International Conference on Information Technology Research and Innovation.* 2022. pp. 24-29.

70. Architecture AlexNet. URL: https://medium.com/@saba99/alexnet-f0cb53648600 (дата звернення: 04.03.2025).

71. Almabdy S., Elrefaei L. An overview of deep learning techniques for biometric systems. *Artificial Intelligence for Sustainable Development: Theory, Practice and Future Applications*. 2021. pp. 127-170.

72. Zhao X., Wang L., Zhang Y., Han X., Deveci M., Parmar M. A review of convolutional neural networks in computer vision. Artificial Intelligence Review. 2024. Vol. 57. No 4. 99 p.

73. Masmoudi Y., Ramzan M., Khan S., Habib M. Optimal feature extraction and ulcer classification from WCE image data using deep learning. *Soft Computing*. 2022. Vol. 26. No 16. pp. 979-992.

74. Understanding ResNet-50 in Depth: Architecture, Skip Connections, and Advantages Over Other Networks. URL: https://wisdomml.in/understanding-resnet-50-in-depth-architecture-skip-connections-and-advantages-over-other-networks/ (дата звернення: 04.03.2025).

75. Nyarko B., Bin W., Zhou J., Agordzo G., Odoom J., Koukoyi E. Comparative analysis of AlexNet, Resnet-50, and Inception-V3 models on masked face recognition. *(AIIoT)*. 2022. pp. 337-343.

76. Cheng S., Han Y., Wang Z., Liu S., Yang B., Li J. An Underwater Object Recognition System Based on Improved YOLOv11. *Electronics*. 2025. Vol. 14. No. 1. 201 p.

77. Zou C., Yu S., Yu Y., Gu H., Xu X. Side-Scan Sonar Small Objects Detection Based on Improved YOLOv11. *Journal of Marine Science and Engineering*. 2025. Vol. 13. No 1. 162 p.

78. YOLOv11 Architecture Explained: Next-Level Object Detection with Enhanced Speed and Accuracy. URL: https://medium.com/@nikhil-rao-20/yolov11-explained-next-level-object-detection-with-enhanced-speed-and-accuracy-2dbe2d376f71 (дата звернення: 04.03.2025).

79. Yang J., Tian T., Liu Y., Li C., Wu D., Wang L., Wang, X. A Rainy Day Object Detection Method Based on YOLOv11 Combined with FFT and MF Model Fusion. In 2024 International Conference on Advanced Control Systems and Automation Technologies (ACSAT). 2024. pp. 246-250.

80. Face Recognition Dataset. URL: https://www.kaggle.com/datasets/cybersimar08/face-recognition-dataset (дата звернення: 04.03.2025).

81. Li G., Ji Z., Chang Y., Li S., Qu X., Cao D. ML-ANet: A transfer learning approach using adaptation network for multi-label image classification in autonomous driving. *Chinese Journal of Mechanical Engineering*. 2021. Vol. 34. pp. 1-11.

82. Liu C., Song Y., Li R., Ma W., Hao J. L., Qiang G. Three-level modular grid system for sustainable construction of industrialized residential buildings: *A case study in China. Journal of Cleaner Production*. 2023. 395 p.

ДОДАТОК А
(обов'язковий)

ЛІСТИНГ ПРОГРАМИ

Код програми, яка була розроблена для виконання роботи.

```
using PassSystemApp.AppCode;
using PassSystemApp.Forms.Systems;
using PassSystemApp.Providers;
using Microsoft.ML;
using Microsoft.ML.Data;
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Diagnostics;
using System.Drawing;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;


namespace PassSystemApp.Forms.Dictionary {
  public partial class TrainModelsForm : Form {
    private int _selectedRowIndex = 0;

    private MLContext mlContext;
    private IDataView data;
    private ITransformer model;
    private IDataView trainData;

    private string _imagePath = "F:\\Data";
    private string _modelPath = "model.zip";
    private string _predictedLabelKeyColumnName =
"PredictedLabelKey";

    private string _keyColumnName = "LabelKey";
    private string _predictedLabelColumnName = "PredictedLabel";
    private string _featuresColumnName = "ImagePixels";

    private bool _IsModelTrain = false;
    private ValidationMy _Validation = new ValidationMy();
```

```
    private ModelsProvider _ModelsProvider = new ModelsProvider();
    private List<Models> _ModelsList = new List<Models>();
    private LogsProvider _LogsProvider = new LogsProvider();

    public TrainModelsForm() {
      InitializeComponent();
      DataLoad();
    }

    // Метод, який викликається при натисканні кнопки "Open"
    private async void OpenBtn_Click(object sender, EventArgs e) {
      // Діалог вибору папки зі зображеннями
      FolderBrowserDialog folderBrowserDialog = new
FolderBrowserDialog();
      if (folderBrowserDialog.ShowDialog() == DialogResult.OK) {
        // Очищуємо попередні дані в текстовому полі
        RaportTBox.Text = "";
        FileNameTBox.Text = folderBrowserDialog.SelectedPath;

        // Запускаємо тренування у фоновому потоці
        await Task.Run(() => {
          try {
            // 1) Ініціалізуємо MLContext
            mlContext = new MLContext();
            this.Invoke((Action)(() => {
              RaportTBox.AppendText("1) Створено MLContext.\r\n");
            }));

            // 2) Зберігаємо шлях до обраної папки
            _imagePath = folderBrowserDialog.SelectedPath;
            this.Invoke((Action)(() => {
              RaportTBox.AppendText($"2)" +
                $" Обрано каталог зі зображеннями:
{_imagePath}\r\n");
            }));

            // 3) Завантажуємо сирі дані зображень (ImagePath,
Label)
            var input = LoadLabeledImagesFromPath(_imagePath);
            this.Invoke((Action)(() => {
              RaportTBox.AppendText("3) Дані успішно
завантажено.\r\n");
            }));

            // 4) Перетворюємо зображення на float[] (224x224x3)
            var imageData = PreprocessImages(input);
```

```
            this.Invoke((Action)(() => {
              RaportTBox.AppendText("4) Зображення перетворено у
тензор (float[]).\r\n");
            }));

            // 5) Створюємо IDataView
            var data = mlContext.Data.LoadFromEnumerable(imageData);
            this.Invoke((Action)(() => {
              RaportTBox.AppendText("5) Створено IDataView із
підготовлених зображень.\r\n");
            }));

            // 6) Поділяємо дані на тренувальну та тестову вибірки
            var trainTestSplit = mlContext.Data.TrainTestSplit(data,
testFraction: 0.2, seed: 1);
            trainData = trainTestSplit.TrainSet;
            var testData = trainTestSplit.TestSet;
            this.Invoke((Action)(() => {
              RaportTBox.AppendText("6) " +
                "Виконано поділ на тренувальну та тестову вибірки
(80/20).\r\n");
            }));

            // 7) Створюємо конвеєр (pipeline) із перетвореннями та
тренером LbfgsMaximumEntropy
            var pipeline =
mlContext.Transforms.Conversion.MapValueToKey(
                                    outputColumnName: "LabelAsKey",
                                    inputColumnName:
nameof(ModelInput.Label))
                                .Append(mlContext.Transforms.NormalizeMi
nMax(
                                    outputColumnName:
nameof(ModelInput.ImagePixels),
                                    inputColumnName:
nameof(ModelInput.ImagePixels)))
                                .Append(mlContext.Transforms.CopyColumns
(
                                    outputColumnName: "Features",
                                    inputColumnName:
nameof(ModelInput.ImagePixels)))
                                .Append(mlContext.MulticlassClassificati
on.Trainers.LbfgsMaximumEntropy(
                                    new
Microsoft.ML.Trainers.LbfgsMaximumEntropyMulticlassTrainer.Options {
                                      LabelColumnName = "LabelAsKey",
```

```
                                    FeatureColumnName = "Features",
                                    L1Regularization = 0.1f,
                                    L2Regularization = 0.3f,
                                    MaximumNumberOfIterations = 100,
                                    HistorySize = 20
                                  }))
                               .Append(mlContext.Transforms.Conversion.
MapKeyToValue(
                                  outputColumnName: "PredictedLabel",
                                  inputColumnName: "PredictedLabel"));

            this.Invoke((Action)(() => {
              RaportTBox.AppendText("7) Створено pipeline з
нормалізацією та тренером LbfgsMaximumEntropy.\r\n");
            }));

            // 8) Навчання моделі
            var stopwatch = System.Diagnostics.Stopwatch.StartNew();
            model = pipeline.Fit(trainData);
            stopwatch.Stop();
            this.Invoke((Action)(() => {
              RaportTBox.AppendText($"8) Навчання завершено за" +
                $" {stopwatch.Elapsed.TotalSeconds:F2} с.\r\n");
            }));

            // 9) Оцінка моделі
            var predictions = model.Transform(testData);
            var metrics =
mlContext.MulticlassClassification.Evaluate(
                predictions,
                labelColumnName: "LabelAsKey",
                predictedLabelColumnName: "PredictedLabel"
            );

            // Вивід метрик
            this.Invoke((Action)(() => {
              RaportTBox.AppendText($"MicroAccuracy:
{metrics.MicroAccuracy:P2}\r\n");
              RaportTBox.AppendText($"MacroAccuracy:
{metrics.MacroAccuracy:P2}\r\n");
              RaportTBox.AppendText($"LogLoss:
{metrics.LogLoss:F4}\r\n");
            }));

            _IsModelTrain = true;
          } catch (Exception ex) {
```

```
            // Якщо виникає помилка, показуємо її в RaportTBox
            this.Invoke((Action)(() => {
              RaportTBox.AppendText($"Помилка: {ex.Message}\r\n");
            }));
          }
        });
      }
    }



    private void AddBtn_Click(object sender, EventArgs e) {
      if (IsDataCorreced()) {
        //Зберігання моделі
        string pathName = @"\teach\" + GenerateFileName() + ".zip";
        string localProj =

System.IO.Path.GetDirectoryName(System.Reflection.Assembly.GetExecut
ingAssembly().Location);
        _ModelsProvider.InsertModels(ModelsNamesTBox.Text,
pathName);
        mlContext.Model.Save(model, trainData.Schema, localProj +
pathName);
        ClearAllData();
       _LogsProvider.InsertLogs(LoginForm.CurrentUser.UsersId,
         "Було навчено модель " + ModelsNamesTBox.Text,
DateTime.Now);
        MessageBox.Show("Дані успішно збережено!");
      }
    }



    private static List<ModelInput> LoadLabeledImagesFromPath(string
path) {
      var images = new List<ModelInput>();
      var directories = Directory.EnumerateDirectories(path);

      foreach (var directory in directories) {
        var files = Directory.EnumerateFiles(directory);

        images.AddRange(files.Select(x => new ModelInput {
          ImagePath = Path.GetFullPath(x),
          Label = Path.GetFileName(directory)
        }));
      }
```

```
      return images;
    }

    private List<ModelInput>
PreprocessImages(IEnumerable<ModelInput> inputs) {
      var imageData = new List<ModelInput>();

      foreach (var input in inputs) {
        using (var bitmap = new Bitmap(input.ImagePath)) {
          var resizedBitmap = new Bitmap(bitmap, new Size(224,
224));
          var imagePixels = new float[224 * 224 * 3];

          for (int y = 0; y < 224; y++) {
            for (int x = 0; x < 224; x++) {
              var color = resizedBitmap.GetPixel(x, y);
              var index = (y * 224 + x) * 3;
              imagePixels[index] = color.R / 255f;
              imagePixels[index + 1] = color.G / 255f;
              imagePixels[index + 2] = color.B / 255f;
            }
          }
          imageData.Add(new ModelInput {
            ImagePath = input.ImagePath,
            Label = input.Label,
            ImagePixels = imagePixels
          });
        }
      }
      return imageData;
    }


    //private void PrintSchema(DataViewSchema schema) {
    //  foreach (var column in schema) {
    //    RaportTBox.Text += ($"Column: {column.Name}, Type:
{column.Type}\r\n");
    //  }
    //}


    public string GenerateFileName() {
      DateTime now = DateTime.Now;
      string fileName = string.Format("{0}_{1}_{2}_{3}_{4}_{5}",
```

```
          now.Year, now.Month, now.Day, now.Hour, now.Minute,
now.Second);

      return fileName;
    }

    private void ClearAllData() {
      _IsModelTrain = false;
      ModelsNamesTBox.Text = String.Empty;
      RaportTBox.Text = String.Empty;
      DataLoad();
    }

    private bool IsDataCorreced() {
      bool isCorrect = true;
      if (!_IsModelTrain) {
        MessageBox.Show("Неможливо зберегти дані. \r\nЩе не навчено
модель!", "Увага!");
        isCorrect = false;
      }
      if (_Validation.IsDataEntering(ModelsNamesTBox.Text)) {
        ModelsNamesValidationLbl.Text =
NamesMy.ProgramButtons.RequiredValidation;
      } else {
        ModelsNamesValidationLbl.Text =
NamesMy.ProgramButtons.ErrorValidation;
        isCorrect = false;
      }
      return isCorrect;
    }

    private void DataLoad() {
      int firstRowIndex = 0;
      if (ModelsGridView.FirstDisplayedScrollingRowIndex > 0) {
        firstRowIndex =
ModelsGridView.FirstDisplayedScrollingRowIndex;
      }
      try {
        _ModelsList = _ModelsProvider.GetAllModels();
        LoadDataInModelsGridView(_ModelsList);
        if (_selectedRowIndex == ModelsGridView.Rows.Count) {
          _selectedRowIndex = ModelsGridView.Rows.Count - 1;
        }
        if (_selectedRowIndex >= 0) {
          ModelsGridView.FirstDisplayedScrollingRowIndex =
firstRowIndex;
```

```
            ModelsGridView.Rows[_selectedRowIndex].Selected = true;
          }
        } catch (Exception ex) {
          MessageBox.Show(ex.ToString());
        }
      }

      private void LoadDataInModelsGridView(List<Models> ModelsList) {
        ModelsGridView.DataSource = null;
        ModelsGridView.Columns.Clear();
        ModelsGridView.AutoGenerateColumns = false;
        ModelsGridView.RowHeadersVisible = false;

        ModelsGridView.DataSource = ModelsList;

        if (ModelsList.Count > 0) {
          if (ModelsList[0].Message ==
NamesMy.NoDataNames.NoDataInModels) {
            DataGridViewColumn messageColumn = new
DataGridViewTextBoxColumn();
            messageColumn.DataPropertyName = "Message";
            messageColumn.Width = ModelsGridView.Width -
NamesMy.SizeOptins.MinusSizePanel;
            ModelsGridView.Columns.Add(messageColumn);
          } else {
            DataGridViewColumn DetailIdColumn = new
DataGridViewTextBoxColumn();
            DetailIdColumn.DataPropertyName = "ModelsId";
            ModelsGridView.Columns.Add(DetailIdColumn);
            ModelsGridView.Columns[0].Visible = false;

            DataGridViewColumn numberColumn = new
DataGridViewTextBoxColumn();
            numberColumn.HeaderText = "№ ";
            numberColumn.DataPropertyName = "Number";
            numberColumn.DefaultCellStyle.Alignment =
DataGridViewContentAlignment.MiddleRight;
            numberColumn.Width = NamesMy.SizeOptins.NumberSize;
            ModelsGridView.Columns.Add(numberColumn);

            DataGridViewColumn ModelsNamesColumn = new
DataGridViewTextBoxColumn();
            ModelsNamesColumn.HeaderText = "Назва моделі";
            ModelsNamesColumn.DataPropertyName = "ModelsName";
            ModelsNamesColumn.Width = 150;
            ModelsGridView.Columns.Add(ModelsNamesColumn);
```

```
          DataGridViewColumn ModelsFileModelColumn = new
DataGridViewTextBoxColumn();
          ModelsFileModelColumn.HeaderText = "Файл";
          ModelsFileModelColumn.DataPropertyName =
"ModelsFileModel";
          ModelsFileModelColumn.Width = 250;
          ModelsGridView.Columns.Add(ModelsFileModelColumn);

          DataGridViewButtonColumn IsResidesBtn = new
DataGridViewButtonColumn();
          IsResidesBtn.HeaderText = "Видалити";
          IsResidesBtn.Text = "Видалити";
          IsResidesBtn.UseColumnTextForButtonValue = true;
          IsResidesBtn.ToolTipText = "Видалити";
          IsResidesBtn.Width = NamesMy.SizeOptins.DeleteBtnSize;
          ModelsGridView.Columns.Add(IsResidesBtn);

        }
        for (int i = 0; i < ModelsGridView.Columns.Count; i++) {
          ModelsGridView.Columns[i].HeaderCell.Style.BackColor =
Color.LightGray;
        }
      }
    }

    private void ModelsGridView_CellClick(object sender,
DataGridViewCellEventArgs e) {
      if (e.ColumnIndex == 4 && ModelsGridView[0,
e.RowIndex].Value.ToString() != _ModelsList[0].Message) {
        if (MessageBox.Show("Ви дійсно хочете видалити цей
елемент?", "Видалити", MessageBoxButtons.YesNo) == DialogResult.Yes)
{

_ModelsProvider.DeleteModelsByModelsId(Convert.ToInt32(ModelsGridVie
w[0, e.RowIndex].Value.ToString()));
          DataLoad();
        }
      }
    }
  }
}



public class ModelInput {
```

```
  public string ImagePath { get; set; }
  public string Label { get; set; }

  [VectorType(224 * 224 * 3)]
  public float[] ImagePixels { get; set; }
}


public class ModelOutput {
  [ColumnName("PredictedLabel")]
  public string PredictedLabel { get; set; }

  [ColumnName("Score")]
  public float[] Score { get; set; }
}
```

ДОДАТОК Б
(обов'язковий)
СТАТТЯ ПО ЯКІЙ БУЛО ВИКОНАНО ПЕРШИЙ РОЗДІЛ

OLGA PAVLOVA, PAVLO YURKO

Khmelnytskyi National University

Analysis of biometric access control systems

The paper presents a method and a software-hardware tool for an access control system based on biometric data. The method involves the collection, processing, and verification of biometric features such as fingerprints, facial recognition, or iris scans to authenticate individuals. The system ensures secure access while minimizing the risks associated with traditional password-based security systems. The software-hardware tool integrates biometric sensors, data storage, and authentication algorithms to provide an efficient and secure means of controlling access to protected areas or resources. This approach aims to enhance security, streamline user access, and reduce the likelihood of unauthorized access or identity theft.

Keywords: biometric access control, biometric data, authentication, security system, software-hardware tool, fingerprint recognition, facial recognition, iris scan, identity protection.

О.О. ПАВЛОВА, П.П. ЮРКО

Хмельницький національний університет, Хмельницький, Україна

Аналіз систем контролю пропуску на основі біометричних даних

У статті представлено метод та програмно-технічний засіб для системи пропуску на основі біометричних даних. Метод включає збір, обробку та перевірку біометричних ознак, таких як відбитки пальців, розпізнавання обличчя чи сканування райдужної оболонки ока для аутентифікації осіб. Система

забезпечує безпечний доступ, мінімізуючи ризики, пов'язані з традиційними системами безпеки на основі паролів. Програмно-технічний засіб інтегрує біометричні сенсори, зберігання даних і алгоритми аутентифікації, щоб забезпечити ефективний та безпечний контроль доступу до захищених територій або ресурсів. Цей підхід має на меті підвищення безпеки, спрощення доступу для користувачів і зменшення ймовірності несанкціонованого доступу чи крадіжки особистості.

Ключові слова: біометричний контроль доступу, біометричні дані, аутентифікація, система безпеки, програмно-технічний засіб, розпізнавання відбитків пальців, розпізнавання обличчя, сканування райдужної оболонки ока, захист ідентичності.

Introduction

Rapid advances in technologies such as digital cameras and portable video recording devices, as well as increased demand for security, make facial recognition technology a major biometric technology. There are many applications for facial recognition, including access control using mobile identity verification devices, mobile active video surveillance systems and rapid retrieval of records from remote facial databases[13]. With the standard authentication methods, such as passwords inheriting the vulnerabilities of being easily seen or stolen, the need for a new and better method was required. Biometric authentication was introduced as the method of protecting our info in our phones by using our biometrics, because it has a lesser chance of being stolen, as it’s impossible to completely steal all of the person's biometric data since there will always be something which won’t connect with the original.

Biometric authentication is one of the most secure forms of identification that can prove who we are. This cutting-edge technology uses our unique physical traits, such as fingerprints, facial features, or DNA, to verify our identity [1]. Due to the uniqueness of human biometrics witch played a master role in degrading imposters’ attacks. Such authentication models have overcome other traditional security methods like passwords and PIN [11]. With such authentication, protecting our identity in phones or in registrations is much easier than entering the password or making keys, since with huge amounts of registrations on different sites, and emails it is always required to have a password to protect the identity of the person. But with a huge amount of passwords, it is very hard to remember all of them, so it will take time and unnecessary work to make another one. There is always the chance that by saving all passwords in the memory bank of the device, it can be accidentally deleted or can be hacked and thus increasing the risk of security to be compromised.

Some of the potential risks associated with biometric authentication include: Appropriate technical and organizational measures, data breaches, false positives and negatives, forgery, user apprehension, regulatory compliance, longevity of biometric features.

To overcome these challenges, biometric authentication should be used carefully, implement strong security practices, and ensure compliance with relevant regulations. Additionally, using multi-factor authentication (MFA), which combines biometrics with other authentication factors, can provide an extra layer of security [2].

Domain analysis

In today's digital world, electronic devices, including biometric access systems, are becoming increasingly widespread. Examples of such technologies can be seen in embedded systems used in smartphones, Global Positioning Systems (GPS) [3], and tablets. With the rapid development and extensive deployment of communication networks, millions of devices utilizing biometric data are connected to the global infrastructure.

Since users' personal data, including biometric information, may be accessible through the network, the need for protecting this data becomes critical. To ensure confidentiality and prevent unauthorized access, it is essential for access control systems based on biometric data to incorporate reliable software and hardware protection methods. This approach ensures a high level of security when using such systems in an open information environment.

Data protection methods, such as authentication and access control, are based on three key mechanisms:

(a) knowledge — information the user knows, such as passwords;

(b) tokens — physical items the user possesses, such as access cards or badges;

(c) biometrics — unique user characteristics, such as fingerprints, iris patterns, or movement dynamics [3].

The combination of these mechanisms forms multi-factor authentication, enhancing the reliability of security systems. For instance, biometric access control systems grant access to facilities or data using unique physiological traits of the user.

The integration of biometric technologies into access control systems significantly strengthens security by combining cryptographic techniques with

biometric data analysis. Such solutions ensure accurate authentication, protection against unauthorized access, and the confidentiality and integrity of information, making them indispensable in modern software and hardware-based access control systems.

The process of verifying an individual's identity using unique physical or behavioral characteristics (such as facial features, fingerprints, hand structure, iris patterns, typing style, signature, or voice traits) is called biometric authentication [4]. This system provides a significantly higher level of protection compared to traditional password-based methods.

The main advantage lies in the necessity of the user's physical presence during authentication, which greatly complicates the possibility of unauthorized access. Additionally, there is no need to remember complex passwords or cryptographic keys, as biometric characteristics are naturally unique and inseparable from the individual. The verification mechanism works by comparing the current biometric data with the previously stored template created during registration [5].

The secure storage of these biometric templates is critical to the system's overall security, as biometric data cannot be changed or updated if compromised. However, research has shown that there are methods for stealing and replicating biometric data [6; 7], and the system may be vulnerable to malicious interference at various stages of the authentication process [8].

Biometric access systems are the systems that use unique physical characteristics data, such as fingerprints, facial recognition, or iris scans to identify individuals and grant them access to restricted areas of buildings [9]. They are used to determine the specific detail with each fingerprint or the detail on the face to recognize the particular person, as each of them have their individual details that make them unique and easily identifiable from the other people.

Biometric system has several components four components such as (Fig. 1) [9]:
- Input Interface (Scanners or Sensors);

- Processing Unit;

- Database Store;

- Output Interface.

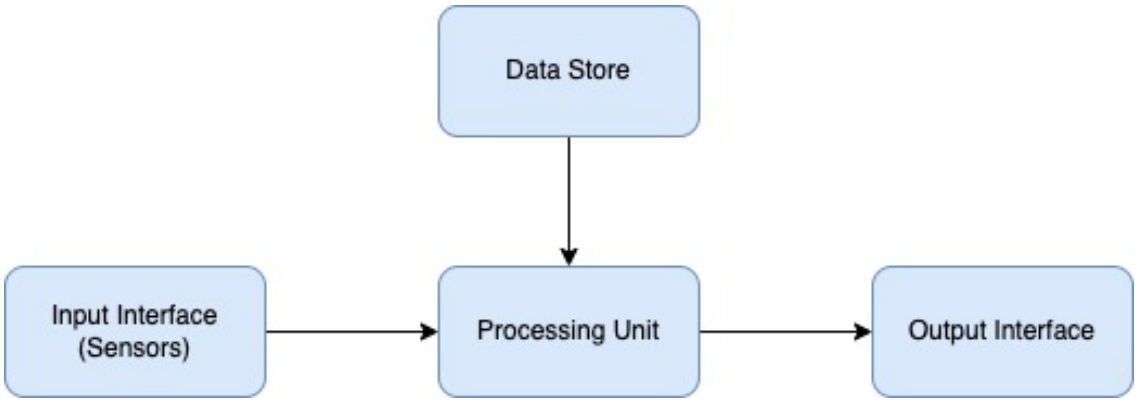


Fig.1. Biometric system components

No doubt biometric authentication increases security. However, biometrics are not immune to data breaches. If a malicious actor manages to get access to the database, then they get hold of the biometrics. This is not only a risk to the business, but it's also a risk to the identity of workers as attackers can steal their biometrics for illegitimate purposes [10].

The risks of the usage of biometric data are to be expected, as there is nothing perfect and there is no 100% guarantee that the data and confidentiality are completely protected and no one can hack them. Most of the risks include theft of biometric templates, misuse of the data by hackers or identity thieves, and even the possibility of falsification of the data also known as spoofing, which could be considered the most dangerous type of risk.

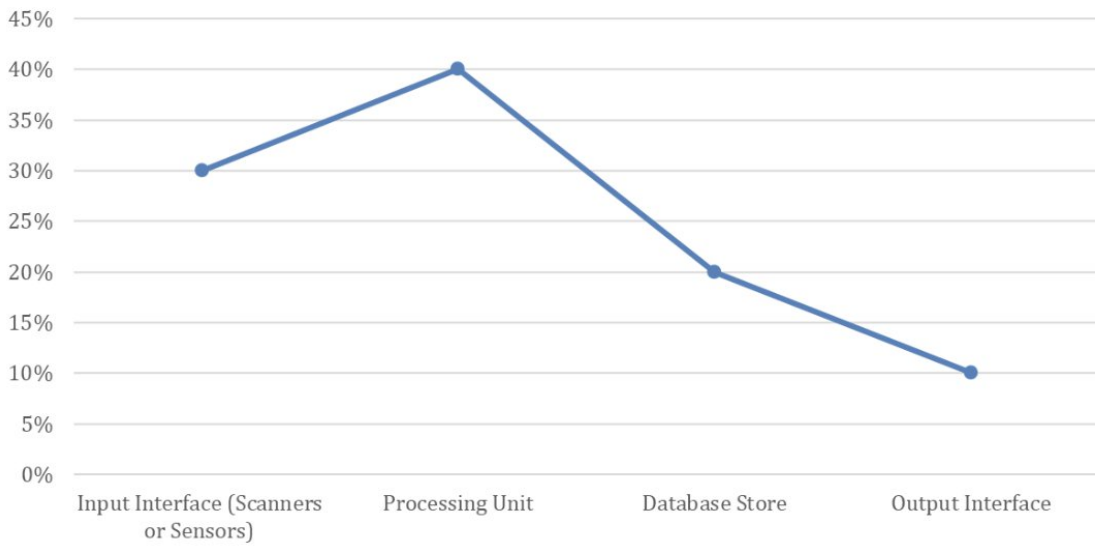


Fig 1. Distribution of Biometric System Components by Their Percentage Shares

Both security and privacy are important in the physical and digital worlds. Privacy is the right to control how the information is viewed and used, while security is protection against threats or danger. In the digital world, security generally refers to the unauthorized access of data, often involving protection against hackers or cyber criminals. Privacy consists of the person's right to manage their personal information, and security is the protection of this information. Both are equally important aspects of cyber safety. Everyone have the privacy rights and should take measures to secure their personal information and data within the digital environment [10].

Analysis of existing solutions and technologies

In today's fast-evolving digital landscape, biometric authentication systems have become increasingly prominent, providing robust security solutions. Several established technologies have been developed and deployed to enhance the security and efficiency of these systems. Here, we analyze the key existing solutions and technologies used in biometric authentication.

Fingerprint Recognition

Fingerprint recognition is one of the most widely used biometric modalities for authentication. It involves scanning the ridge patterns of a user's finger and comparing them against a stored template in the database. Many modern mobile devices and security systems use fingerprint scanners embedded in touchscreens or external sensors. Technologies like capacitive and optical scanners are commonly used in fingerprint recognition, offering quick and reliable identification. Despite its advantages, fingerprint recognition can be prone to spoofing through artificial fingerprints.

Facial Recognition

Facial recognition technology analyzes the unique features of a person's face, including the distance between eyes, nose shape, and overall facial structure. This form of biometric identification is increasingly employed in security systems such as smartphones, government identification programs, and surveillance cameras. 3D facial recognition and infrared sensors have advanced the robustness of this technology, improving accuracy even in low-light conditions. However, issues related to privacy, accuracy, and spoofing (e.g., using photos or videos to deceive the system) persist.

Iris Recognition

Iris recognition technology is based on the unique patterns in the colored part of the eye, providing a high level of security due to its uniqueness and stability. Unlike fingerprints or faces, the iris does not change over time, making it a reliable means of biometric identification. While iris recognition systems are accurate and fast, they are generally more expensive to implement and less commonly found in consumer devices compared to fingerprint or facial recognition systems. Despite the higher cost, iris recognition remains a favored choice for high-security applications, such as in government and military installations.

Table 1:

Comparison of Biometric Authentication Technologies

Biometric Technology	Description	Advantages	Challenges
Fingerprint Recognition	Scans the ridge patterns on a person's finger and compares them to stored templates. Used in mobile devices and security systems.	Widely available Fast and reliable Low-cost technology	Prone to spoofing (e.g., using artificial fingerprints) Can be less effective with damaged or worn fingerprints
Facial Recognition	Analyzes features of the face, such as distance between eyes, nose shape, and structure. Used in smartphones and surveillance systems.	Non-intrusive Fast and convenient Works in various environments (including low-light conditions with advanced tech like infrared)	Privacy concerns Spoofing with photos/videos Accuracy can be affected by facial changes or angles
Iris Recognition	Scans the unique patterns in the iris, providing high security due to the iris's stability over time. Typically used in high-security applications.	Extremely accurate Extremely accurate Stable over time Very difficult to spoof	High cost Less commonly used in consumer devices Can be less convenient (requires close proximity)

Biometric authentication technologies offer various advantages, depending on the use case. **Fingerprint recognition** is widely used due to its affordability and reliability, though it has vulnerabilities related to spoofing and fingerprint wear. **Facial recognition** is gaining popularity for its non-intrusiveness and versatility, but privacy concerns and the potential for spoofing are significant drawbacks. **Iris recognition** offers high accuracy and robustness against spoofing but is less accessible due to high costs and less convenience.

Definition of Similarity of Biometric Samples

Methods for comparing biometric templates typically involve calculating the similarity between two vectors that contain biometric data. Various mathematical models can be used for this.

Correlation:

One approach for comparing templates is calculating the correlation between two biometric data vectors is represented by the formula 1:

(1)

$$\text{Correlation } X, Y = \sum i=1n Xi-\bar{X} Yi-\bar{Y} \sum i=1n (Xi-\bar{X})2 \sum i=1n (Yi-\bar{Y})2$$

Where Xi and Yi are the elements of the vectors X and Y (biometric samples), and bar $\bar{X}$ and $\bar{Y}$ are the mean values for each sample.

Euclidean Distance Method:

Another approach is using Euclidean distance, represented by the formula 2 which defines how similar two biometric templates are.

(2)

$$D,\ X, Y = i=1nXi=Yi2$$

Where D(X, Y) is the distance between two biometric templates, indicating their similarity.

Biometric Authentication:

The authentication process involves checking the similarity between the current biometric data and the stored templates by the formula 3:

(3)

$$S = \text{Similarity}(X, Y)$$

Where S is the result of the comparison, indicating the level of similarity between the two biometric samples (from 0 to 1).

If S Threshold, authentication is successful, and access is granted to the user.

Calculation of the Probability of Successful Authentication and Error Rates are calculated by the formulas 4 and 5.

False Accept Rate (FAR):

(4)

$$\text{FAR} = \text{Number of False Acceptances} \text{Total Number of Rejestions}$$

False Reject Rate (FRR):

(5)

$$\text{FRR} = \text{Number of False Rejestions} \text{Total Number of Acceptancer}$$

Algorithms for Improving Accuracy

Filters are applied to biometric data samples to reduce noise, which enhances the accuracy of readings. This noise reduction is important for ensuring that the biometric system captures the most accurate and clean data possible. Various types of filters, such as median filters or Gaussian filters, can be used to smooth out unwanted variations and artifacts, making the data more reliable for identification.

To ensure that biometric templates are stored accurately, methods like adaptive encoding and image processing are used. These techniques reduce data loss during the storage process, helping preserve the fidelity of the original biometric data. Adaptive encoding adjusts the way the data is encoded based on the characteristics of the sample, ensuring that relevant features are preserved while minimizing storage requirements.

Image processing methods, such as contrast enhancement and edge detection, can also be applied to improve the quality of biometric samples before they are stored, ensuring higher accuracy in the later comparison stages. Biometric parameters differ in the cost, terms of efficiency and application. Differences in each parameter are presented in Table 2.

Table 2.

Analysis of Main Biometric SKUDs

Biometric Parameter	Device Cost (USD)	False Acceptance Rate (FAR), %	Advantages	Disadvantages	Applicability for Detecting Authorized Operator Impersonation
Fingerprint	100	0.001	High reliability. Resistance of the parameter. Small identification code. Compact reader. Low cost. Use of additional sensors (temperature, pressure).	Direct contact with the device. Complex algorithms. Easy to damage the fingerprint pattern. Quality depends on skin condition. Possibility of fingerprint forgery.	Used in mice, keyboards, laptops, mainly for authentication. Difficult to detect impersonation due to the need for continuous finger contact with the device.
Iris	>500	0.00001	Parameter resistance. High accuracy. Extremely difficult to fake. No direct contact with the	Complex algorithms. High cost. Low availability of high-resolution solutions. Limited by eye alignment and	Difficult to apply for continuous monitoring, requires specific eye positioning towards the camera with small scanning

			device. High speed. Can be scanned from a distance.	scanning angle.	angles.
Hand Geometry	>600	0.2	Parameter resistance. Simple algorithms.	Direct contact with the device. Inconvenient scanning procedure. Large size of the reader.	Continuous monitoring is impossible if the operator's hand is out of the scanner's range.
Retina	4000	0.000001	Unchanging over time. High accuracy. No direct contact with the device.	Difficulty in reading. Complex algorithms. High processing time for templates. High system cost.	Not applicable due to the need for specific conditions for reading the characteristic.
Face Geometry	>100	0.0047	Continuous authentication possibility. No direct contact with the device. Low cost.	Dependent on lighting conditions, head position. Sensitive to facial expressions. Sensitive to obstructions (glasses, hats, hairstyle changes).	Applicable for continuous monitoring, but with certain limitations due to the method's disadvantages.
Hand Veins	>300	0.0008	High accuracy. No direct contact with the device. Hidden characteristic.	Sensitivity to natural and artificial lighting. The characteristic depends on the state of the circulatory system.	Continuous monitoring is not possible if the operator's hand is out of the scanner's range.

Currently, there are many methods and approaches to facial recognition, each with its own characteristics and features. However, the fundamental principle of facial recognition remains common across all methods.
The facial recognition algorithm involves creating a biometric model of the face for subsequent analysis and identification.
Typically, the structure of a facial recognition system consists of three main stages: the first is acquiring data about the face, the second is extracting distinguishing features, and the third is the actual recognition process. To do this, the object is fed into the system for identification, after which the face image is processed to extract key features, which are then used for verifying the identity [9].

Let's consider an example of such a process in a real-world case, where facial recognition methods are used in modern security systems.

However, upon further examination, the method of facial recognition consists of five key steps:

1. Face Detection.

The primary function of this step is to detect a face in the captured image. The face detection process essentially checks whether there is a face in the image. Once the face is identified, the result is passed on to the next step, which is preprocessing.

2. Preprocessing.

This step serves as the initial processing stage for facial recognition. During preprocessing, unwanted noise, blurring, varying lighting conditions, and shadow effects are removed using appropriate techniques. Once the image is smooth and clear, it is then ready for the feature extraction process.

3. Feature Extraction.

In this stage, facial features are extracted using a feature extraction algorithm. This process helps to condense information, reduce the image size, enhance brightness, and eliminate noise. After this step, the facial fragment is typically transformed into a fixed-dimensional vector or a set of points with their corresponding locations.

4. Face Recognition.

Once feature extraction is complete, the system analyzes the representation of the face. The extracted feature vector of the input face is compared with the stored faces in the database. If a match is found with sufficient confidence, the identity of the face is recognized; otherwise, the system indicates an unknown face [10].

The geometric method of facial recognition is one of the earliest approaches in this field. It involves detecting key points on the face, such as the corners of the mouth, eyes, and the tip of the nose, and using them to create a set of features. These features help identify a person by using geometric lines formed between the identified points (Fig. 2).

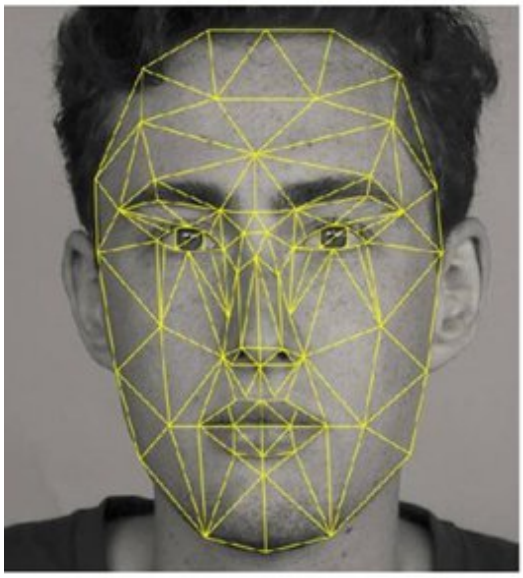


Fig. 2. Example of constructing geometric lines on a human face

The advantages of this method include the low cost of equipment and the ability to recognize faces from considerable distances. However, its drawbacks include high lighting requirements and the necessity for a frontal view of the person.

The method of elastic graph matching was also analyzed. It is based on comparing graphs that represent a facial image. These graphs consist of vertices and edges that describe the key facial features and their relationships. During recognition, one graph is fixed as the reference, while others deform to closely match the reference graph. This approach effectively handles variations such as changes in facial expressions, head movements, or distortions, making it a reliable method for face recognition (Fig.3).

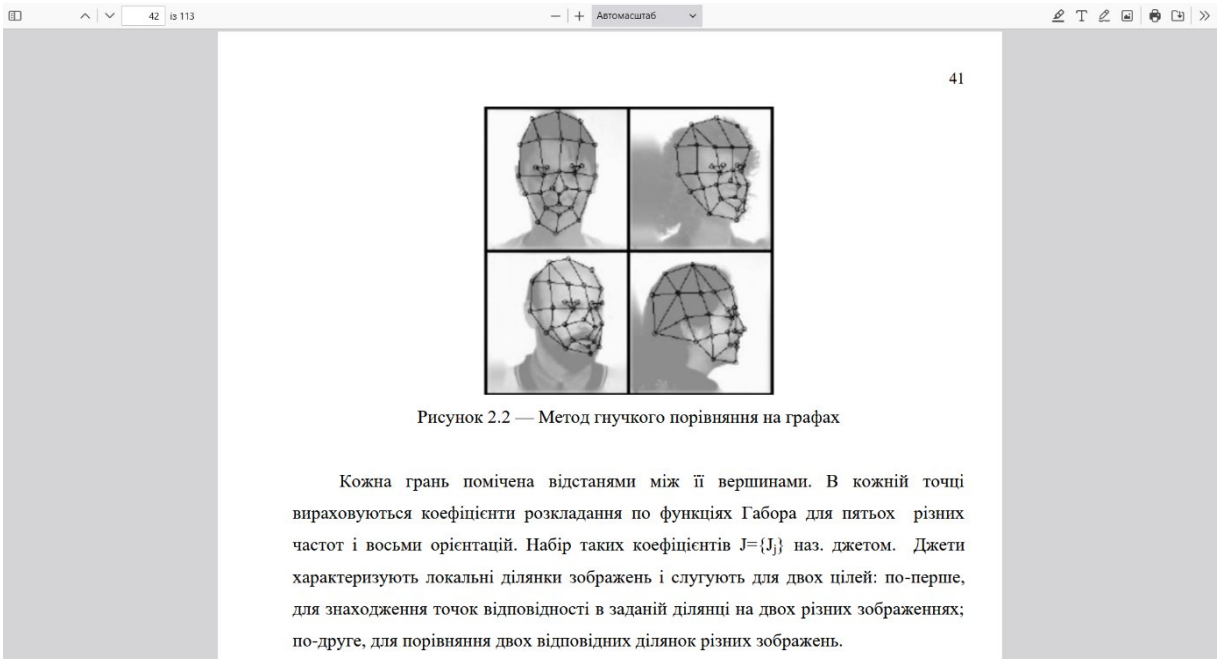


Fig. 3. Elastic Graph Matching Method

Each edge is defined by the distances between its vertices. For each point, the coefficients of the decomposition using Gabor functions with five different frequencies and eight orientations are calculated. This set of coefficients, J = {Jj}, is called a "jet." Jets describe local regions of the image and serve two main purposes: first, to find corresponding points in a given area on two different images; second, to compare the corresponding regions of different images. Each coefficient Jj = ajexp(iϕj) for points within the same region of different images is characterized by the amplitude aj, which changes gradually with the position of the point, and the phase ϕj, which varies at a rate proportional to the frequency of the wavevector of the basis function. In the simplest case, when searching for a point with similar characteristics in a new image, the phase is not considered in the similarity function.

The similarity function with a single jet at a fixed position and variable position is smooth enough to ensure fast and reliable convergence during the search using simple methods like gradient descent (GD). More advanced similarity functions incorporate phase information. For different angles, corresponding key points are manually marked in the training set. Additionally, to represent various variations of the same person's image in a single graph, multiple jets are used for each point, corresponding to different local characteristics of that point, such as open and closed eyes. The main advantage of this method is its low sensitivity to changes in lighting and facial angle [10].

There is also the Viola-Jones method, which is based on several key principles:

- It uses images in an integral form, allowing for quick computation of required objects.

- Haar features are used to search for the necessary objects.
- Boosting is applied to select the most suitable features in a given area of the image.
- The features are passed to a classifier, which outputs the result as either "True" or "False."
- Cascade features are used for quickly discarding windows where no face is found.

The algorithm works as follows: an image containing the desired objects is given. It is represented as a two-dimensional matrix of pixels with dimensions w*h, where each pixel has a value from 0 to 255 for grayscale images or from 0 to 255^3 for color images. The result of the algorithm is to detect the face and its features in the image, with the search carried out in the active region using rectangular Haar features. These features are used to describe the found faces and their characteristics: rectanglei={x, y, w, h, a}, where x, y are the coordinates of the center of the i-th rectangle, w is the width, h is the height, and a is the angle of the rectangle relative to the vertical axis of the image.

Fig. 4. Haar primitives

LeNet5 is a classic neural network architecture proposed by LeCun, originally designed for handwritten digit recognition. It consists of seven layers, with 60,000 learnable parameters and 345,308 connections. The reduction in the resolution of feature maps is achieved using subsampling layers. In a 2x2 subsampling filter network, the number of feature maps in a layer is halved, but it retains the same number of feature maps as the previous convolutional layer. LeNet5 accepts raw input images of

size 32x32 pixels. It consists of three convolutional layers (C1, C3, C5), two subsampling layers (S2, S4), one fully connected layer (F6), and an output layer. The output layer is an RBF (Radial Basis Function) layer with 10 units for classification into 10 classes. The LeNet5 architecture can be applied to biometric data recognition in access control systems, where biometric features are used for user authentication.

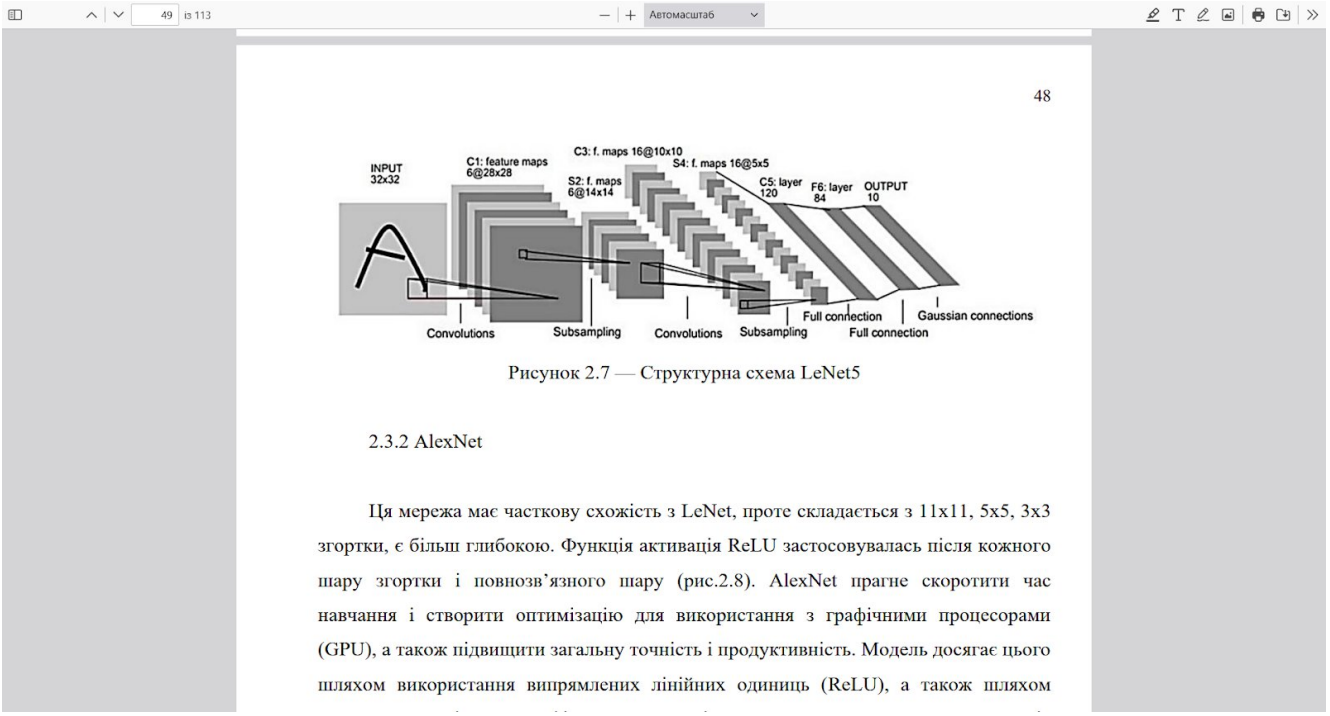


Fig. 5. Architecture of LeNet5

AlexNet is similar to LeNet but features deeper architecture with convolutional layers of sizes 11x11, 5x5, and 3x3. ReLU activation function is applied after each convolutional and fully connected layer (Fig. 6). The network aims to reduce training time and optimize performance for GPU usage, while also improving accuracy and overall efficiency. It achieves this by utilizing Rectified Linear Units (ReLU) and incorporating multiple GPUs. The introduction of these methods allowed AlexNet to significantly cut down training time and reduce errors, even with an increase in dataset size.

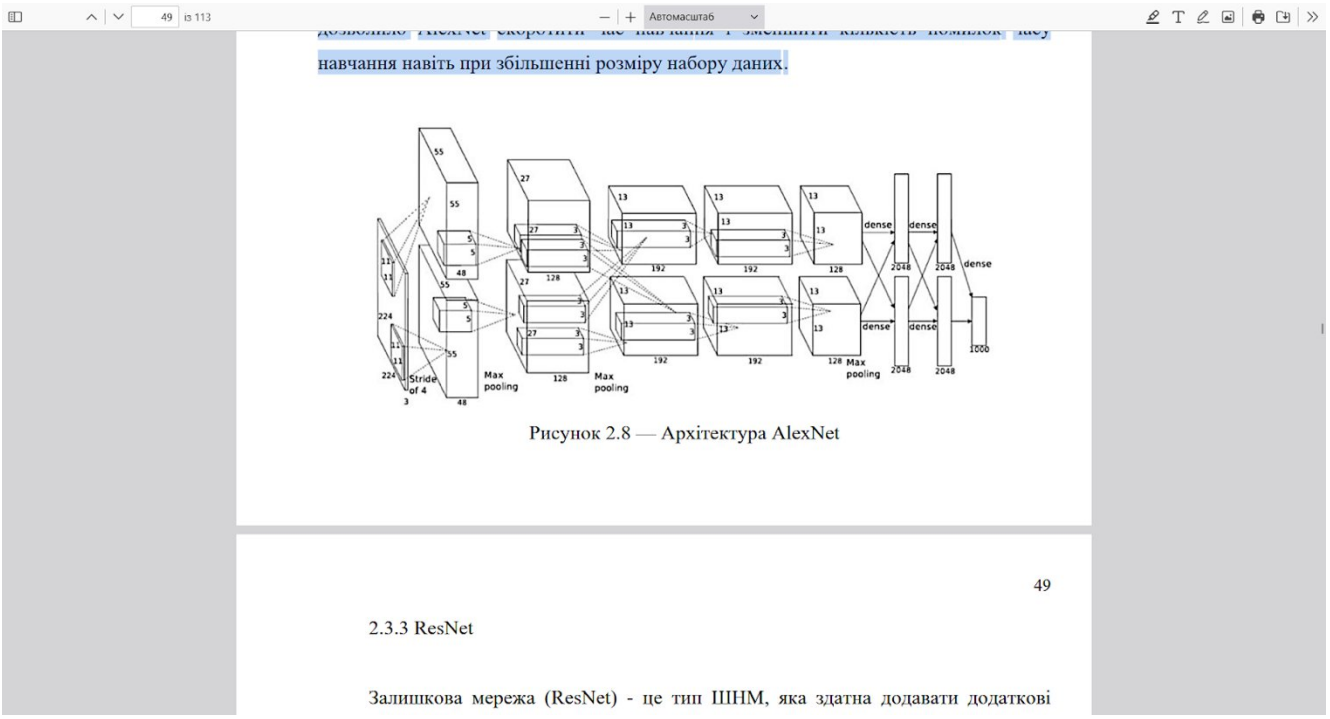


Fig. 6. Architecture of AlexNet

Residual Network (ResNet) is a type of CNN that can add extra layers to improve performance and accuracy. The added layers are capable of learning increasingly complex features, which correlates with better overall system performance and significantly improved image classification accuracy.

Practice shows that for average users who apply biometric identification and authentication systems, the convenience of using these tools is crucial. This involves not only the speed and simplicity of the procedure but also the ability to use existing equipment. Most experts agree that among various recognition methods, such as fingerprint, iris, or face recognition, three main methods are chosen based on the specific task. Today, facial recognition provides the optimal balance between authentication reliability, cost, and usability, which explains the rapid development and widespread adoption of such technologies.

A study of biometric access control and management systems was conducted. These systems are based on the recognition of physiological and behavioral characteristics of a person. The systems are classified depending on the type of characteristic they recognize.

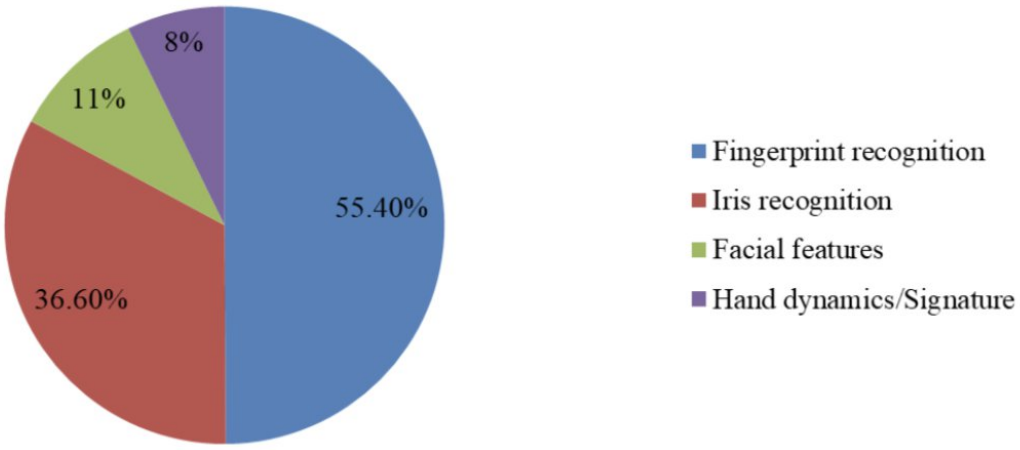


Fig.7. Distribution of Biometric Authentication Methods by Popularity

Based on the conducted analysis, it can be concluded that biometric systems using fingerprints, iris patterns, hand geometry, vein structure, and facial geometry have significant limitations in detecting the substitution of a legitimate user. These systems require specific conditions for scanning and may be ineffective for continuous monitoring. In particular, iris biometrics do not allow for continuous observation, as they also require specific conditions for scanning.

Therefore, for effective user substitution detection, it is most appropriate to use biometric characteristics that manifest during tasks the user typically performs. One of the most suitable options for continuous monitoring is keystroke dynamics, as this behavioral biometric most accurately reflects the individual traits of a user during computer interaction, particularly when typing or using a mouse.

Conclusion

In recent years, biometric technology has been vigorously promoted globally to enhance security in information technology (IT) and promote the development of emerging industries. Although biometric technologies have been employed in particular fields for a long time, they have gradually gained popularity to enhance the security of consumers and consumer electronics [12]. As a result of the analysis of modern biometric technologies and authentication methods, several important conclusions can be made. Biometric authentication is one of the most reliable and effective ways of identification, as it uses unique physical or behavioral characteristics of the user, making unauthorized access more difficult. Technologies such as fingerprint, facial, and iris recognition have their advantages and disadvantages, but combining these methods in multi-factor authentication provides an additional layer of security.

Despite their high reliability, biometric systems are not entirely immune to attacks and threats, such as theft of biometric templates or data forgery. Therefore, it is important to implement proper security measures and comply with privacy and security regulations. Given the convenience of using biometric technologies, especially facial recognition, their popularity and development are growing, opening new opportunities for improving the protection of personal data in various fields.

Overall, biometric authentication is a promising direction for ensuring security in the digital environment, but it is necessary to continually improve protection against potential threats and maintain a balance between security and user convenience.

References

1. Innovatrics, an EU-based provider based on biometric solutions. URL: https://www.innovatrics.com/glossary/biometric-authentication/ (Last accessed January 23, 2025)
2. Sumsub blog. URL: https://sumsub.com/blog/biometric-authentication-benefits-risks/ (Last accessed January 09, 2025)
3. Sangeetha, T., Kumaraguru, M., Akshay, S., & Kanishka, M. (2021, May). Biometric based fingerprint verification system for ATM machines. In *Journal of Physics: Conference Series* (Vol. 1916, No. 1, p. 012033). IOP Publishing.
4. Security Gallargather. URL: https://security.gallagher.com/en/Blog/An-Introduction-to-Biometric-Access-Control (Last accessed January 07, 2025)
5. Okta. URL: https://www.okta.com/identity-101/privacy-vs-security/ (Last accessed January 23, 2025)
6. Bio connect. URL: https://bioconnect.com/2024/08/01/6-ways-to-ensure-compliance-with-biometric-data-regulations/ (Last accessed January 07, 2025)
7. Terranova Security. URL: https://www.terranovasecurity.com/blog/hacking-biometrics (Last accessed January 09, 2025)
8. Analysis of Different Face Recognition Algorithms. URL: https://www.ijert.org/research/analysis-of-different-face-recognition-algorithms-IJERTV3IS111235.pdf (Last accessed January 09, 2025)
9. Face Recognition Using Neural Network: A Review. URL: https://www.researchgate.net/publication/301727666_Face_Recognition_Using_Neural_Network_A_Review (Last accessed January 09, 2025)
10. Face Recognition Methods: A Brief Overview. URL: http://itcm.comp-sc.if.ua/2017/Holubiak.pdf (Last accessed January 09, 2025)
11. A comprehensive survey on the biometric systems based on physiological and behavioural characteristics. URL: https://www.sciencedirect.com/science/article/abs/pii/S2214785321048513 (Last accessed January 27, 2025).
12. Exploring biometric identification in FinTech applications based on the modified TAM. URL: https://link.springer.com/article/10.1186/s40854-021-00260-2 (Last accessed January 27, 2025).

13. An Ensemble Approach To Face Recognition In Access Control Systems. URL: https://journals.riverpublishers.com/index.php/JMM/article/view/24241 (Last accessed January 27, 2025).

Павлова Ольга Олександрівна – доктор філософії, завідувач кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет, Хмельницький, Україна.

Юрко Павло Петрович – студент кафедри комп'ютерної інженерії та інформаційних систем, Хмельницький національний університет, Хмельницький, Україна.

Olga Pavlova – PhD, Associate professor of Computer Engineering & Information Systems Department, Khmelnytskyi National University, Khmelnytskyi, Ukraine,

e-mail: pavlovao@khmnu.edu.ua

orcid.org/0000-0001-7019-0354, Scopus Author ID: 57218181382

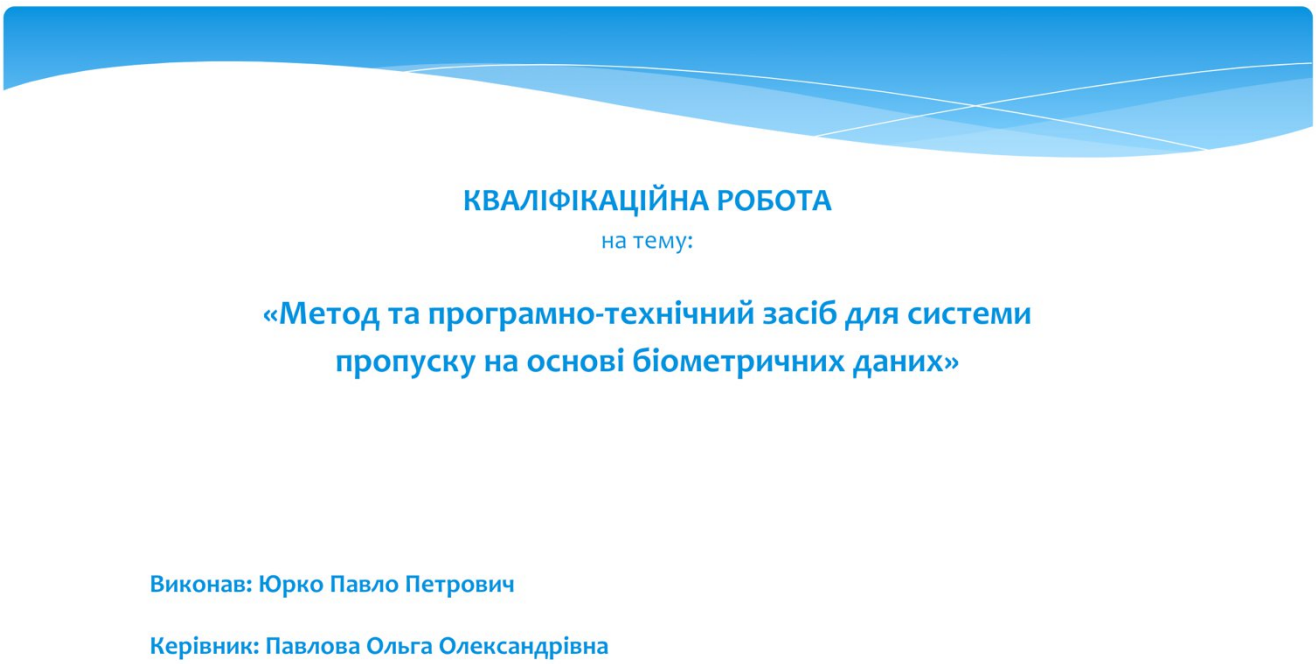
https://scholar.google.com.ua/citations?user=sQfkv30AAAAJ&hl=uk&authuser=1

Pavlo Yurko – Student of Computer Engineering & Information Systems Department, Khmelnytskyi National University, Khmelnytskyi, Ukraine,

e-mail: pavel.yurko.7654@gmail.com

ДОДАТОК В
(обов'язковий)

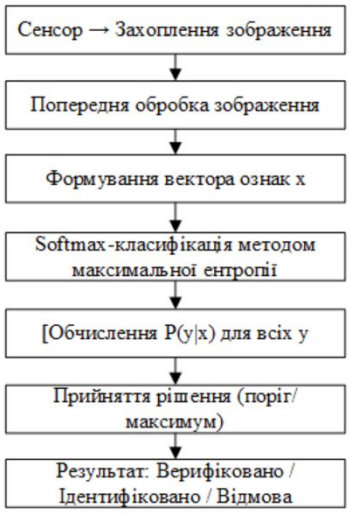
ПРЕЗЕНТАЦІЯ ДО ЗАХИСТУ



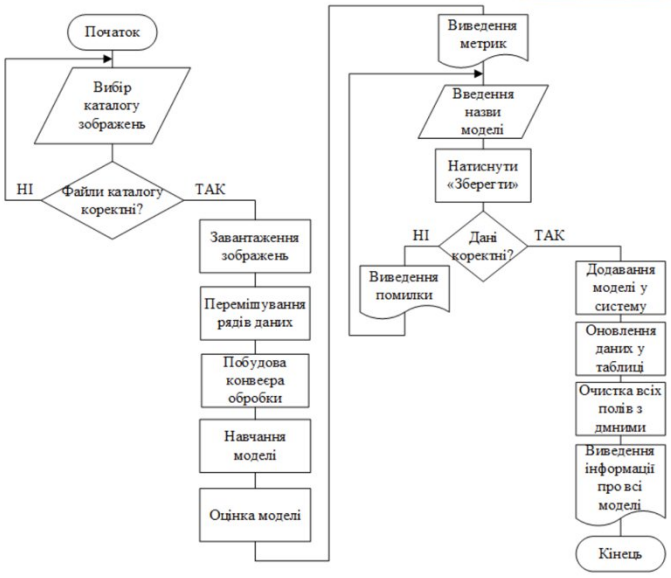
Інструменти розробки:

- мова програмування C#;
- платформа .NET Framework 4.7;
- бібліотека машинного навчання ML .NET;
- середовище розробки MS Visual Studio 2022;
- система управління базами даних MS SQL Server 2019;
- інструмент автоматичного тестування коду MSTest.

Структурна схема методу ідентифікації особи



Алгоритм навчання моделі для розпізнавання обличчя людини



Метод обробки похибок при розпізнаванні рис обличчя

У біометричній системі на основі аналізу зображення обличчя користувачеві надається дозвіл на вхід або ж відмовляється у доступі. Під час цього процесу можуть виникнути наступні фундаментальні помилки:

– хибний допуск (False Acceptance): система приймає сторонню особу, яка не має прав на доступ;

– хибне відхилення (False Rejection): система відмовляє у доступі легітимному користувачеві, що справді має право доступу.

False Acceptance Rate (*FAR*) відображає частку випадків, коли система помилково видала дозвіл на вхід стороннім особам порівняно зі всіма відмовами. Формалізовано це подається у вигляді:

$$FAR = \frac{Number\ of\ False\ Acceptances}{Total\ Number\ of\ Rejections}, \quad (3.4)$$

де:

- *Number of False Acceptances* – кількість помилкових позитивних спрацювань (наприклад, система визнала неавторизовану людину як законного користувача);

- *Total Number of Rejections* – загальна кількість випадків, коли біометричний алгоритм «відхилив» доступ.

Чим менше це відношення, тим рідше система ризикує пропустити сторонніх, що посилює безпеку.

Зворотньою проблемою є *False Reject Rate* (*FRR*), який розкриває, скільки відмов система видала несправедливо щодо власних зареєстрованих користувачів. Алгебраїчно його описують так:

$$FRR = \frac{Number\ of\ False\ Rejections}{Total\ Number\ of\ Acceptances}, \quad (3.5)$$

де:

- *Number of False Rejections* – показує, скільки разів законні користувачі були «відкинуті» системою;

- *Total Number of Acceptances* – загальна кількість успішних входів, включно з усіма випадками, де користувач отримав доступ (як справжні, так і випадково пропущені імпостери, але зазвичай акцент роблять на відношенні саме до прийняття законних користувачів).

Математична модель

Нехай кожне вхідне зображення має роздільну здатність 224×224 і 3 колірні канали (RGB). Тоді, позначивши індекс пікселя за двома координатами (i,j) і каналом c, можна подати зображення у вигляді векторизованого тензора:

$$x = (x_1, x_2, \dots, x_d) \in R^d \quad (3.6)$$

де d=224×224×3. Елемент x_k відповідає інтенсивності (нормалізованому значенню) каналу c для пікселя з координатами (i,j).

Оскільки вихідні кольорові значення R,G,B звичайно знаходяться в діапазоні [0,255], проводять їх масштабування до [0,1]. Формально для кожного пікселя (до векторизації) можна записати:

$$x_{\text{НОРМ}}(i,j,c) = \frac{x_{\text{ПОЧ}}(i,j,c)}{255}. \quad (3.7)$$

Після цього зображення вважається попередньо підготовленим, а результуючий вектор має вигляд:

$$\hat{x} = (\hat{x}_1, \hat{x}_2, \dots, \hat{x}_d). \quad (3.8)$$

Припустимо, що є набір $\{(\hat{x}_n, y_n)\}_{n=1}^{N}$, де $\hat{x}_n$ – нормалізований вектор пікселів n-го зображення, а y_n – відповідна мітка класу (наприклад, ідентифікатор особи). Для забезпечення коректної оцінки якості частину даних виділяють у тестову вибірку. Розбиття можна формально подати так:

$$D = D_{train} \cup D_{test}, D_{train} \cap D_{test} = \emptyset, \quad (3.9)$$

де D_{train} – навчальна множина, а D_{test} – тестова множина (наприклад, у співвідношенні 80% до 20%).

З метою зручності реалізації алгоритмів багатокласової класифікації кожну початкову мітку y перетворюють у ключ-ідентифікатор (LabelAsKey):

$$y \mapsto k_y \in \{1,2, \dots, K\}, \quad (3.10)$$

де K – загальна кількість різних осіб (класів) у системі, а k_y – ціле число, яке відповідає оригінальній мітці y.

Для навчання моделі використовується вектор $\hat{x}$ без додаткових перетворень або з мінімальними змінами (наприклад, масштабуванням). У найпростішому випадку вектор ознак f є копією нормалізованого вектора пікселів:

$$f = (f_1, f_2, \dots f_d) = \hat{x}. \quad (3.11)$$

Якщо додатково застосовувати нормалізацію ℓ_2 або MinMax, формально це можна описати як:

$$f_i = \frac{\hat{x}_i - min(\hat{x})}{max(\hat{x}) - min(\hat{x})} \text{ (для кожного } i). \quad (3.12)$$

Нехай θ є вектором параметрів класифікатора. Тоді імовірність належності зразка f до класу k можна записати як:

$$p(f, \theta) = \frac{exp(\theta_k^T f)}{\sum_{j=1}^{K} exp(\theta_j^T f)}, \quad (3.13)$$

де θ_k – підмножина параметрів, що відповідає класу k. У коді цей блок реалізовано через алгоритм максимальної ентропії на основі обмеженого спуску за градієнтом для багатокласової класифікації, який використовує узагальнення моделі логістичної регресії для кількох класів.

Для кожної пари (f_n, k_{y_n}) у навчальній вибірці функція втрат має вигляд:

$$l(\theta; f_n, k_{y_n}) = -\sum_{k=1}^{K} \mathbb{1}(k = k_{y_n}) \ln \ln p(k|f_n, \theta), \quad (3.14)$$

де $1(\cdot)$ – індикаторна функція (дорівнює 1, якщо $= k_{y_n}$, і 0 – в іншому випадку).

Наступним кроком регуляризація додається до функції втрат за формулою:

$$R(\theta) = \alpha \sum_{k-1}^{K} \|\theta_k\|_1 + \beta \sum_{k-1}^{K} \|\theta_k\|_2^2, \quad (3.15)$$

де $\|\cdot\|_1$ – норма L_1, $\|\cdot\|_2^2$ – норма L_2 у квадраті, а α і β – вагові коефіцієнти регуляризації (α=0.1, β=0.3).

Загальна цільова функція з урахуванням регуляризації (сума втрат для всієї навчальної множини) приймає вигляд:

$$L(\theta) = \frac{1}{N} \sum_{n=1}^{N} l(\theta, f_n, k_{y_n}) + R(\theta). \quad (3.16)$$

У процесі навчання знаходимо:

$$\theta^* = arg L(\theta). \quad (3.17)$$

Тренер тренування моделі застосовує квазі-ньютонівський метод (L-BFGS), що послідовно оновлює оцінки θ на підставі градієнтів функції $L(\theta)$. Формально ітераційний процес можна подати як:

$$\theta^{(t+1)} = \theta^{(t)} - \eta^{(t)} B_t^{-1} \nabla_\theta L(\theta^{(t)}), \quad (3.18)$$

де B_t – наближена матриця Гессіана (оновлювана на кожній ітерації), а $\eta^{(t)}$ – крок (learning rate).

Отримавши оптимальні параметри θ^*, для нового вектора ознак f_{new} вираховуємо імовірності:

$$p(f_{new}, \theta^*) = \frac{exp((\theta_k^*)^T f_{new})}{\sum_{j=1}^{K} exp((\theta_j^*)^T f_{new})}. \quad (3.19)$$

Для прийняття рішення обираємо клас $\hat{k}$ із максимальною імовірністю:

$$\hat{k} = arg p(f_{new}, \theta^*). \quad (3.20)$$

А числові значення *Score* відповідають вектору імовірностей $[p(f_{new}, \theta^*), \dots, p(f_{new}, \theta^*)]$.

Для оцінювання ефективності навчених моделей розпізнавання зображень обличчя використовуються метрики: *MicroAccuracy*, *MacroAccuracy*, *LogLoss*. Ці метрики обчислюються після завершення тренування та дають кількісну оцінку якості прийняття рішень (класифікації).

Метрика *MicroAccuracy* (мікро-точність) оцінює загальну частку правильно класифікованих прикладів з усієї вибірки без розрізнення на класи. У такий спосіб кожен випадок має однакову вагу у формуванні кінцевого відсотка коректних прогнозів.

Нехай:

- TP_i – кількість прикладів, які фактично належать класу i і були правильно віднесені до класу i (*True Positive*);
- FP_i – кількість прикладів, що належать іншим класам, але помилково зараховані до класу ii (*False Positive*);
- FN_i – кількість прикладів, що належать класу ii, проте були помилково віднесені до інших (*False Negative*).

Якщо в системі C класів, для мікро-підходу підсумовуємо ці величини по всім класам:

$$TP_{(all)} = \sum_{i=1}^{C} TP_i, FP_{(all)} = \sum_{i=1}^{C} FP_i, \; FN_{(all)} = \sum_{i=1}^{C} FN_i. \quad (3.21)$$

Тоді *MicroAccuracy* має вигляд:

$$MicroAccuracy = \frac{TP_{(all)}}{TP_{(all)} + FP_{(all)}}, \quad (3.22)$$

що еквівалентно відношенню кількості правильно класифікованих прикладів до загальної кількості передбачень. Таким чином, для мікро-точності головне – урахувати всі зразки як «однорідний набір», без зосередження на конкретних класах.

На відміну від мікро-підходу, який агрегує всю інформацію, *MacroAccuracy* (макро-точність) грунтується на почерговому обчисленні точності для кожного класу з подальшим усередненням. Це дозволяє однаково зважувати кожен клас, навіть якщо їхні розміри нерівномірні (наприклад, один клас має значно більше прикладів, ніж інший).

Для класу i точність ($Accuracy_i$) визначається через суму правильних спрацювань (TP) та правильних відхилень (TN) від загальної кількості прикладів, деяким чином пов'язаних із цим класом:

$$Accuracy_i = \frac{TP_i+TN_i}{TP_i+TN_i+FP_i+FN_i}, \quad (3.23)$$

де, TN_i – кількість прикладів, що не належать класу i і не були до нього віднесені (тобто система не помилилась з відсутністю приналежності до i).

Загальна метрика *MacroAccuracy* обчислюється як середнє арифметичне по всіх C класах:

$$MacroAccuracy = \frac{1}{C}\sum_{i=1}^{C} Accuracy_i. \quad (3.24)$$

Оскільки кожен клас робить однаковий внесок у формування фінального значення, MacroAccuracy стає особливо корисною, коли структура вибірки є незбалансованою (класів із різною кількістю прикладів).

Логарифмічна втрата (*LogLoss*) оцінює не лише те, чи правильно система класифікувала приклад, а й те, наскільки «впевнено» вона це зробила. Ця метрика вимагає, аби модель повертала повноцінний вектор імовірностей належності до кожного з C можливих класів.

Нехай $p_{i,n}$ – імовірність, із якою модель «вважає», що приклад із номером n належить класу i. У задачі багатокласової класифікації виконується:

$$\sum_{i=1}^{N} p_{i,n} = 1, \quad (3.25)$$

а для коректної оцінки потрібні цільові змінні $y_{i,n}$, що дорівнюють 1, якщо приклад n справді належить класу i, і 0 – в іншому випадку.

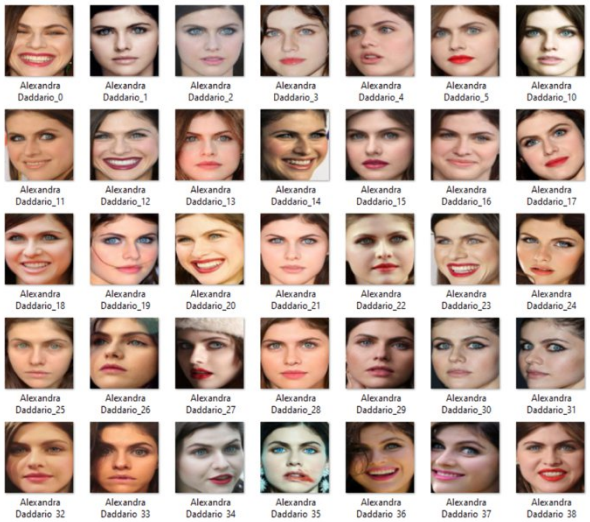
Загальний вираз *LogLoss* для N прикладів можна подати так:

$$LogLoss = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{C} y_{i,n} \cdot ln\, ln\,(p_{i,n}). \quad (3.26)$$

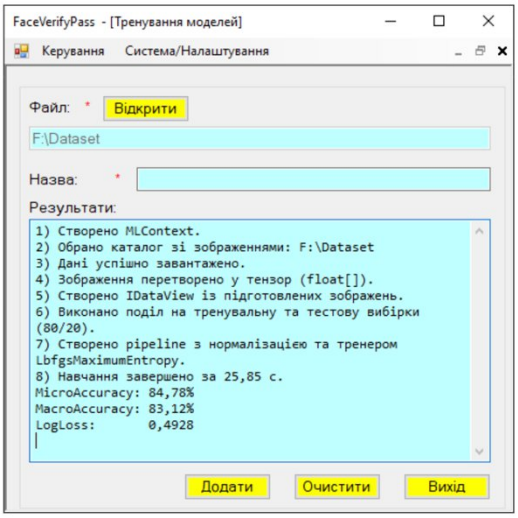
Якщо модель приписує високі імовірності справжнім класам, значення $ln\, ln\,(p_{i,n})$ буде від'ємним, але близьким до нуля, отже внесок до суми втрат стане невеликим. Натомість, якщо модель «помиляється» і ставить неадекватно низьку імовірність правильному класу, $ln\, ln\,(p_{i,n})$ набуде великих від'ємних значень, унаслідок чого *LogLoss* суттєво зростає.

Використання *MicroAccuracy*, *MacroAccuracy* та *LogLoss* у комплексі дає змогу всебічно проаналізувати поведінку алгоритму розпізнавання: від суто правильно-неправильної оцінки до збалансованості по класах і, зрештою, розгляду рівня впевненості в прогнозах. Це забезпечує повну характеристику якості системи, що вкрай важливо для надійного використання біометричних даних у реальних умовах.

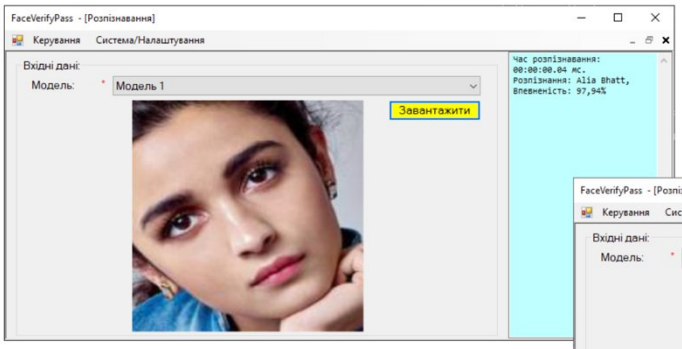
Приклад навчальних зображень з Face Recognition Dataset



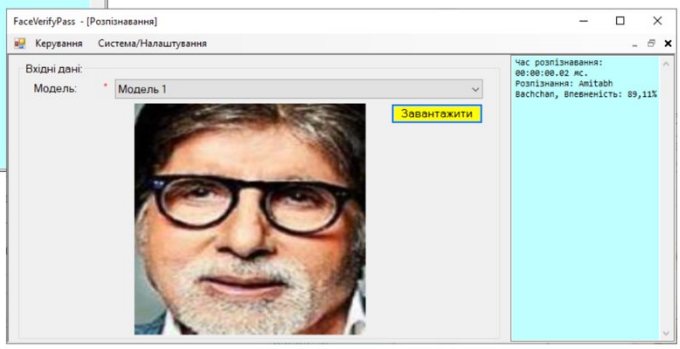
Результат навчання моделі



Результати прогнозування першого випадку



Результат прогнозування другого випадку



Порівняння стійкості до змін зовнішніх умов

Система	Зміни освітлення	Наявність аксесуарів	Зміни виразу обличчя	Чутливість до шуму
Розроблена система	Висока	Впізнає з окулярами	Стійка до змін	Середня
DeepFace	Середня	Чутлива до аксесуарів	Чутлива до змін	Висока
Amazon Rekognition	Висока	Впізнає з окулярами	Стійка до змін	Низька

Порівняння швидкості обробки

Система	Час розпізнавання (мс)	К-сть оброблених кадрів в секунду
Розроблена система	4	250
DeepFace	15	67
Amazon Rekognition	6	167

Порівняння вимог до апаратних ресурсів

Система	Мінімальні апаратні вимоги	Рекомендовані апаратні вимоги	Підтримувані платформи
Розроблена система	CPU: 2 ядра, RAM: 4 ГБ	CPU: 4 ядра, RAM: 8 ГБ	Windows, Linux
DeepFace	CPU: 2 ядра, RAM: 4 ГБ	CPU: 4 ядра, RAM: 8 ГБ	Windows, Linux, MacOS
Amazon Rekognition	Хмарне рішення	Хмарне рішення	Windows, Linux, MacOS, Android, iOS

Висновки

- проведено аналіз існуючих рішень у сфері систем пропуску та біометричної ідентифікації, що дозволило обґрунтувати доцільність використання розпізнавання облич;
- розроблено та адаптовано метод машинного навчання з використанням нейронних мереж для точного розпізнавання облич у різних умовах;
- реалізовано алгоритм функціонування системи пропуску, що охоплює збір, обробку та класифікацію біометричних даних з високою швидкодією;
- створено та протестовано програмно-технічний засіб, який продемонстрував ефективність роботи моделі у реальних сценаріях, підтверджуючи досягнення поставленої мети.

Дякую за увагу!

Завідувачу кафедри КІІС
доктору філософії, доценту
Ользі ПАВЛОВІЙ

Юрка Павла Петровича
ПІБ здобувача вищої освіти

ФІТ, 2 курсу, групи КІ2м-23-3

ЗАЯВА

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

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

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

22 квітня 2025 року

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

РЕЦЕНЗІЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

Дипломник: Юрко Павло Петрович

Тема: Метод та програмно-технічний засіб для системи пропуску на основі біометричних даних

Спеціальність: 123 «Комп'ютерна інженерія»

Обсяг дипломної роботи:

Кількість сторінок записки 102

1. Короткий зміст роботи та прийнятих рішень: Метою кваліфікаційної роботи є розробка методу та програмно-технічного засобу для системи пропуску на основі біометричних даних.

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

3. Характеристика виконання кожного розділу, ступінь використання останніх досягнень науки і техніки і передових методів роботи: У першому розділі проаналізовано існуючі програмно-технічні засоби із використанням біометричних даних, проведено огляд літературних джерел та виконано порівняльний аналіз існуючих рішень. У другому розділі проведено огляд апаратних та програмних засобів для вирішення поставленого у роботі завдання. У третьому розділі запропоновано метод та алгоритм використання біометричних даних для системи пропуску. У четвертому розділі спроектовано архітектуру та запропоновано програмну реалізацію системи пропуску на основі біометричних даних.

Наукова новизна отриманих результатів полягає у вдосконаленні існуючих методів та алгоритмів застосування біометричних даних у системах пропуску.

4. Позитивні сторони роботи: отримання одного пункту наукової новизни.

5. Негативні сторони роботи: не вказано обмеження пропонованої системи

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

7. Відгук про роботу в цілому: Робота виконана на високому науково-технічному рівні.

8. Інші зауваження: _____

9. Оцінка дипломної роботи: відмінно.

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

Бармак Олександр Володимирович, д.т.н., проф., зав. каф. комп'ютерних наук ХНУ

"28" 04 2025 р. _____(підпис)

РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ СИСТЕМ

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод та програмно-технічний засіб для системи пропуску на основі біометричних даних

Автор: Юрко Павло Петрович

Спеціальність: 123 – Комп'ютерна інженерія

Освітня програма: освітньо-наукова

Науковий керівник: Павлова Ольга Олександрівна, зав. кафедри КІІС, доктор філософії, доцент

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

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

1) запозичення розміщені в розділах є збіг зі звітом з науково-дослідної практики автора Павла Юрка " Аналіз систем контролю пропуску на основі біометричних даних", який було додано в репозиторій ХНУ 21 березня 2025 року;
2) усі запозичення фрагментарні, або мають належним чином оформленні посилання;
3) окремі виявлені збіги є загальновживаними фразами або виразами, які було використано з журналів, блогів і веб-сайтів;
4) всі зафіксовані системою ознаки модифікації тексту відносяться до комбінування латинських символів зі україномовними скороченнями індексів в формулах, що не є модифікацією тексту. Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ ідентичності/схожості StrikePlagiarism, складає 5% і адресується до 82 першоджерела; та системою Anti-Plagiarism складає 7%, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи _____ Ольга ПАВЛОВА

Гарант ОП _____ Олег САВЕНКО

Завідувач кафедри КІІС _____ Ольга ПАВЛОВА

Протокол аналізу звіту подібності експертом

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

Автор: Павло ЮРКО

Співавтор:

Назва: ЮРКО_Метод та програмно-технічний засіб для системи пропуску на основі біометричних даних

Експерт:

Підрозділ: Кафедра комп'ютерної інженерії та інформаційних систем

Коефіцієнт подібності 1:4.1%

Коефіцієнт подібності 2:0.9%

Мікропробіли: 4

Заміна букв: 1

Інтервали: 0

Білі знаки: 1

Дата створення звіту: 2025-04-24 16:41:55.0

Після аналізу Звіту подібності констатую наступне:

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

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

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

Обгрунтування:

2025-04-24 Доцент Андрій Нічепорук

Дата експерт

Fri Apr 25 06:47:30 EEST 2025, Медзатий Дмитро Миколайович, Хмельницький національний університет, ХНУ

Anti-Plagiarism v-15.274 Educational

Максимальне співпадіння з одним документом 2.0%

Словники перевірки: en_US, ru_RU, ua_UA. **Помилок в документах: 7%**

	Документ		Сумарний збіг по Базі Даних	
ID: 240472 Назва: МКР Метод та програмно-технічний засіб для системи пр[illegible]ску на основі біометричних даних Додано в БД: 2025-04-25 Автора: Павло ЮРКО Керівники: Ольга ПАВЛОВА Консультанти: Опоненти:	Символи	Лексеми	Символи	Лексеми
	163264	1245	3997 (2%)	49 (4%)

Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

Довідка

Видана Юрка П., що його стаття «Analysis of biometric access control systems» прийнята та буде опублікована у №2 фахового наукового журналу категорії Б «Computer systems and information technologies».

Головний редактор журналу Тетяна ГОВОРУЩЕНКО

24.04.2025