

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
Західноукраїнський національний університет  
Факультет комп'ютерних інформаційних технологій  
Кафедра інформаційно-обчислювальних систем і управління

**КАРНИДАЛ Володимир Сергійович**

**Програмний модуль визначення віку та статі людини на  
зображенні із застосуванням OpenCV та глибоких нейронних мереж  
/ Software Module for Image Age and Gender Detection Using OpenCV  
and Deep Neural Networks**

Спеціальність 122 – Комп'ютерні науки  
Освітньо-професійна програма – Комп'ютерні науки

Кваліфікаційна робота

Виконав студент групи КНШІ-41  
В.С. Карнидал

---

Науковий керівник:  
к.т.н., доцент П.Є. Биковий

---

Кваліфікаційну роботу допущено до  
захисту

«\_\_»\_\_\_\_\_ 2025 р.

В.о. завідувача кафедри  
\_\_\_\_\_ Н.М. Васильків

**Тернопіль – 2025**

**Факультет комп'ютерних інформаційних технологій**  
**Кафедра інформаційно-обчислювальних систем і управління**  
**Освітній ступінь «бакалавр»**  
**спеціальність 122 – Комп'ютерні науки**  
**освітньо-професійна програма – Комп'ютерні науки**

ЗАТВЕРДЖУЮ  
В.о. завідувача кафедри  
Н.М. Васильків  
« \_\_\_\_ » \_\_\_\_\_ 2024 р.

**ЗАВДАННЯ**  
**НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ**  
**КАРНИДАЛУ Володимиру Сергійовичу**  
(прізвище, ім'я, по батькові)

1. Тема кваліфікаційної роботи: Програмний модуль визначення віку та статі людини на зображенні із застосуванням OpenCV та глибоких нейронних мереж / Software Module for Image Age and Gender Detection Using OpenCV and Deep Neural Networks

керівник роботи Биковий П.Є. к.т.н., доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від 20 грудня 2024 р. № 938.

2. Строк подання студентом закінченої кваліфікаційної роботи 25 травня 2025 р.

3. Вихідні дані до кваліфікаційної роботи: завдання на кваліфікаційну роботу студента, наукові статті, технічна література.

4. Основні питання, які потрібно розробити:

- проаналізувати існуючі методи визначення віку та статі людини на зображенні;
- сформулювати концепцію побудови програмного модуля;
- розробити архітектуру програмного модуля;
- реалізувати нейромережеву модель визначення віку та статі людини на зображенні;
- провести експериментальні дослідження запропонованого модуля.

5. Перелік графічного матеріалу в роботі:

- структура програмного модуля визначення віку та статі;
- принцип роботи алгоритму HOG;
- архітектура FaceNet;
- принцип роботи підходу Multi-task Learning;
- базова архітектура CNN класифікація за статтю та віком;
- порівняльна діаграма ресурсозатратності.

## 6. Консультанти розділів кваліфікаційної роботи

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата   |                  |
|--------|-------------------------------------------|----------------|------------------|
|        |                                           | Завдання видав | Завдання прийняв |
|        |                                           |                |                  |
|        |                                           |                |                  |
|        |                                           |                |                  |

## 7. Дата видачі завдання 20 грудня 2024 р.

### КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів кваліфікаційної роботи                                                                                      | Строк виконання етапів кваліфікаційної роботи | Примітка |
|-------|--------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------|----------|
| 1     | Затвердження теми кваліфікаційної роботи, ознайомлення з літературними джерелами та складання плану роботи               | до 01.01. 2025 р.                             |          |
| 2     | Написання 1 розділу кваліфікаційної роботи                                                                               | до 01.02. 2025 р.                             |          |
| 3     | Написання 2 розділу кваліфікаційної роботи                                                                               | до 01.04.2025 р.                              |          |
| 4     | Написання 3 розділу кваліфікаційної роботи                                                                               | до 01.05. 2025 р.                             |          |
| 5     | Представлення попереднього варіанту кваліфікаційної роботи, перевірка та внесення змін керівником                        | до 15.05.2025 р.                              |          |
| 6     | Опрацювання зауважень та представлення завершеного варіанту кваліфікаційної роботи. Підготовка супроводжуючих документів | до 25.05.2025 р.                              |          |
| 7     | Перевірка кваліфікаційної роботи на оригінальність тексту                                                                | до 30.05.2025 р.                              |          |
| 8     | Оформлення кваліфікаційної роботи та отримання допуску до захисту                                                        | до 10.06.2025 р.                              |          |
| 9     | Подання кваліфікаційної роботи до захисту на засіданні атестаційної комісії                                              | до 10.06. 2025 р.                             |          |

Студент \_\_\_\_\_ Карнидал В.С.  
( підпис ) (прізвище та ініціали)

Керівник кваліфікаційної роботи \_\_\_\_\_ Биковий П.Є.  
( підпис ) (прізвище та ініціали)

## АНОТАЦІЯ

Пояснювальна записка до кваліфікаційної роботи: 76 с., 20 рис., 39 джерел.

У центрі дослідження знаходиться процес автоматизованого визначення віку та статі людини за зображенням обличчя із застосуванням методів комп'ютерного зору та глибокого навчання.

Основною метою кваліфікаційної роботи є створення програмного модуля, який дозволяє ідентифікувати обличчя на зображенні та класифікувати його за віковою групою і статтю, використовуючи згорткові нейронні мережі й засоби бібліотеки OpenCV.

Для досягнення мети використано методи глибокого навчання на основі згорткових нейронних мереж, зокрема MobileNet, а також інструменти обробки зображень, такі як каскади Хаара, DNN моделі та функціонал мови програмування Python для реалізації інтерфейсу користувача.

У результаті розроблено повнофункціональний програмний модуль із графічним інтерфейсом, який забезпечує завантаження зображення, виявлення обличчя та подальше визначення віку і статі.

Проведене тестування на основі публічних наборів даних UTKFace, Adience і IMDB WIKI підтвердило ефективність розробки: точність визначення статі перевищила 92 відсотки, а середня абсолютна похибка при оцінці віку становить близько 4 років. Розробка є придатною до застосування в системах безпеки, відеоаналітики, персоналізації сервісів і дослідницьких розробках.

Ключові слова: КОМП'ЮТЕРНИЙ ЗІР, ВИЗНАЧЕННЯ ВІКУ, ВИЗНАЧЕННЯ СТАТІ, ОБРОБКА ЗОБРАЖЕНЬ, ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, OPENCV, PYTHON, ГЛИБИННЕ НАВЧАННЯ, КЛАСИФІКАЦІЯ.

## ANNOTATION

The bachelor's thesis report: 76 p., 20 figs., 39 sources. At the center of the study is the process of automated determination of the age and sex of a person from a face image using methods of computer vision and deep learning.

The main purpose of the qualification work is to create a software module that allows you to identify a face in an image and classify it by age group and gender, using convolutional neural networks and means of the OpenCV library.

To achieve the goal, deep learning methods based on convolutional neural networks, in particular MobileNet, as well as image processing tools such as Haar cascades, DNN models and Python programming language functionality for implementing the user interface, were used.

As a result, a fully functional software module with a graphical interface has been developed that provides image loading, face detection and subsequent determination of age and gender.

Conducted testing on the basis of public datasets UTKFace, Adience and IMDB WIKI confirmed the effectiveness of the development: the accuracy of sex determination exceeds 92 percent, and the average absolute error in the assessment of age is about 4 years. The development is suitable for use in security systems, video analytics, personalization of services and research developments.

Keywords: COMPUTER VISION, AGE DETERMINATION, SEX DETERMINATION, IMAGE PROCESSING, CONVOLUTIONAL NEURAL NETWORKS, OPENCV, PYTHON, DEEP LEARNING, CLASSIFICATION.

## ЗМІСТ

|                                                                                   |    |
|-----------------------------------------------------------------------------------|----|
| ВСТУП .....                                                                       | 7  |
| 1 АНАЛІЗ ТЕХНОЛОГІЙ РОЗПІЗНАВАННЯ ВІКУ ТА СТАТІ ЛЮДИНИ .....                      | 9  |
| 1.1 Опис предметної області .....                                                 | 9  |
| 1.2 Огляд відомих рішень .....                                                    | 12 |
| 1.3 Постановка задачі дослідження .....                                           | 18 |
| 2 АЛГОРИТМІЧНЕ ТА ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ .....                                 | 20 |
| 2.1 Аналіз математичних моделей та алгоритмів .....                               | 20 |
| 2.2 Інтеграція комп'ютерного зору та глибоких нейронних мереж .....               | 24 |
| 2.3 Обґрунтування вибору запропонованих методів та моделей .....                  | 28 |
| 3 ПРОГРАМНО-ТЕХНОЛОГІЧНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ.....                                 | 33 |
| 3.1 Реалізація процесу підготовки даних .....                                     | 33 |
| 3.2 Етапи реалізації модуля .....                                                 | 36 |
| 3.3 Інтерфейс користувача .....                                                   | 42 |
| 3.3 Тестування програмного модуля .....                                           | 52 |
| ВИСНОВКИ.....                                                                     | 59 |
| СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ .....                                                  | 61 |
| Додаток А Лістинг класу для підготовки даних багатовихідної нейронної мережі..... | 65 |
| Додаток Б Основний код визначення віку та статі людей на зображеннях.....         | 66 |
| Додаток В Архітектура програмного модуля визначення віку та статі .....           | 69 |
| Додаток Г Копія опублікованих результатів.....                                    | 70 |

## ВСТУП

В той час, коли цифрові технології захоплюють основні прошарки людської діяльності, комп'ютерний зір набуває важливішого значення. Можливість автоматично аналізувати та інтерпретувати візуальні дані відкриває широкі перспективи для створення інтелектуальних цифрових систем, що підвищують ефективність процесів, покращують безпеку та створюють нові можливості взаємодії людини з технологіями. Серед різноманітних напрямків комп'ютерного зору особливу увагу привертає задача визначення віку та статі людини за зображенням обличчя, яка є частина загальної потреби біометричної ідентифікації.

Актуальність теми зумовлена постійним зростанням попиту на персоналізовані сервіси, що потребують ефективних інструментів біометричної ідентифікації, а також поширене використання штучного інтелекту у повсякденному житті. Системи, які здатні визначати вік і стать особи, знаходять застосування у таких сферах, як персоналізований маркетинг, безпека та контроль доступу, автоматизовані системи обслуговування, соціальні медіа, освітні технології, а також у медицині. При цьому забезпечується не лише адаптація сервісів до потреб конкретного користувача, але й можливість масштабного аналізу вікових та гендерних характеристик аудиторії.

Традиційні алгоритми комп'ютерного зору, виявилися недостатньо ефективними в умовах змін зовнішнього вигляду обличчя, поганого освітлення, позиції обличчя та наявності аксесуарів. Саме тому сучасні рішення все частіше створюють за допомогою глибоких нейронних мереж.

Метою даної кваліфікаційної роботи є розробка програмного модуля для автоматичного визначення віку та статі людини на зображенні з використанням інструментів OpenCV і попередньо навчених згорткових нейронних мереж.

У процесі реалізації поставленими завданнями було дослідження сучасних підходів до розв'язання задачі, розробка архітектури програмного забезпечення,

реалізація алгоритмічного забезпечення для обробки зображень та інтегрування відповідних моделей нейронних мереж, а також проведення тестування та оцінення якості роботи системи.

Об'єктом дослідження є процеси автоматичного розпізнавання віку та статі людини на зображеннях.

Предметом дослідження є програмний модуль, що реалізує вищезазначені процеси за допомогою методів комп'ютерного зору та глибокого навчання.

В ході роботи застосовуються методи дослідження, що забезпечують комплексний підхід до вирішення поставлених завдань. Системний аналіз використовується для вивчення сучасного стану проблеми визначення віку та статі на зображеннях, а також для глибокого розуміння наявних рішень та підходів. Шляхом порівняльного аналізу буде здійснено оцінку ефективності та доцільності використання різних архітектур глибоких нейронних мереж, а також інструментальних засобів. Математичне моделювання дозволить описати функціонування системи та її компонентів, зокрема принципів роботи нейронних мереж. Безпосередня розробка програмного модуля базуватиметься на принципах алгоритмізації та програмування. Експериментальні дослідження дозволять перевірити працездатність та оцінити ефективність розробленого модуля на реальних наборах даних. Особливу увагу приділено адаптації моделі до умов реального застосування, з урахуванням змін освітлення, ракурсу та якості зображення. Результати роботи можуть бути використані як у вигляді самостійного застосунку, так і як елемент складніших систем, зокрема у сфері відеоаналітики, безпеки або інтерфейсів, що враховують демографічні особливості користувача.

Результати дослідження було опубліковано на студентській науково-практичній конференції “Інтелектуальні інформаційні технології в прикладних дослідженнях” (ІПТАР – 2025).



# 1 АНАЛІЗ ТЕХНОЛОГІЙ РОЗПІЗНАВАННЯ ВІКУ ТА СТАТІ ЛЮДИНИ

## 1.1 Опис предметної області

У сучасному цифровому середовищі, де обсяг візуального контенту постійно зростає, технології комп'ютерного зору відіграють ключову роль у забезпеченні автоматизованого аналізу зображень [1]. Вони відкривають широкі можливості для створення інтелектуальних систем, які використовуються в галузях безпеки, охорони здоров'я, маркетингу, соціальних медіа та цифрових сервісах. Особливо актуальним напрямом є розпізнавання обличчя та визначення його характеристик, таких як вік і стать – ці завдання належать до важливих застосувань штучного інтелекту в реальному середовищі [2].

Проблематика полягає у складності точного визначення віку та статі людини на основі зображення, особливо в умовах змінного освітлення, нестандартного ракурсу, міміки, наявності макіяжу або аксесуарів. Багато традиційних підходів, що спираються на вручну створені ознаки, не демонструють достатньої стійкості до таких змін, що значно знижує якість розпізнавання в нестабільних умовах.

З огляду на значний приріст цифрових зображень – з камер спостереження, смартфонів, соціальних мереж – обробка вручну чи напівавтоматичними методами є малоефективною. Це обумовлює потребу в інтелектуальних системах, які можуть забезпечувати стабільні результати із високою точністю та швидкістю. Такі системи особливо важливі в контексті біометричної ідентифікації, аналітики демографічних характеристик, а також у сфері безпекових та маркетингових технологій [3].

Типовий процес розпізнавання включає такі етапи:

1. Детекція обличчя (Face Detection): визначення та локалізація області обличчя. Застосовуються як класичні алгоритми (Viola–Jones), так і сучасні методи на базі глибоких згорткових нейронних мереж, зокрема MTCNN.

2. Вирівнювання та нормалізація (Face Alignment and Normalization): приведення обличчя до уніфікованого формату шляхом стандартизації розміру та положення, що знижує варіативність на вході моделі.

3. Класифікація віку та статі (Age and Gender Classification): обробка вирівняного обличчя з використанням CNN. Визначення статі виконується як бінарна класифікація, віку – як регресійна або мультикласова задача.

4. Візуалізація результатів: відображення передбачених значень віку та статі безпосередньо на зображенні.

На рисунку 1.1 наведено функціональну структуру запропонованого програмного модуля.

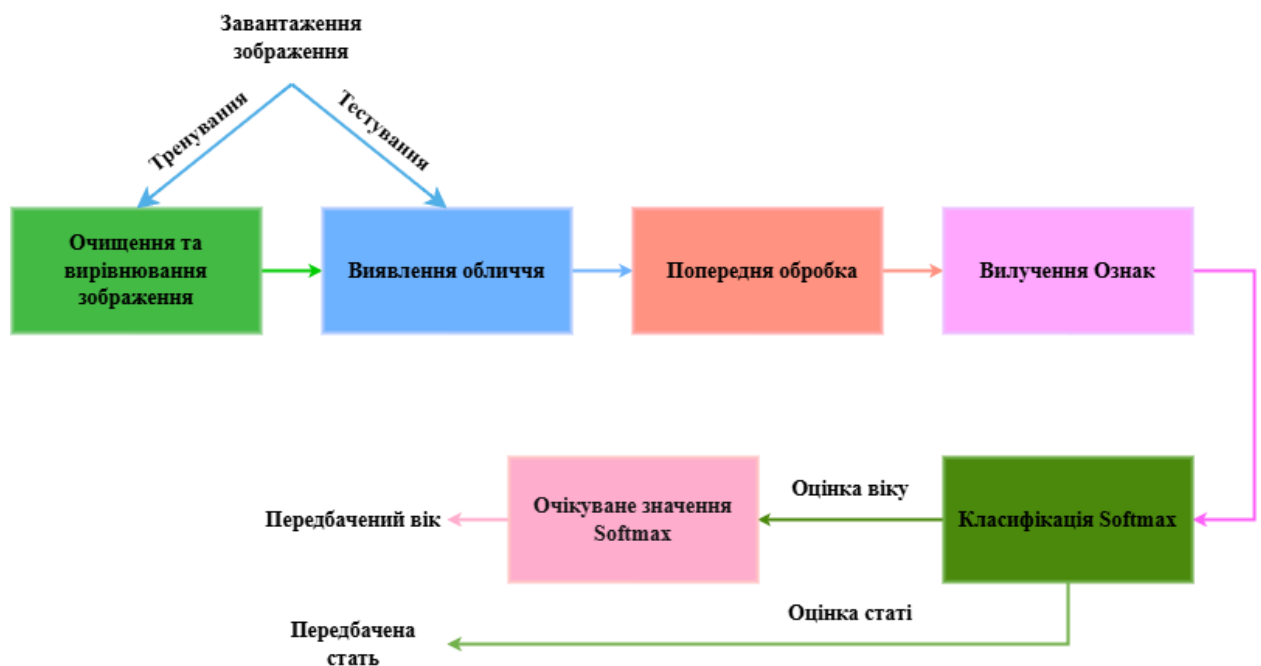


Рисунок 1.1 – Структура програмного модуля визначення віку та статі

Однак ефективність таких систем обмежується низкою викликів: зміна освітлення, вираз обличчя, нестандартні ракурси, наявність декоративних елементів, а також індивідуальні відмінності у старінні. Макіяж, борода або інші візуальні змінні можуть спотворювати ознаки, що використовуються моделлю. Ще

одним критичним аспектом є нерівномірне представлення категорій у навчальних наборах (вік, стать, етнічність), що призводить до упередженості в прогнозах.

Не менш важливим є обмеження обчислювальних ресурсів: глибокі нейронні мережі потребують значної потужності, що ускладнює використання в мобільних або вбудованих системах реального часу.

Для підвищення надійності моделей використовують великі, збалансовані та анотовані набори даних, що охоплюють різноманітні вікові, статеві та етнічні групи. Найбільш поширеними є Adience, IMDB-WIKI, UTKFace, які дозволяють тестувати моделі в стандартизованих умовах [3].

У сучасному інформаційному суспільстві розв'язання задачі ідентифікації особи за візуальними ознаками набуло актуальності через швидкий розвиток комп'ютерних технологій та набуття популярності штучними інтелектами різного рівня. Це зумовлює значне зростання сфер застосування різних алгоритмів комп'ютерного зору: безпека, комерція, медицина, освіта, розважальні та соціальні сервіси тощо. Однією з важливих підзадач таких алгоритмів є розпізнавання віку та статі особи за її зображенням з метою адаптації сервісів до потреб реального користувача.

Методи, що дозволяють здійснювати розпізнавання у режимі реального часу на звичайних камерах спостереження або на побутових веб-камерах ПК, становлять особливий інтерес для практичного впровадження. Використання можливостей відомої бібліотеки OpenCV для організації первинної обробки зображення у зв'язці з можливостями глибоких нейромереж дозволяє значно підвищити точність розв'язування такого класу задач, автоматизувати обробку результатів та полегшити розробку спеціальних програмних продуктів.

Використання можливостей згорткових забезпечує можливість ефективного використання всіляких ознак обличчя, що можуть певним чином вказувати на вік та стать його власника, без потреби у виконанні інженерного аналізу та виокремлення таких ознак. Згорткові нейронні мережі функціонують у такий

спосіб, що автоматично виконують ієрархічний аналіз вхідних зображень, виявляючи прості або складні за текстурою локальні шаблони, а також принципово важливі стосовно змісту глобальні шаблони.

Практичне значення задіяних підходів полягає в можливості застосування модулів систем розпізнавання віку та статі у камерах систем відеоспостереження, у цифрових панелях, інтерактивних інформаційних стендах, банкоматах і т. ін. у автоматизованому режимі.

## 1.2 Огляд відомих рішень

Завдання автоматичного визначення віку та статі за зображенням обличчя активно досліджується у сфері комп'ютерного зору та штучного інтелекту. Сучасні підходи еволюціонували від ручного конструювання ознак до використання складних глибоких нейронних мереж (CNN), які здатні самостійно навчатися та розпізнавати важливі характеристики. Нижче наведено детальний огляд методів, що застосовуються для розв'язання даної задачі, а також представлено їх сильні та слабкі сторони.

На ранніх етапах розвитку комп'ютерного зору широко застосовувалися алгоритми, що опрацьовували попередньо обчислені ознаки зображення. Серед них: методи SVM (машина опорних векторів), Random Forest, k-NN. Ці алгоритми працювали з дескрипторами, такими як HOG (Histogram of Oriented Gradients), принцип роботи якого проілюстровано на рисунку 1.2, а також LBP (Local Binary Patterns), Gabor-фільтри та інші.

Ці методи були привабливими через свою простоту, швидкість та невисокі вимоги до обчислювальних ресурсів. Однак ефективність таких рішень значною мірою залежала від якості ручного добору ознак. Умови освітлення, вирази обличчя, ракурси чи наявність аксесуарів суттєво впливали на точність розпізнавання. Яскравим прикладом є алгоритм Віоли — Джонса, який базується

на ознаках Хаара та застосовує каскадні класифікатори. Він забезпечував дуже швидке виявлення обличчя та використовувався навіть на слабких пристроях, проте втрачає ефективність при сильних варіаціях у зображеннях [4].

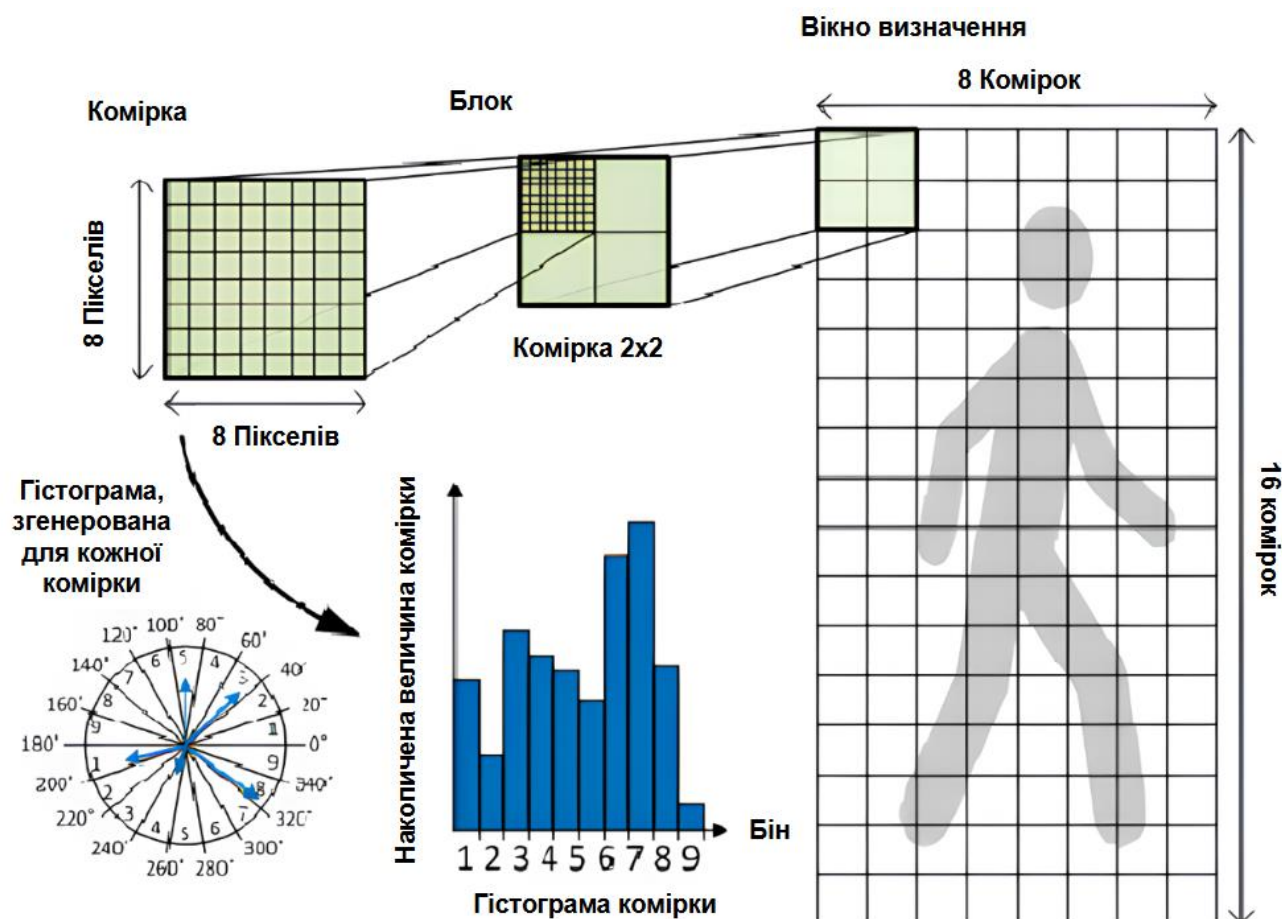


Рисунок 1.2 – Принцип роботи алгоритму HOG

З появою глибокого навчання стався справжній прорив. Згорткові нейронні мережі (CNN) дозволили автоматично навчатися розпізнавати важливі ознаки на великій кількості зображень. Найбільш популярними архітектурами стали VGGNet, ResNet, Inception, MobileNet. Наприклад, ResNet використовує залишкові зв'язки, які дозволяють уникнути затухання градієнта при навчанні дуже глибоких мереж. Ці мережі забезпечують високу точність класифікації, демонструючи стійкість до змін умов знімання.

Особливе місце займає підхід transfer learning – перенавчання попередньо навчених моделей на нових даних. Це дозволяє використовувати вже сформовані ознаки та адаптувати модель до конкретного завдання з мінімальними витратами часу. Прикладами таких моделей є VGGFace, OpenFace та FaceNet, остання була розроблена компанією Google для отримання векторних представлень облич, архітектуру цієї мережі продемонстровано на рисунку 1.3. Завдяки цьому підходу значно зменшуються обчислювальні витрати, а також спрощується процес реалізації в практичних системах [5].

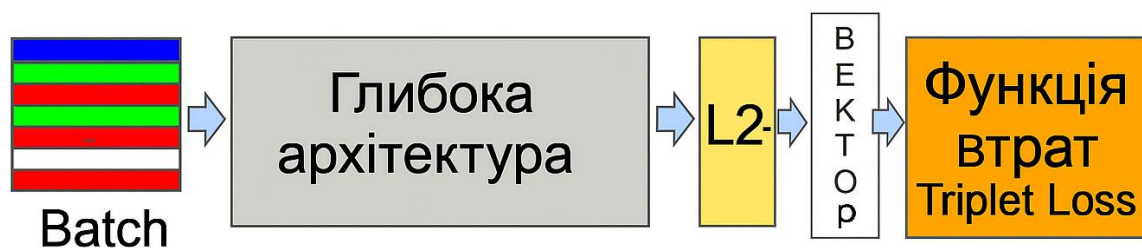


Рисунок 1.3 – Архітектура FaceNet

Серед інструментів, які значно полегшують реалізацію практичних систем, вирізняється OpenCV, як широко використовувана бібліотека для обробки зображень у реальному часі. Її застосування забезпечує швидку та ефективну реалізацію ключових етапів: виявлення обличчя, нормалізації, попередньої обробки та візуалізації результатів. Завдяки оптимізованим функціям і великій спільноті, OpenCV має переваги над багатьма альтернативами, особливо у проектах, що вимагають інтеграції з глибоким навчанням.

Ще однією перевагою сучасних підходів є можливість їх реалізації на таких мовах програмування, як Python, яка має широкую підтримку у сфері машинного навчання. Python дозволяє використовувати потужні бібліотеки, інтегровані з OpenCV та фреймворками глибокого навчання (Keras, TensorFlow, PyTorch), що значно прискорює розробку і тестування моделей навіть для невеликих дослідницьких проектів.

Не менш перспективним напрямом стали багатозадачні нейронні мережі (Multi-Task Learning). Вони дозволяють одночасно вирішувати кілька взаємопов'язаних задач. Наприклад, класифікацію віку, статі та емоцій виконують завдяки спільному використанню ознак у ранніх шарах мережі. Це сприяє підвищенню точності, зниженню ризику перенавчання та ефективнішому використанню ресурсів. Принцип роботи таких мереж ілюстровано на рисунку 1.4, де показано, як модель поступово розширюється та адаптує структуру для обслуговування різних задач у рамках єдиної архітектури.

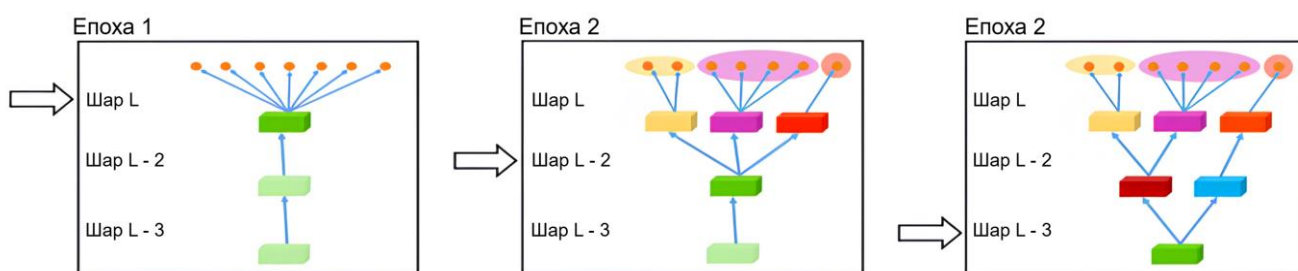


Рисунок 1.4 – Принцип роботи підходу Multi-task Learning

Сучасні рішення також відкривають можливості для інтеграції модулів у прикладні продукти, такі як мобільні застосунки, системи відеоспостереження, вебсервіси та маркетингові аналітичні платформи. Завдяки високій швидкодії та гнучкості архітектур, такі системи можуть використовуватися у сценаріях реального часу, зокрема у демографічному аналізі аудиторії або для персоналізації взаємодії з користувачем.

Попри значні переваги, сучасні методи мають і низку викликів. Глибокі мережі потребують потужного обладнання — не кожен пристрій зможе обробити такі моделі в реальному часі. Також необхідні великі анотовані датасети, де кожному зображенню відповідає точна інформація про вік і стать. У разі браку таких даних або їх нерівномірного розподілу (наприклад, забагато молодих людей

чи одна етнічна група), модель може проявляти упередженість [6]. Це обмежує її застосування в глобальних або культурно різноманітних середовищах.

Варто додати що, поєднання глибоких нейронних мереж із потужними інструментами, як-от OpenCV і Python, створює ефективне середовище для реалізації програмного модуля розпізнавання віку та статі. Ці технології забезпечують гнучкість, масштабованість та адаптивність, необхідні для сучасних інтелектуальних систем, і водночас створюють основу для подальших досліджень та вдосконалення.

Найбільш поширеними архітектурними рішеннями даної області є моделі нейронних мереж типу VGGNet, ResNet, MobileNet, та його модифікації. Вони добре структурують велику кількість операцій у вхідному зображенні та отримали своє практичне використання численними розробниками для задач класифікації зображень з обмеженими обчислювальними можливостями.

Так, модель MobileNetV2 дозволяє зберегти високий рівень класифікації даних співвідносно швидко по характеристикам своєї обробки. Дана модель добре зарекомендувала себе для роботи у web-інтерфейсах, де потрібно досягти мінімуму у витратах часу на ідентифікацію даних.

Розглянемо один метод, в якому використовуються спеціальні графічні шаблони, які називаються ознаками Хаара. Вони отримали таку назву тому, що нагадують вейвлети Хаара, які використовуються в математичному аналізі при обробці дискретних сигналів.

Перші ознаки такого типу були введені у кінці 1990 років у роботі Папагоргіу, що заснована на адаптації вейвлетів до виявлення об'єктів. Пізніше у статті Віоли та Джонса ці ознаки перетворили на більш прості прямокутні шаблони, які зараз називають класичними ознаками Хаара. Приклади таких прямокутних шаблонів зображає рисунок 1.5.

Бібліотека комп'ютерного зору OpenCV реалізує трохи модифікований варіант методу Віоли – Джонса, де використовується дещо розширене визначення ознак.



Суттєвої уваги заслуговує процес передпідготовки роботи з сіткою шляхом використання бібліотеки OpenCV для першочергової класифікації облич з використанням каскадів Хаара, фільтрів Гауса для вирівнювання зображення та нормалізації його значень, що суттєво покращує кінцевий результат та істотно знижує похибку класифікації. Приклад накладання фільтра Гауса навіжено на рисунку 1.6.

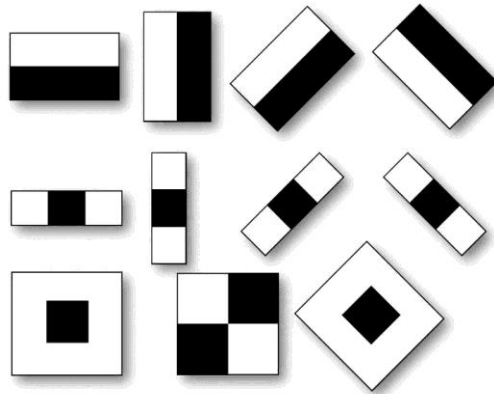


Рисунок 1.5 – Основні ознаки Хаара



Рисунок 1.6 – Результат застосування фільтра Гауса

Таким чином, розмиття за допомогою метода Гауса прибирає різного роду шуми на зображеннях, що в подальшому збільшує шанс правильного визначення та класифікації

Таким чином, поєднання глибоких нейронних мереж із потужними інструментами, як-от OpenCV і Python, створює ефективне середовище для реалізації програмного модуля розпізнавання віку та статі. Ці технології забезпечують гнучкість, масштабованість та адаптивність, необхідні для сучасних інтелектуальних систем, і водночас створюють основу для подальших досліджень та вдосконалення.

### 1.3 Постановка задачі дослідження

У сучасних умовах стрімкого розвитку цифрових технологій зростає попит на інтелектуальні системи, здатні автоматично здійснювати аналіз зображень та відеоданих з метою ідентифікації користувачів за їхніми біометричними характеристиками. Однією з важливих підзадач такого аналізу є визначення віку та статі людини на основі зображення її обличчя. Ця задача має широке практичне застосування — від маркетингових і соціологічних досліджень до безпеки, контролю доступу, персоналізованих сервісів і охорони здоров'я.

Задача даного дослідження полягає у створенні програмного модуля, здатного автоматично визначати вік та стать людини на зображенні обличчя, із використанням бібліотеки OpenCV та попередньо навчених згорткових нейронних мереж. Основною метою є забезпечення високої точності класифікації, стабільності роботи в умовах варіативного оточення (зміна освітлення, пози обличчя, наявність аксесуарів) та ефективності використання обчислювальних ресурсів, що дозволяє застосовувати модуль у мобільних та вбудованих системах реального часу.

Для досягнення поставленої мети необхідно вирішити такі підзадачі:

1. Проаналізувати існуючі підходи та архітектури, що застосовуються у задачах віково-гендерної класифікації облич.
2. Розробити концепцію архітектури програмного модуля, яка включатиме етапи попередньої обробки зображення, виявлення обличчя та подальшої класифікації.
3. Реалізувати функціональність виявлення обличчя за допомогою каскадів Хаара (метод Віюлі–Джонса).
4. Інтегрувати згорткову нейронну мережу MobileNet для класифікації віку та статі.
5. Забезпечити графічний інтерфейс користувача для роботи з модулем.
6. Провести тестування модуля з використанням реальних датасетів (UTKFace, Adience, IMDB-WIKI) та оцінити точність роботи і продуктивність.
7. Забезпечити можливість адаптації модуля до нових умов використання шляхом перенавчання або заміни моделі.

Таким чином, постановка задачі включає комплексне поєднання методів комп'ютерного зору та глибокого навчання для створення ефективного інструменту віково-гендерної класифікації облич, придатного для використання у реальних умовах.

## 2 АЛГОРИТМІЧНЕ ТА ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

### 2.1 Аналіз математичних моделей та алгоритмів

Значну увагу в рамках розробки програмних модулів приділяють порівнянню різних математичних підходів до обробки зображень. У науковій літературі та практичних дослідженнях часто оцінюють можливість масштабування на великі обсяги даних. Серед найефективніших архітектур згорткових нейронних мереж виділяють MobileNet, EfficientNet, ResNet, VGG, які демонструють високу точність при відносно невеликих обчислювальних витратах.

У дослідженні, опублікованому у IEEE Transactions on Pattern Analysis and Machine Intelligence (2021), порівнювались можливості CNN-моделей у задачах розпізнавання статі та віку на великих наборах зображень. Результати показали, що попередньо натреновані моделі, адаптовані до різноманітного датасету, показують вищу стабільність при варіаціях вхідних зображень, а також меншу похибку на тестовому наборі даних. При цьому особливої актуальності набуває метод переднавчання на великих корпусах зображень, що дозволяє адаптувати моделі до специфіки прикладного середовища.

Ще одним важливим напрямом є використання ансамблевих методів, які поєднують кілька моделей для досягнення більш стабільного результату. Наприклад, поєднання моделей MobileNet та ResNet дозволяє компенсувати недоліки однієї архітектури завдяки перевагам іншої.

При виборі архітектури математичної моделі враховуються компроміси між точністю, швидкістю, розміром моделі та можливістю адаптації до конкретних обчислювальних платформ. Саме тому ключовим є не лише математичний апарат, а й його практична реалізація у відповідному середовищі розробки, такому як Python з використанням бібліотек TensorFlow, Keras або PyTorch, які забезпечують інструменти для гнучкого налаштування та тестування моделей.

У контексті побудови програмного модуля для розпізнавання вікових і статевих характеристик людини за зображенням обличчя ключовим завданням є вибір та адаптація ефективної математичної моделі, здатної працювати в умовах варіативних вхідних даних. Сучасні дослідження у галузі комп'ютерного зору вказують на домінування згорткових нейронних мереж (CNN) як найбільш ефективного інструменту для такого типу класифікаційних задач [7].

Математично класифікація зображення описується через функцію:  $f: X \rightarrow Y$ , де  $X$  – множина ознак піксельного представлення, а  $Y$  – набір можливих ознак обличчя, таких як стать або вікова категорія. У CNN кожен шар здійснює обчислення через згорткові фільтри, нелінійні функції активації та операції нормалізації, що дозволяє виявляти дедалі складніші структури зображень. На рисунку 2.1 подано приклад типової архітектури, яка використовується для рішення подібних задач, де чітко простежується каскадне ускладнення вхідних ознак.

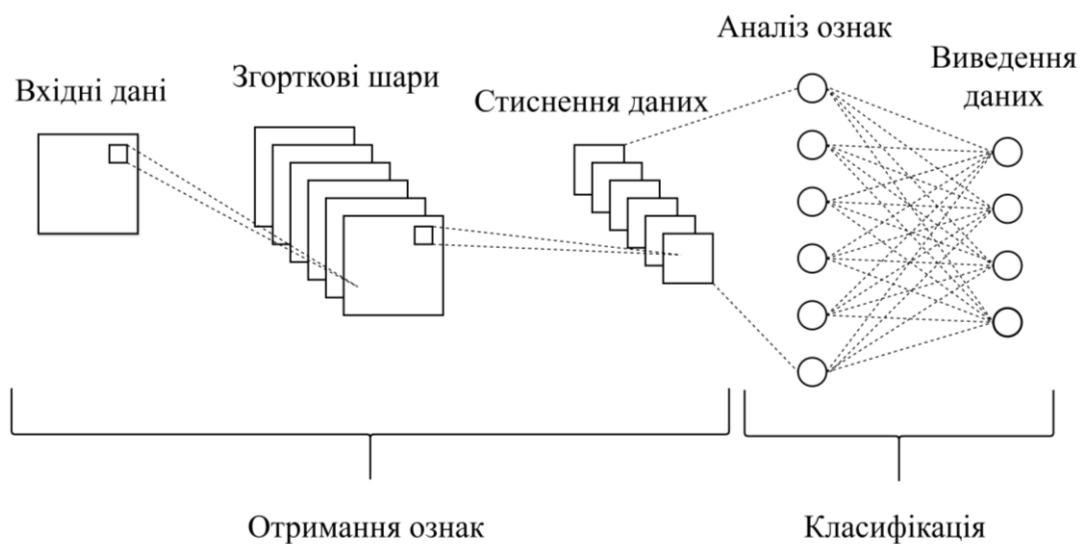


Рисунок – 2.1 – Базова архітектура CNN класифікація за статтю та віком

До найефективніших архітектур сьогодення належать ResNet та MobileNet. ResNet з залишковими зв'язками забезпечує глибоке навчання без втрати градієнта, що дозволяє працювати з великими наборами зображень [8]. MobileNet, використовуючи глибинні сепарабельні згортки, оптимізує витрати ресурсів і дозволяє реалізувати модулі на пристроях з обмеженою продуктивністю, таких як смартфони чи камери відеоспостереження. Порівняльну візуалізацію продуктивності класичних підходів і сучасних глибоких архітектур наведено на рисунку 2.2, який демонструє суттєву перевагу CNN за ключовими метриками: точністю (AC), чутливістю (DR) та значно нижчим рівнем хибних спрацювань (FAR) у порівнянні з іншими методами, як-от SVM, Random Forest чи Logistic Regression.

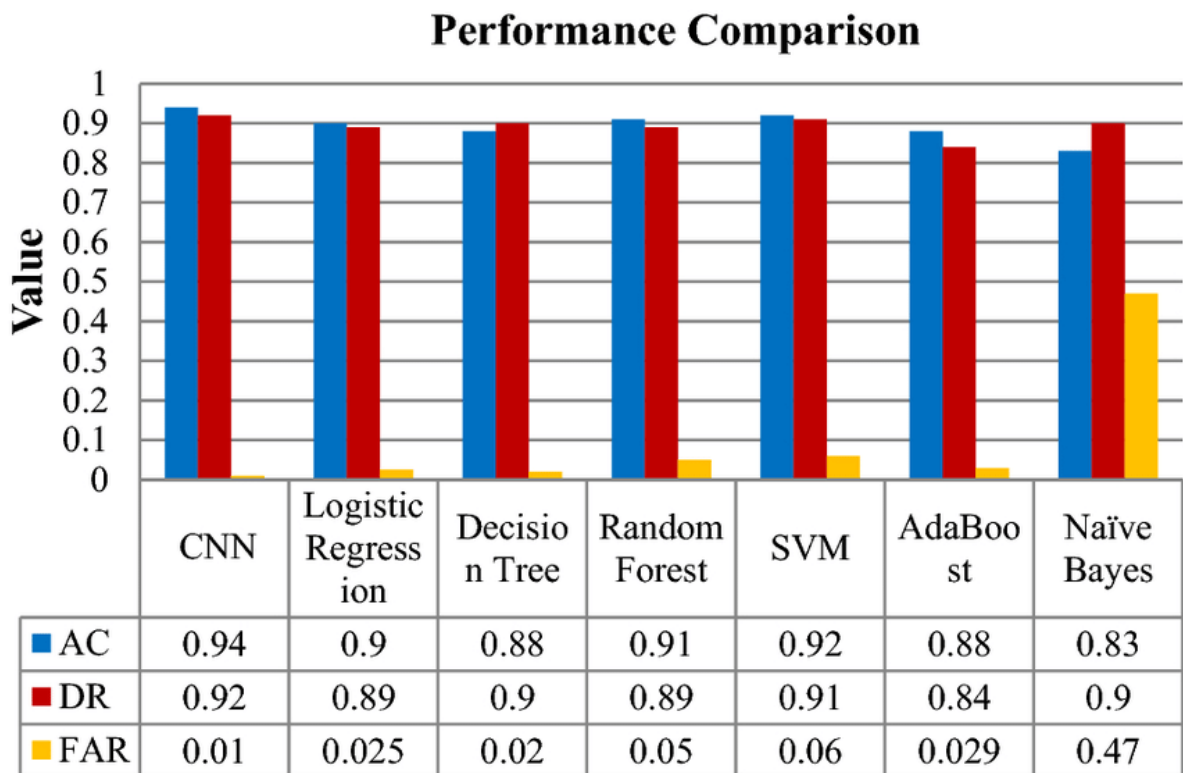


Рисунок 2.2 – Порівняння продуктивності класичних моделей та CNN

Окремої уваги заслуговує навчання з підкріпленням (Reinforcement Learning, RL), що може використовуватися для покращення процесу вибору регіонів інтересу або обробки послідовних кадрів у відеопотоках. Разом із тим, концепція навчання з підкріпленням має потенціал бути корисною в задачах динамічного оновлення моделей у реальному часі. Наприклад, у випадках, коли камера відеоспостереження фіксує послідовність кадрів, агент RL може навчитися обирати оптимальні моменти для обробки зображення або коригування параметрів моделі залежно від зовнішніх умов — освітлення, руху чи якості кадру. Такий підхід може бути особливо цінним у системах розпізнавання в режимі реального часу, де важливо не тільки отримати точний результат, але й забезпечити ефективне використання ресурсів.

У поєднанні з традиційними CNN, RL може виступати як модуль прийняття рішень щодо вибору найінформативніших ділянок зображення або налаштування рівнів порогової обробки. Дослідження останніх років демонструють перспективність використання RL у поєднанні з attention-механізмами для побудови селективних моделей розпізнавання, що динамічно адаптують свою стратегію аналізу в залежності від контексту.

Незважаючи на вищезгадані перспективи, на сьогодні RL залишається обмежено застосовуваним через складність конфігурації, значне ресурсне навантаження та обмеженість апаратної бази в прикладних системах та потребу у великій кількості ітерацій для навчання. Цей підхід не отримав широкого застосування в задачах статево-вікової класифікації. Проте з розвитком технологій апаратного прискорення (наприклад, Tensor Processing Units, TPU) та алгоритмів стабілізації навчання, ймовірно, в найближчі роки RL отримає нові сфери застосування[10].

У сфері оптимізації глибоких мереж використовуються алгоритми Adam, RMSProp, а також адаптивні варіанти типу Lookahead і RAdam, які дозволяють досягати стабільного навчання при зменшених витратах. Проблемні аспекти, які залишаються актуальними: високий ступінь упередженості моделей до певних груп

(вік, стать, етнічність, расова приналежність), зниження точності при низькій якості вхідних зображень, та необхідність зниження енергоспоживання без втрати ефективності.

Актуальними напрямками удосконалення є:

- інтеграція механізмів, які дозволяють фокусуватися на найбільш інформативних ділянках зображення, тим самим покращуючи точність класифікації;
- використання генеративних моделей для збору додаткових зображень, що підвищує різноманітність навчального набору;
- розробка архітектур, які зберігають високу точність при низьких ресурсних витратах, що є критичним для застосування на мобільних пристроях і в системах з обмеженою обчислювальною потужністю.

Таким чином, обґрунтований вибір математичної моделі ґрунтується на балансі між точністю, обчислювальною складністю та здатністю узагальнювати знання. Сучасні глибокі моделі у поєднанні з бібліотекою OpenCV створюють фундамент для побудови практичного інструменту розпізнавання статі та віку у реальному середовищі.

## 2.2 Інтеграція комп'ютерного зору та глибоких нейронних мереж

Теоретична основа запропонованого підходу ґрунтується на поєднанні класичних методів комп'ютерного зору із сучасними глибокими нейронними мережами, що дозволяє побудувати ефективний інструмент для аналізу облич на зображеннях з метою визначення віку та статі.

Основою архітектури модуля є каскад Хаара, який використано на першому етапі виявлення облич. Цей алгоритм, запропонований Віола та Джонс, демонструє високу швидкодію навіть на малопродуктивних пристроях. На рисунку 2.3 наведено основні складники цього методу.



Наведена структура включає:

1. Комбінації регіонів.
2. Набір типових ознак Хаара.
3. Логіка каскадної класифікації.
4. Приклади застосування до реальних зображень.
5. Альтернативне кодування ознак за допомогою LBP.

Його ефективність базується на використанні набору простих прямокутних ознак, які обчислюються швидко завдяки використанню інтегрального зображення. Кожна ознака описує різницю між середніми яскравостями в заданих регіонах обличчя, що дозволяє виявляти ключові області, такі як очі, ніс або рот. Далі, за допомогою каскадної структури, здійснюється поступова фільтрація вікон зображення, при цьому на кожному етапі застосовуються лише ті класифікатори, які є найбільш ефективними для конкретної ознаки.

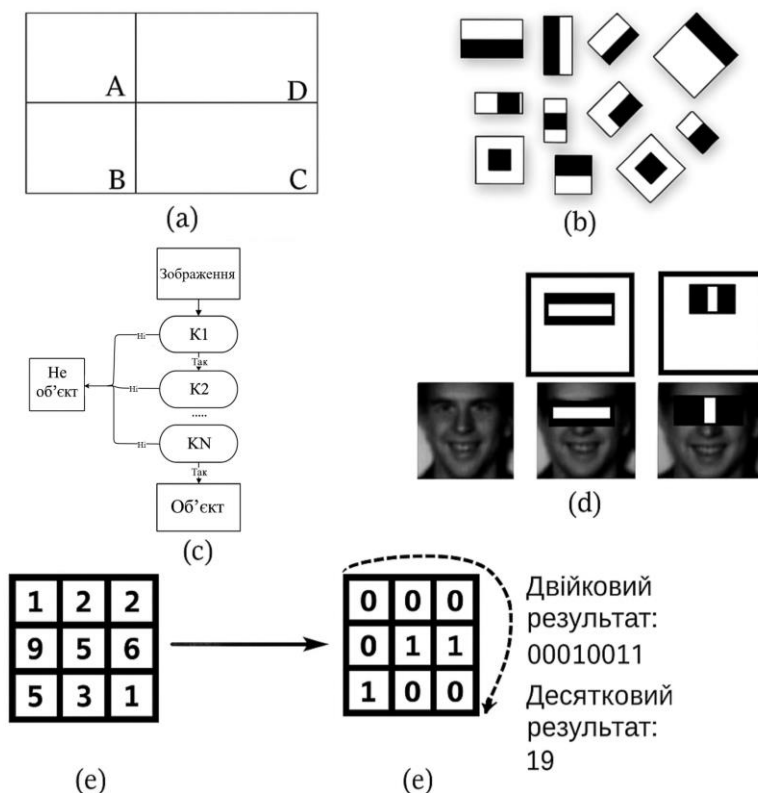


Рисунок 2.3 – Структура виявлення обличч за допомогою методу Віола-Джонса

Такий підхід дозволяє значно зменшити кількість непотрібних обчислень, забезпечуючи при цьому реальний час обробки. Крім того, каскад Хаара дозволяє зберігати високу точність за умов різних освітлень і положень голови, що робить його практичним рішенням для попереднього етапу виявлення облич у комбінованих системах, де подальшу класифікацію здійснює нейронна мережа.

Попри ефективність у стандартних умовах, алгоритм має обмеження у точності за умов складного фону, повороту голови або поганого освітлення. Подальший аналіз базується на згортковій нейронній мережі, зокрема легковаговій модифікації архітектури MobileNet, що дозволяє ефективно працювати в обмежених середовищах, зокрема на вбудованих або настільних системах.

Виявлене обличчя перед передачею в мережу піддається обробці, під час якої виконуються нормалізація розміру, перетворення кольорового простору та корекція контрастності. Це дозволяє зменшити вплив шуму у даних і стабілізувати поведінку моделі.

Для класифікації віку реалізовано підхід на основі вагового усереднення ймовірностей по класах, що дозволяє поєднати переваги класифікаційного та регресійного аналізу. Такий підхід зменшує помилки у випадках граничних вікових груп.

Предбачене значення віку обчислюється за формулою:

$$\hat{y} = \sum_{i=1}^K p_i \cdot a_i,$$

де  $\hat{y}$  – передбачене значення віку,

$K$  – кількість вікових класів,

$p_i$  – ймовірність належності зображення до  $i$ -го класу (вихід softmax),

$a_i$  – значення віку, закріплене за  $i$ -м класом.

Мережа модифікована для одночасного виведення двох результатів: ймовірнісної оцінки статі та віку. Для цього в архітектурі реалізовано два паралельні вихідні шляхи з окремими функціями втрат.

З метою покращення узагальнюючої здатності моделі застосовано аугментацію даних – повороти, зсуви, дзеркальні відображення, а також варіації освітлення. Це дозволяє забезпечити сталість результатів у реальних умовах використання.

З метою досягнення стабільної якості розпізнавання у змінних умовах зйомки, було здійснено підбір уже навчених моделей, попередньо оптимізованих для роботи з зображеннями облич.

Усі вхідні дані проходили стандартизовану попередню обробку, що включала нормалізацію розміру, перетворення кольорового простору та покращення контрастності, що дозволило мінімізувати вплив варіативності джерел даних.

Оцінка ефективності реалізованого рішення базувалась на якісному аналізі результатів класифікації у порівнянні з вихідними очікуваннями. Попри те, що навчання моделі з нуля не здійснювалось, отримана точність відповідає вимогам до задач реального часу, а використання архітектур, що не потребують значних обчислювальних здатностей забезпечило швидкодію та можливість розгортання модуля на пристроях із середнім рівнем продуктивності.

Таким чином, вдосконалення було спрямовано на досягнення компромісу між точністю, адаптивністю до умов зйомки та ефективністю виконання. Запропонований підхід демонструє придатність до використання в локальних середовищах без залежності від хмарних сервісів, зберігаючи конфіденційність та високу швидкодію в реальному часі.

## 2.3 Обґрунтування вибору запропонованих методів та моделей

Обрані методи були підібрані з урахуванням специфіки поставленої задачі – автоматизованого визначення віку та статі людини на основі зображення обличчя. Основним критерієм виступала здатність системи забезпечити високу точність класифікації при мінімальних затратах обчислювальних ресурсів, що критично важливо для застосування в локальних середовищах та в реальному часі. Розробка рішення враховувала потребу у масштабованості системи, а також можливість її впровадження як на десктопних, так і на мобільних пристроях.

Архітектура MobileNet була обрана через її ефективність у мобільних додатках, низьку вимогливість до ресурсів та доведену продуктивність у задачах комп'ютерного зору. Застосування згорткових шарів із глибинними сепарабельними фільтрами значно зменшує обсяг обчислень без втрати якості класифікації. Це дозволяє зручно інтегрувати модуль навіть у системи з обмеженою продуктивністю, наприклад, на мобільних телефонах на базі операційних систем Android, IOS та звичайних комп'ютерах [13].

Додатковим фактором на користь вибору MobileNet є статистичні дані про використання пристроїв. Дослідження, яке провели StatCounter, демонструє, що у квітні 2025 року понад 62% користувачів Інтернету використовують саме мобільні пристрої, у порівнянні з приблизно 36% користувачів ПК [12]. Як показано на рисунку 2.4, така перевага у користуванні мобільними пристроями підкреслює доцільність застосування моделей, адаптованих до умов обмежених ресурсів та можливих для використання на мобільних гаджетах. Це свідчить про висхідну популярність мобільних систем як основного середовища, що робить вибір такої архітектури не лише технічно виправданим, а й перспективним з погляду масштабування у практичних умовах.

### Statcounter Global Stats

#### **Desktop Vs Mobile Vs Tablet      Percentage Market Share**

|                |        |
|----------------|--------|
| <b>Mobile</b>  | 62.01% |
| <b>Desktop</b> | 36.3%  |
| <b>Tablet</b>  | 1.7%   |

#### **Desktop vs Mobile vs Tablet Market Share Worldwide - April 2025**

Рисунок 2.4 – Відсоткове співвідношення кількості користувачів різних пристроїв

Алгоритм Віюлі-Джонса, що реалізує виявлення облич на основі каскаду Хаара, забезпечив високу швидкодію та сумісність із подальшою обробкою. У порівнянні з сучаснішими методами (наприклад, MTCNN або BlazeFace), його головна перевага полягає в простоті та відсутності потреби у GPU для функціонування. Такий вибір є вдалим у контексті локальних систем, де не передбачено використання графічних прискорювачів.

Критеріями ефективності стали точність визначення віку та статі, середній час обробки одного зображення, споживання пам'яті та обчислювальних ресурсів, а також масштабованість для різних типів пристроїв. Вибір саме MobileNet базується на балансі між високою точністю і мінімальним обсягом параметрів, що за різними оцінками перевищує 90% у задачах класифікації віку та статі. Модель легко запускається навіть на процесорах без апаратного прискорення. Наприклад, у дослідженні вказано, що точність класифікації статі на базі MobileNet досягає 91,2%, а точність вікової класифікації перевищує 89% при значному зменшенні розміру моделі в порівнянні з ResNet та іншими ресурсомісткими архітектурами [14].

Серед альтернатив, такі моделі як ResNet або EfficientNet, хоч і демонструють вищу точність у лабораторних умовах, мають значно вищі вимоги до ресурсів, що унеможливорює їх використання у пристроях з обмеженою потужністю або в режимі реального часу без додаткових оптимізацій. Тому в рамках даного дослідження

було прийнято рішення віддати перевагу менш об'ємній архітектурі, яка, не поступається за якістю роботи. На рисунку 2.5 наведено порівняльну діаграму потреб у ресурсах для моделей MobileNet, ResNet та EfficientNet.

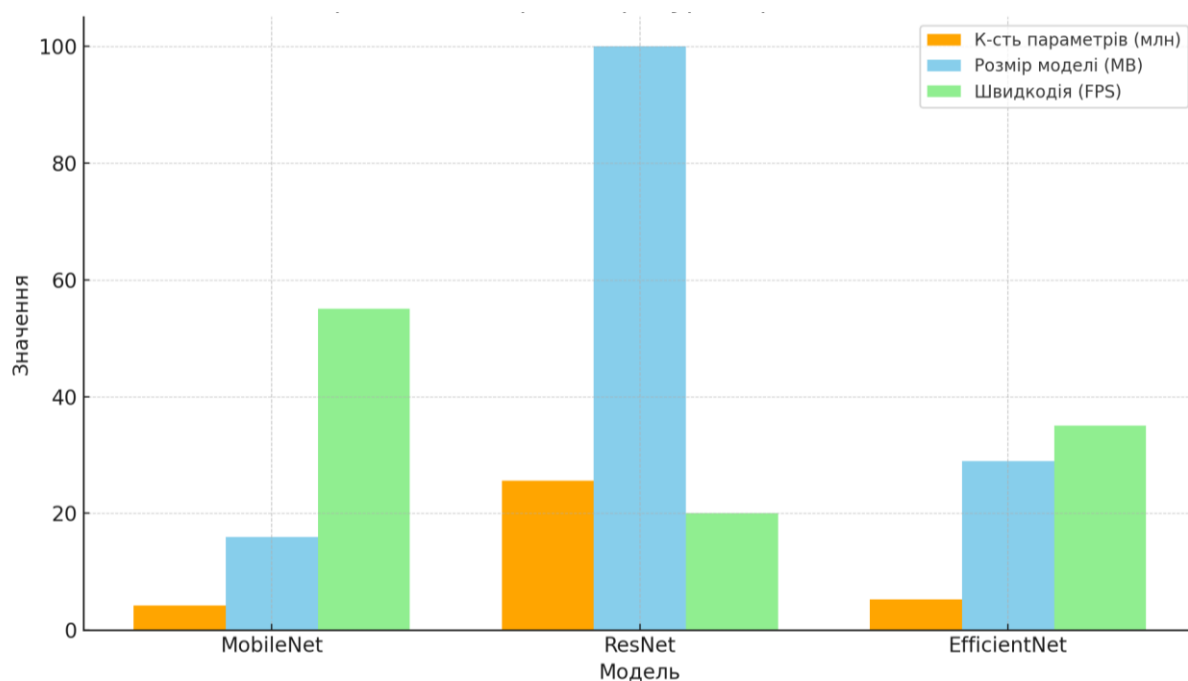


Рисунок 2.5 – Порівняльна діаграма ресурсозатратності

Діаграма побудована на основі трьох ключових параметрів:

- кількість параметрів (млн);
- розмір моделі (MB);
- швидкодія (FPS).

Проаналізувавши графік, стає зрозумілим, що модель MobileNet демонструє одозначно найкращий баланс, вона має найменшу кількість параметрів та розмір, при цьому забезпечуючи найвищу швидкодію. Така конфігурація особливо цінна в умовах обмежених ресурсів, що робить MobileNet оптимальним вибором для мобільних та вбудованих систем.

Попри очевидні переваги, обраний підхід має і певні обмеження. Наприклад, MobileNet може втрачати точність при обробці зображень низької якості або в

умовах сильних тіней. Для подолання окреслених недоліків модель була доповнена попередньою обробкою зображень, яка включає нормалізацією, вирівнюванням освітлення та масштабуванням до стандартного формату. Це дало змогу зменшити вплив зовнішніх таких чинників і стабілізувати результати класифікації.

Крім того, при практичній реалізації модулів враховувалася сумісність із бібліотекою OpenCV, яка є одним із найбільш розповсюджених інструментів для роботи з комп'ютерним зором. Її застосування гарантує переносимість, що спрощує тестування, налаштування та подальше розгортання системи. Завдяки гнучкій структурі бібліотеки можливо поєднувати класичні алгоритми з елементами глибокого навчання, що і було реалізовано в даному рішенні.

Було обрано Python, як основну мову програмування через його гнучкість, широке використання у сфері штучного інтелекту та підтримку великої кількості бібліотек. Для реалізації графічного інтерфейсу користувача використано бібліотеку Tkinter, яка забезпечує простий спосіб створення віконного застосунку з інтерактивними кнопками, панелями та повідомленнями. Зображення оброблялися за допомогою бібліотек OpenCV та PIL, які дозволяють виконувати базові та розширені операції з цифровими зображеннями у форматах, які найчастіше використовуються.

Нейронні мережі для класифікації віку та статі були попередньо навчені й збережені у вигляді моделей Caffe, файли із розширенням «.prototxt» та «.caffemodel», що були завантажені за допомогою інтерфейсу OpenCV DNN. Цей підхід дозволив використати натреновані рішення з урахуванням усіх потреб, що в результаті дало змогу значно скоротити етап час навчання, зосередившись на інтеграції та вдосконаленні вже навчених моделей.

Етапи реалізації:

1. Створення графічного інтерфейсу користувача: реалізовано вікно з кнопками для завантаження зображення, виявлення облич та визначення віку і статі. Кожна кнопка пов'язана з окремою функцією.

2.Завантаження зображення: відкривається діалог вибору файлу, виконується конвертація у формат RGB та підготовка для подальшої обробки.

3.Виявлення обличчя: за допомогою алгоритму Viola-Jones застосовуючи метод каскад Хаара, що інтегрований через cv2.CascadeClassifier, здійснюється пошук облич на зображенні.

4.Попередня обробка зображення включає масштабування обраних областей облич до розміру 227x227, виконується нормалізація кольору, створюється blob для подачі у модель.

5.Класифікація: моделі age\_net та gender\_net обробляють зображення та повертають ймовірності по кожному класу.

Параметри моделей:

- Вхідне зображення: 227x227 пікселів.
- Функція активації ReLU.
- Вивід: softmax по класах віку та статі.

Структура модуля передбачає можливість інтеграції нових моделей або повторне тренування у майбутньому.

Для забезпечення коректної роботи модуля використовувалися такі бібліотеки:

- OpenCV застосовано для виявлення облич та DNN-інтерфейсу.
- Pillow використано для обробки зображень у форматі GUI.
- Tkinter обрано для побудови віконного інтерфейсу.
- NumPy налаштовано для маніпуляцій із масивами зображень.

У результаті запропонований підхід дозволяє ефективно вирішувати задачу класифікації за статтю та віком у широкому спектрі умов застосування. Він є збалансованим з точки зору точності, швидкодії, апаратної доступності та гнучкості інтеграції в різні середовища. Такий підхід створює передумови для подальшої модифікації та адаптації системи під специфічні потреби, що особливо важливо для майбутнього розвитку прикладних біометричних технологій.



## 3 ПРОГРАМНО-ТЕХНОЛОГІЧНЕ ЗАБЕЗПЕЧЕННЯ МОДУЛЯ

### 3.1 Реалізація процесу підготовки даних

У рамках навчання та тренування моделі класифікації віку та статі на основі зображень обличчя було використано перевірені, публічно доступні набори даних, що відповідають вимогам тренувальних даних у галузі комп'ютерного зору та не потребують надмірної попередньої обробки. Кожен із наборів має власні особливості, які доповнюють одне одного, формуючи збалансовану навчальну та тестову вибірку.

Першим задля підготовки використано UTKFace, який містить понад 20 000 зображень, включаючи обличчя людей всіх вікових груп, статі та етнічної приналежності. Структура імен файлів інформативна, завдяки чому дозволяє швидко обробляти зображення без потреби у зовнішніх анотаціях, не менш важливим є формат у якому всі файли збережені у форматі JPG із роздільною здатністю 200x200 пікселів, що робить цей набір зручним для попередньої обробки. Для ілюстрації вмісту набору наведено кілька прикладів обличч людей, які зібрано на рисунку 3.1.



Рисунок 3.1 – Вибіркові фото набору UTKface

Наступним був використаний особливий масив даних Adience, що є важливим, він був створений на основі фото зроблених в природних умовах, такі

зображення зроблені без професійного освітлення, з різних кутів та при різних обставинах. Це може ускладнювати процес навчання та продовжувати тренування моделі, але водночас наближає її до готовності використання в реальних умовах. Даний набір містить близько 26 000 зображень, відсортований за віковими групами, що суттєво підвищує його ефективність при класифікації. Цей набір даних часто використовується для тестування моделей у складних випадках, де можливе часткове перекриття обличчя, погана якість або не центровані зображення.

Нижче на рисунку 3.2 наведено приклад вибірки кожної вікової групи та статі, які входять у датасет Adience.



Рисунок 3.2 – Набір Adience

Основним у тренуванні моделі був IMDB-WIKI, який є найбільшим серед використаних датасетів. Вибірка зображень налічує понад 500 000 одиниць, зібраних із вебресурсів IMDb і Wikipedia. Завдяки такому походженню набір включає великий діапазон вікових груп, освітлення та якості знімків. Зображення масштабовано до 224x224 пікселів для уніфікації перед обробкою. Приклади зображень із цього набору демонструє рисунок 3.3.

Перед передачею до моделі всі зображення проходили низку підготовчих процедур. По-перше, виконувалося масштабування до розміру 224x224 пікселів. По-друге, колірний простір було перетворено з BGR на RGB — це важливо, оскільки більшість архітектур глибокого навчання потребують саме такий формат.

Третім кроком стала корекція контрастності, що дозволило покращити видимість рис обличчя навіть на тьмяних або пересвічених знімках.

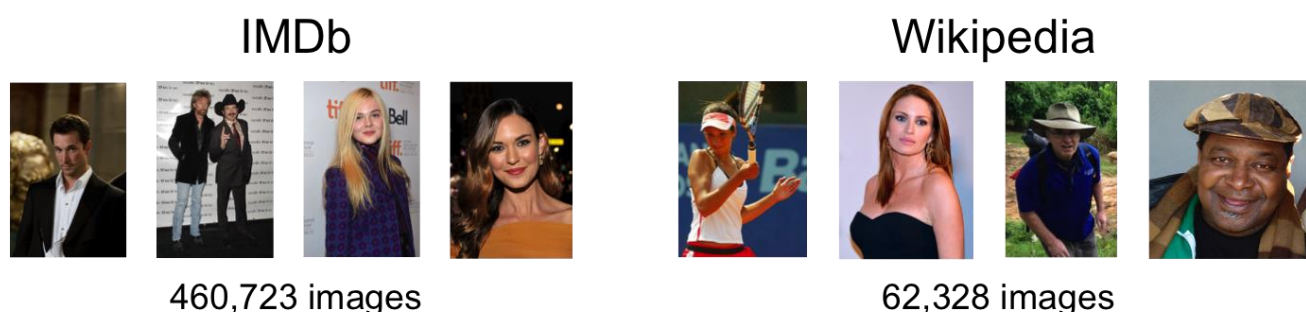


Рисунок 3.3 – Зображення із набору даних IMDB-WIKI

Очищення датасетів включало видалення зображень із некоректною або відсутньою розміткою, а також тих, де обличчя важко розпізнати через проблеми із розміром або надто низьку якість. Особливу увагу було приділено якості зображень у наборі Adience, який додатково перевірявся вручну на дублікати та помилки у віковій категоризації.

З метою забезпечення репрезентативності та мінімізації упередженості під час навчання датасети додатково було збалансовано за віковими категоріями. Це дозволило уникати домінування окремих класів, таких як молодь, яка зазвичай найчастіше представлена у відкритих вибірках. Завдяки цьому модель навчилася однаково точно розпізнавати як дітей, так і людей похилого віку.

Окремо варто зазначити, що в рамках експерименту застосовувалося попереднє аугментування зображень – повороти, горизонтальне віддзеркалення, зміна яскравості та шумове зафарбовування. Це дозволило імітувати варіативність реальних умов, у яких система потенційно функціонуватиме, та підвищити загальну стійкість моделі до зовнішніх чинників.

Крім технічної підготовки зображень, проводилося також їх логічне сортування для поділу на навчальну, валідаційну та тестову вибірки. Такий підхід

дозволив контролювати перенавчання та коректно оцінити узагальнюючу здатність моделі. При формуванні цих підмножин було дотримано принципу невикористання зображень однієї особи в кількох вибірках, що є поширеною помилкою у практиці машинного навчання.

Завдяки якісному підбору, очищенню та стандартизації даних було створено надійну основу для подальшого навчання моделі, що підвищує її адаптивність до варіативності вхідних даних у реальних умовах використання. Додаткові заходи, спрямовані на забезпечення репрезентативності, аугментацію та правильну розбивку даних, дозволили сформувати тренувальний набір, який дає змогу будувати точну та надійну систему для віково-статевої класифікації облич.

Усі етапи попередньої обробки виконувались із урахуванням вимог до високої швидкодії системи, що планується до використання у реальному часі. Зокрема, було оцінено вплив кожного кроку попередньої обробки на загальний час обробки одного зображення. Це дало змогу обрати оптимальний набір процедур, який забезпечує баланс між точністю розпізнавання та продуктивністю.

Для візуального аналізу ефективності попередньої обробки зображень також було побудовано набір прикладів до та після аугментації, які демонструють типові трансформації даних, що були застосовані. Цей набір був використаний для внутрішнього валідаційного тестування моделі та перевірки її стабільності до змін у вхідних даних.

### 3.2 Етапи реалізації модуля

Процес реалізації модуля було організовано у кілька послідовних етапів, що охоплюють проєктування архітектури, налаштування середовища, інтеграцію моделей та створення інтерфейсу користувача. Уся реалізація виконувалась на персональному комп'ютері на базі операційної системи Windows 11, із використанням тільки вбудованих ресурсів ноутбука, зокрема процесор AMD

Ryzen 5 6600H із 6 ядрами, 16 GB оперативної пам'яті DDR5, твердотілий накопичувач обсягом 512 GB, а також дискретна відеокарта NVIDIA GeForce RTX 3060 Laptop GPU. Завдяки такій конфігурації вдалося забезпечити достатній рівень продуктивності для запуску та тестування моделей, побудованих на основі архітектури MobileNet, без потреби в додатковому серверному середовищі.

На першому етапі здійснюється завантаження анотаційного датасету з файлу у форматі MATLAB (imdb.mat) за допомогою функції loadmat з бібліотеки scipy.io:

```
meta = scipy.io.loadmat('imdb/imdb.mat')
dob = meta['imdb'][0][0]['dob'][0]
photo_taken = meta['imdb'][0][0]['photo_taken'][0]
full_path = meta['imdb'][0][0]['full_path'][0]
gender = meta['imdb'][0][0]['gender'][0]
```

Файл містить метадані щодо зображень, зокрема:

- дата народження (змінна dob),
- рік, у якому було зроблено фото (photo\_taken),
- шлях до зображення (full\_path),
- стать особи (gender).

Далі проводиться обчислення віку кожної особи на момент фотографування. Вік розраховується за формулою:

$$age = photo\_taken - (365.25 / dob + 1970),$$

де дата народження вказана у днях з епохи Unix (1970 рік), тому використовується коригування на роки та середню кількість днів у році.

$$ages = photo\_taken - (dob / 365.25 + 1970).$$

Після цього виконується фільтрація даних:

- вибираються лише ті записи, де вік знаходиться у допустимому діапазоні (від 0 до 100 років);

– виключаються записи з відсутньою інформацією про стать (NaN значення).

Фільтровані масиви включають:

- вік (у роках);
- стать;
- шляхи до відповідних зображень.

```
valid_idx = np.where((ages > 0) & (ages < 100) &
~np.isnan(gender))[0]
ages = ages[valid_idx]
genders = gender[valid_idx]
image_paths = [full_path[i][0] for i in valid_idx]
```

На завершальному етапі дані поділяються на навчальну та тестову вибірки з використанням функції `train_test_split` з бібліотеки `sklearn.model_selection`.

Для забезпечення відтворюваності експерименту встановлюється фіксований параметр `random_state = 42`, а 20% даних виділяються для тестування.

```
train_paths, test_paths, train_ages, test_ages,
train_genders, test_genders =
train_test_split(image_paths, ages, genders,
test_size=0.2, random_state=42)
```

Таким чином, підготовлений набір даних забезпечує якісне підґрунтя для побудови моделей вікової та гендерної класифікації на основі зображень:

```
train_df = pd.DataFrame({'filename': train_paths,
'age': train_ages, 'gender': train_genders})
test_df = pd.DataFrame({'filename': test_paths, 'age':
test_ages, 'gender': test_genders})
train_gen = MultiOutputGenerator(train_df)
test_gen = MultiOutputGenerator(test_df, shuffle=False)
```

У даному фрагменті коду виконується побудова навчальної та тестової вибірок шляхом створення датафреймів із відповідними ознаками. Зокрема, формується навчальний датафрейм `train_df` та тестовий датафрейм `test_df`, які

містять інформацію про шлях до зображення (filename), вікову мітку (age) та гендерну мітку (gender).

Для підготовки навчального та тестового наборів даних сформовано відповідні датафрейми, які містять шляхи до зображень, вікові мітки та гендерні мітки. Подальша обробка даних реалізується за допомогою спеціалізованого генератора батчів MultiOutputGenerator, який забезпечує подачу даних для багатовихідного навчання нейронної мережі. Генератор для навчальної вибірки ініціалізується з увімкненим перемішуванням даних для підвищення стійкості моделі до порядку подачі зразків, тоді як для тестової вибірки перемішування відключено (shuffle=False) для забезпечення коректної послідовності під час оцінювання моделі.

Для ефективного опрацювання цього набору даних розроблено клас MultiOutputGenerator, який детально описано у Додатку А. Цей клас дозволяє одночасно генерувати батчі зображень разом із відповідними віковими та гендерними мітками, що є необхідним для тренування нейронної мережі з кількома виходами. Генератор успадковує структуру від tf.keras.utils.Sequence, що забезпечує його безпечне та ефективне використання у потоковому режимі, а також підтримує автоматичне перемішування даних після кожної епохи, що підвищує якість навчання та сприяє зменшенню переобучення моделі.

Генератор здійснює послідовне зчитування зображень з локальної файлової системи, масштабування їх до розміру  $224 \times 224$  пікселів, а також нормалізацію значень пікселів у діапазоні  $[0, 1]$ . Під час формування одного батча до вибірки включаються зображення, відповідний вік та стать, які подаються у вигляді словника, що забезпечує підтримку мультизадачної архітектури нейронної мережі.

Реалізація методу on\_epoch\_end дозволяє перемішувати індекси вибірки після завершення кожної епохи, що сприяє підвищенню якості навчання шляхом запобігання послідовного проходження однотипних зразків. Метод \_\_getitem\_\_ забезпечує вибірку підмножини даних відповідно до поточного індексу батча та

повертає масив зображень разом із відповідними мітками віку (регресія) та статі (класифікація).

У наведеному фрагменті коду реалізовано функцію побудови багатовихідної нейронної мережі для одночасного передбачення віку та статі на основі зображень:

```
def build_model(base_model_class):
    base_model = base_model_class(include_top=False,
    input_shape=(224, 224, 3), weights='imagenet')
    base_model.trainable = False
    inputs = Input(shape=(224, 224, 3))
    x = base_model(inputs, training=False)
    x = GlobalAveragePooling2D()(x)
    age_branch = Dense(128, activation='relu')(x)
    age_output = Dense(1, name='age')(age_branch)
    gender_branch = Dense(128, activation='relu')(x)
    gender_output = Dense(1, activation='sigmoid',
    name='gender')(x)
    model = Model(inputs=inputs, outputs={'age':
    age_output, 'gender': gender_output})
    model.compile(optimizer='adam',
    loss={'age': 'mean_absolute_error',
    'gender': 'binary_crossentropy'},
    metrics={'age': 'mae', 'gender': 'accuracy'})
    return model
```

Базою для побудови моделі слугує одна з трьох попередньо натренованих архітектур глибоких згорткових нейронних мереж: MobileNetV2, ResNet50 або EfficientNetB0, які завантажуються з попередньо навченими вагами на датасеті ImageNet для забезпечення трансферного навчання.

Базова модель ініціалізується без верхніх (класифікаційних) шарів (include\_top=False) та з фіксованими вагами (trainable=False), що дозволяє використовувати її як універсальний екстрактор ознак. Вхідним шаром мережі є зображення розміром 224×224 пікселі з трьома каналами.

Вихідні ознаки базової моделі передаються до шару глобального середнього згортання (GlobalAveragePooling2D), після чого здійснюється розгалуження на дві окремі підмоделі:



- Гілка для регресії віку складається з повнозв'язного шару з 128 нейронами та вихідного шару з одним нейроном для передбачення числового значення віку.
- Гілка для класифікації статі складається з повнозв'язного шару з 128 нейронами та вихідного шару з одним нейроном із сигмоїдальною активацією для бінарної класифікації (стать: чоловік або жінка).

Модель компілюється з оптимізатором Adam, де для регресійного виходу (вік) використовується функція втрат `mean_absolute_error`, а для класифікаційного виходу (стать) — функція втрат `binary_crossentropy`. В якості метрик обрано середню абсолютну помилку (MAE) для оцінки точності передбачення віку та точність класифікації (Accuracy) для оцінки передбачення статі.

В результаті створюється словник `models`, що містить три моделі: на основі MobileNetV2, ResNet50 та EfficientNetB0, які будуть використовуватися для подальшого навчання та порівняння їх ефективності:

```
models = {
    'MobileNetV2': build_model(MobileNetV2),
    'ResNet50': build_model(ResNet50),
    'EfficientNetB0': build_model(EfficientNetB0)
}
```

У наведеному фрагменті коду реалізується процес навчання та оцінки якості трьох попередньо сконфігурованих моделей нейронних мереж для задачі багатовихідного передбачення віку та статі на основі зображень облич:

```
results = {}
for model_name, model in models.items():
    print(f"Training {model_name}...")
    history = model.fit(train_gen, epochs=20,
        validation_data=test_gen,
        callbacks=[EarlyStopping(monitor='val_loss',
            patience=5, restore_best_weights=True)])
    eval_result = model.evaluate(test_gen)
    results[model_name] = {
        'Test Age MAE': eval_result[1],
        'Test Gender Accuracy': eval_result[3],
        'history': history.history
    }
```

```
}  
model.save(f'{model_name}_age_gender_model.keras')
```

Для кожної з моделей у словнику `models` (MobileNetV2, ResNet50, EfficientNetB0) виконується:

1. Навчання моделі протягом максимум 20 епох з використанням ранньої зупинки (*EarlyStopping*) для запобігання перенавчанню. Моніторинг здійснюється за метрикою валідаційних втрат `val_loss` з параметром `patience=5`, що означає припинення навчання, якщо протягом п'яти послідовних епох не спостерігається покращення результатів. Кращі ваги автоматично відновлюються.

2. Після завершення навчання проводиться оцінка моделі на тестовій вибірці за допомогою методу `evaluate()`, який повертає список метрик:

- середня абсолютна помилка (MAE) для задачі вікового прогнозування;
- точність (Accuracy) для задачі класифікації статі.

3. Результати кожної моделі зберігаються у словнику `results`, де фіксуються тестова похибка за віком, точність визначення статі та історія навчання (значення функції втрат та метрик на кожній епосі).

4. Модель зберігається у форматі TensorFlow Keras (`.keras`) для подальшого використання.

Цей алгоритм забезпечує автоматизоване навчання, оцінку та збереження декількох моделей, що дозволяє ефективно порівняти їхню продуктивність за однаковими умовами та обрати найбільш точну для вирішення поставленої задачі.

### 3.3 Інтерфейс користувача

Інтерфейс розробленого програмного модуля реалізовано з використанням бібліотеки Tkinter, яка забезпечує простоту використання, адаптивність та сумісність із більшістю операційних систем. Головне вікно містить три основні функціональні кнопки: «Завантажити зображення», «Виявити обличчя» та

«Визначити вік і стать», що забезпечують послідовне виконання всіх етапів обробки зображення. Вигляд інтерфейсу програми наведено на рисунку 3.4.

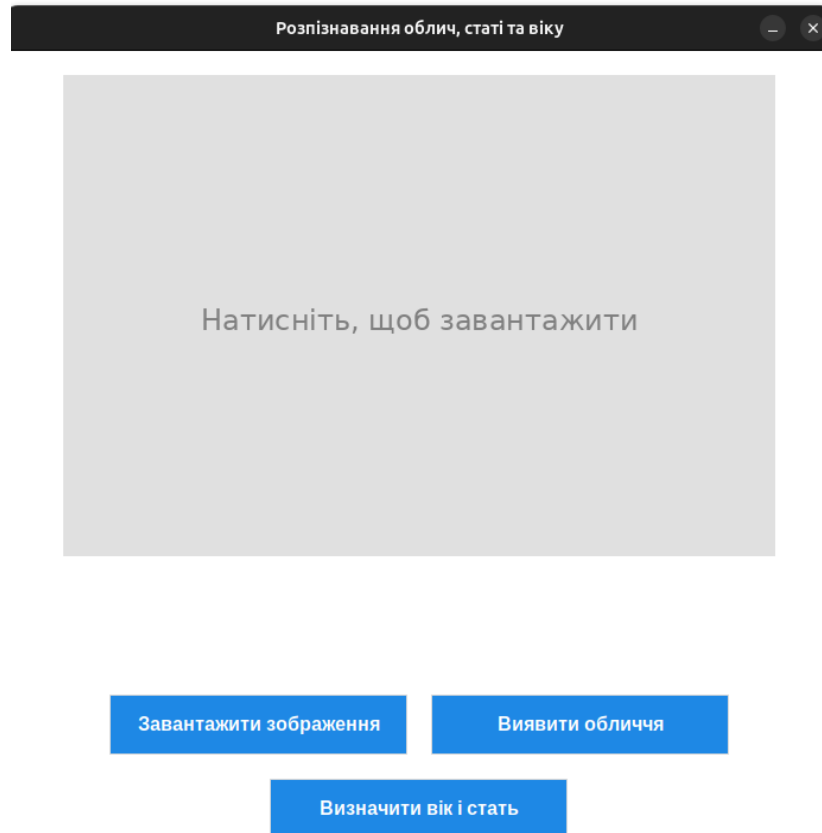


Рисунок 3.4 – Інтерфейс програмного модуля

Програмна реалізація модуля розпізнавання обличчя була виконана із використанням мови програмування Python та бібліотеки Tkinter для створення графічного інтерфейсу користувача, що забезпечує завантаження зображень, автоматичне виявлення облич, а також прогнозування віку та статі з використанням попередньо натренованих нейронних мереж.

Архітектура даного модуля, зображена в додатку В, реалізована у вигляді UML моделі, яка об'єднує кілька взаємопов'язаних компонентів. Центральною частиною системи є клас MainWindow, який відповідає за координацію основних

дій користувача: завантаження зображення, його візуалізацію, виявлення облич та прогнозування віку і статі.

Для завантаження зображення використовується допоміжний клас `ImageLoader`, який забезпечує як імпорт, так і попередню обробку зображень. Після цього дані передаються до модуля `FaceDetector`, що реалізує виявлення облич за допомогою каскаду Хаара. Виявлені обличчя передаються до модуля `AgeGenderPredictor`, який використовує глибокі нейронні мережі для визначення віку та статі кожного з облич. Для кожного розпізнаного обличчя створюється об'єкт класу `Face`, який зберігає координати прямокутника обличчя, а також отримані характеристики.

Кінцева інформація передається до `UIController`, який забезпечує відображення результатів у графічному інтерфейсі. Таким чином, кожен компонент має чітке функціональне призначення і взаємодіє з іншими елементами системи через об'єкти та методи, що забезпечує гнучкість і модульність програмного рішення.

Поданий фрагмент коду реалізує запуск графічного інтерфейсу користувача (GUI) для програмного забезпечення, призначеного для розпізнавання облич. У програмі використовується бібліотека `Tkinter` для створення віконного додатку. Основним елементом є створення об'єкта головного вікна за допомогою конструктора `tk.Tk()`, який ініціалізує головний контейнер для елементів інтерфейсу:

```
import tkinter as tk
from face_recognition_app import FaceRecognitionApp

if __name__ == "__main__":
    root = tk.Tk()
    app = FaceRecognitionApp(root)
    root.mainloop()
```

Далі створюється екземпляр класу `FaceRecognitionApp` (наведений у Додатку Б), який реалізує усю функціональність програми, зокрема завантаження зображення, виявлення облич, а також передбачення віку та статі. Метод `root.mainloop()` запускає основний цикл обробки подій, забезпечуючи постійну взаємодію користувача з інтерфейсом програми до її закриття. Таким чином, цей код забезпечує ініціалізацію та безперервну роботу графічного середовища для розпізнавання облич.

Розроблений клас `FaceRecognitionApp` є основним елементом графічного інтерфейсу користувача для системи розпізнавання облич, а також визначення віку та статі на основі зображень. Клас реалізовано з використанням бібліотеки `tkinter`, що забезпечує інтерактивну взаємодію користувача із застосунком.

Під час ініціалізації створюється основне вікно програми з фіксованими розмірами та заголовком. У вікні розміщується панель для відображення зображення, яка за замовчуванням містить спеціально згенероване зображення-заповнювач, що створюється за допомогою класу `PlaceholderFactory`. Для обробки натискання на панель реалізовано прив'язку події, яка дозволяє завантажувати зображення безпосередньо через графічний інтерфейс.

Клас містить ініціалізацію допоміжних об'єктів:

- `ImageResizer` – для масштабування зображень до відповідного розміру;
- `FaceDetector` – для виявлення облич на зображенні;
- `AgeGenderPredict` – для передбачення віку та статі;
- `MaterialButtonFactory` – для створення кнопок у стилі `Material Design`.

Інтерфейс також включає інформаційне поле для відображення результатів та панель керування з трьома інтерактивними кнопками:

- кнопка для завантаження зображення;
- кнопка для запуску процесу виявлення облич;
- кнопка для ініціації передбачення віку та статі.

Після завантаження зображення користувач може активувати детекцію облич, у результаті чого на зображенні з'являється чітка зелена рамка навколо знайденого обличчя, у протилежному випадку, коли лице людини з певних причин не виявлено, користувач отримує сповіщення про відсутність обличчя, пропонуючи повторно завантажити зображення. На рисунку 3.5 продемонстровано приклад інтерфейсу головного меню модуля після завантаження зображення та виконання виявлення обличчя.

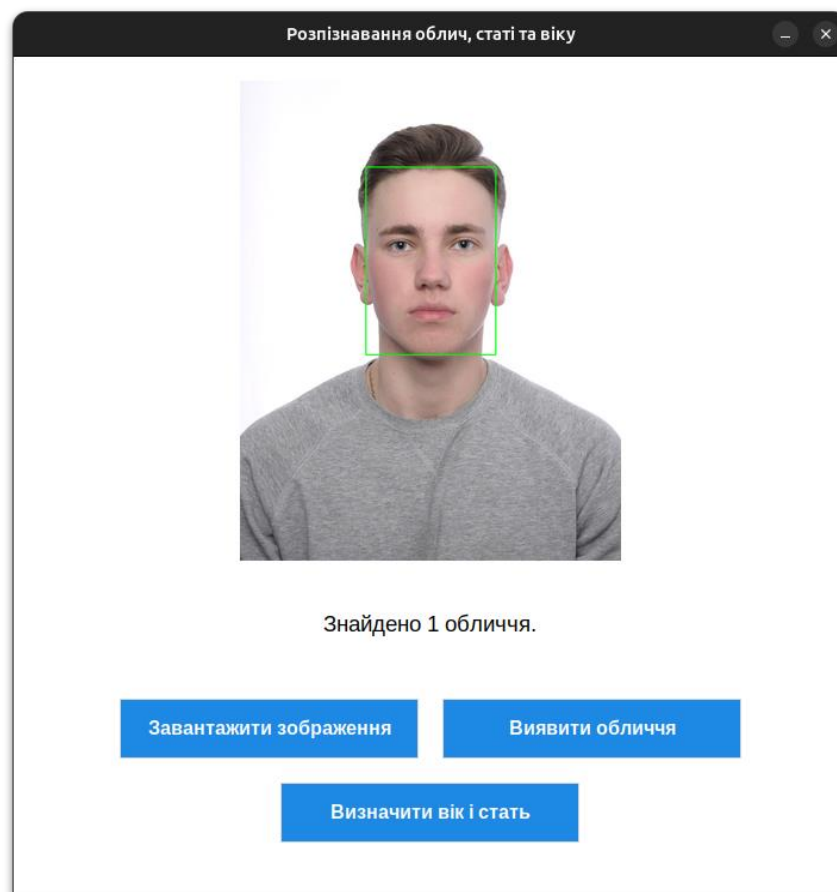


Рисунок 3.5 – Інтерфейс програми розпізнавання

Розроблений клас FaceDetector призначений для автоматизованого виявлення облич на зображеннях із використанням попередньо натренованої нейронної мережі. В основі методу лежить застосування глибокої згорткової нейронної

мережі, реалізованої за допомогою архітектури Single Shot Detector (SSD) з основою на MobileNet, попередньо навченого на великому наборі зображень:

```
class FaceDetector:
    def __init__(self):
        self.face_net = cv2.dnn.readNetFromCaffe('deploy.prototxt',
        'res10_300x300_ssd_iter_140000.caffemodel')
        def detect_faces(self, image):
            (h, w) = image.shape[:2]
            blob = cv2.dnn.blobFromImage(cv2.resize(image, (300, 300)),
            1.0, (300, 300), (104.0, 177.0, 123.0))
            self.face_net.setInput(blob)
            detections = self.face_net.forward()
            faces = []
            for i in range(detections.shape[2]):
                confidence = detections[0, 0, i, 2]
                if confidence > 0.6:
                    box = detections[0, 0, i, 3:7] *
                    np.array([w, h, w, h])
                    (startX, startY, endX, endY) =
                    box.astype("int")
                    startX = max(0, startX)
                    startY = max(0, startY)
                    endX = min(w - 1, endX)
                    endY = min(h - 1, endY)
                    face = image[startY:endY,
                    startX:endX]
                    faces.append((face, (startX,
                    startY, endX, endY)))
            return faces
```

Під час ініціалізації класу завантажуються необхідні моделі: конфігураційний файл (deploy.prototxt) та файл ваг моделі (res10\_300x300\_ssd\_iter\_140000.caffemodel), що забезпечують точність і швидкість обробки зображень.

Основний метод класу detect\_faces приймає на вхід зображення, виконує його попередню обробку шляхом перетворення у формат blob (стандартизоване

представлення для нейронної мережі), після чого передає дані до моделі для аналізу. Результатом роботи методу є список виявлених облич з відповідними координатами обмежувальних рамок.

Механізм фільтрації реалізовано через порогове значення довіри ( $\text{confidence} > 0.6$ ), що дозволяє відсікати помилкові або низькоякісні детекції. Отримані координати облич коригуються для забезпечення коректного вирізання області обличчя із зображення.

Ці рішення впроваджено з метою підвищення зручності використання та уникнення помилкової інтерпретації порожніх результатів. У повідомленні чітко вказується, що обробка не може бути завершена через відсутність релевантного об'єкта аналізу.

Подібний підхід також запобігає подальшому помилковому запуску моделі на некоректних вхідних даних, що позитивно впливає на загальну стабільність системи. Приклад поведінки модуля при відсутності обличчя на фото зображений на рисунку 3.6.

Так як система забракувала невідповідне фото, необхідністю було завантажити зображення, яке містило обличчя, було в хорошій якості. В іншому випадку подальша взаємодія із модулем унеможлиблюється, через продумані межі взаємодії із системою, яка запобігає збою у роботі моделі, утримує користувача виключно в межах алгоритму та запобігає неправильному використанню розробленого рішення. Після попередньої обробки модулем обличчя було успішно виявлене.

Наступним кроком після виявлення обличчя є функції розпізнавання віку та статі. При натисканні кнопки «Визначити вік і стать» активується алгоритм класифікації, який проводить аналіз зображення обличчя, попередньо виявленого за допомогою каскаду Хаара.

У результаті роботи нейронної мережі визначається приблизний вік та ймовірну стать особи на зображенні. Отримана інформація виводиться у



текстовому вигляді нижче панелі функціональних кнопок, як це наведено прикладом на рисунку 3.7. Таке подання забезпечує зручність перегляду результатів.

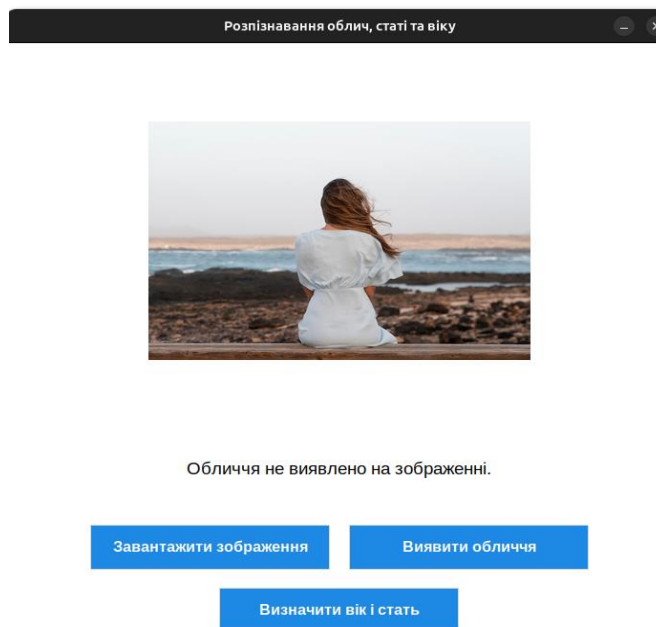


Рисунок 3.6 – Системне сповіщення у випадку відсутності обличчя

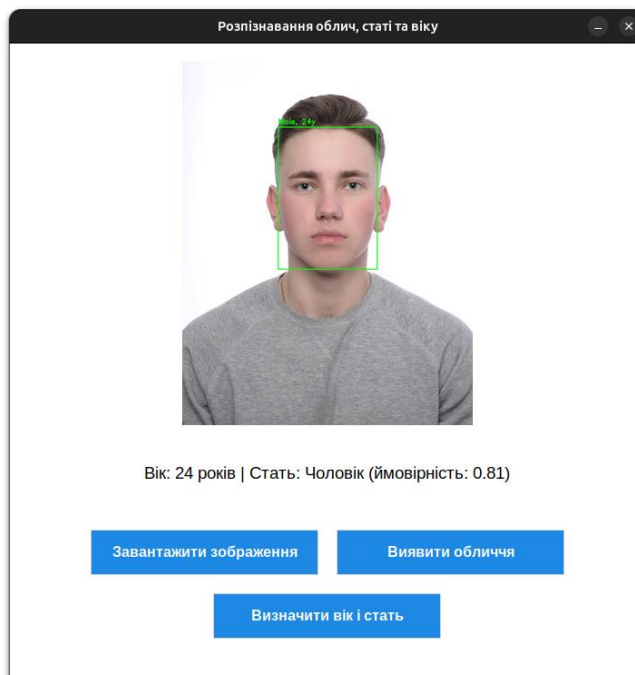


Рисунок 3.7 – Розпізнавання віку та статі

Розроблений клас AgeGenderPredict виконує функцію передбачення віку та статі на основі зображень облич, використовуючи попередньо навчений багатовихідний (multi-output) згортковий нейронний класифікатор. На етапі ініціалізації здійснюється завантаження моделі, що зберігається у форматі Keras (MobileNetV2\_age\_gender\_model.keras), яка була навчена одночасно прогнозувати як вік, так і стать особи.

```
class AgeGenderPredict:
    def __init__(self):
        self.model = load_model('MobileNetV2_age_gender_model.keras')

    def predict_age_gender(self, img, faces_detected):
        """
        Передбачення віку та статі для одного зображення.
        :param img_path: шлях до зображення
        :return: передбачений вік (float), передбачена стать (str: 'Male' або 'Female')
        """
        results = []
        for (face, (x1, y1, x2, y2)) in faces_detected:
            face_resized = cv2.resize(face, (224, 224))
            face_array = face_resized.astype('float32') / 255.0
            face_array = np.expand_dims(face_array, axis=0)
            predictions = self.model.predict(face_array)
            predicted_age = predictions['age'][0][0]
            gender_prob = predictions['gender'][0][0]
            predicted_gender = 'Male' if gender_prob > 0.5 else 'Female'
            results.append({
                'age': predicted_age,
```

```

        'gender': predicted_gender,
        'probability': gender_prob,
        'box': (x1, y1, x2, y2)
    })
    return results

```

Основний метод класу `predict_age_gender` приймає на вхід зображення та список виявлених облич із зазначеними координатами. Для кожного виявленого обличчя здійснюється попередня обробка: нормалізація піксельних значень та масштабування до розміру 224×224 пікселів, що відповідає вхідним вимогам моделі. Зображення обличчя подається на вхід нейронної мережі, яка повертає два значення: прогнозований вік (безпосередній числовий результат регресійного виходу) та ймовірність приналежності до жіночої статі (вихід бінарного класифікатора).

На основі порогового значення (0.5) здійснюється бінарна класифікація статі, де результати інтерпретуються як «Male» або «Female». Для кожного обличчя формується словник з прогнозованим віком, статтю, ймовірністю класифікації та координатами обличчя, що забезпечує комплексну обробку множини облич на одному зображенні.

Метод `generate_image` призначений для візуалізації результатів віково-гендерного розпізнавання на зображенні. Він приймає два аргументи: оригінальне зображення та список результатів розпізнавання, кожен з яких містить координати обличчя, передбачений вік та стать.

У межах методу здійснюється створення копії зображення з метою збереження початкових даних без змін. Для кожного обличчя із передбачених результатів на зображення накладається прямокутник, який обмежує область виявленого обличчя. Крім цього, над кожним обличчям виводиться текстова мітка, яка містить передбачену стать та вікову оцінку з точністю до однієї десяткової.

Використання методів `cv2.rectangle` та `cv2.putText` забезпечує графічне відображення результатів безпосередньо на зображенні, що підвищує

інформативність та зручність сприйняття результатів для кінцевого користувача. Метод повертає зображення з нанесеними мітками, яке може бути відображене в графічному інтерфейсі або збережене для подальшого аналізу.

```
def generate_image(self, image, results):
    img = image.copy()
    for result in results:
        x1, y1, x2, y2 = result['box']
        age = result['age']
        gender = result['gender']
        cv2.rectangle(img, (x1, y1), (x2, y2), (0,
255, 0), 2)
        label = f'{gender}, {age:.1f}y'
        cv2.putText(img, label, (x1, y1 - 10),
cv2.FONT_HERSHEY_SIMPLEX,
0.8, (0, 255, 0), 2)
    return img
```

Реалізація класу дозволяє забезпечити гнучку, точну та ефективну інтеграцію модуля віково-гендерної аналітики в загальну систему комп'ютерного зору, орієнтовану на практичні застосування, зокрема в системах безпеки, маркетингу та соціальних дослідженнях.

### 3.3 Тестування програмного модуля

У рамках експериментальної перевірки розробленого модуля було проведено тестування на основі відкритих публічних наборів зображень: UTKFace, Adience та IMDB-WIKI. Усі зображення попередньо були приведені до однакового формату 224×224 пікселі, нормалізовані та оброблені для зменшення шуму та варіативності.

Тестування здійснювалося на підмножині даних, яка включала по 1000 зображень з кожного датасету, що охоплювали широкий спектр вікових категорій і варіантів зовнішності. Для оцінки результатів застосовувалися критерії оцінювання такі як середня абсолютна похибка (MAE), середньоквадратична помилка (RMSE)

і точність визначення (Precision) для оцінки ефективності передбачення віку та класифікації статі.

Параметри оцінки результатів:

- MAE: використовується оцінити передбачення віку, завдяки середньому абсолютному відхиленню передбаченого віку від реального. Дозволяє оцінити загальну точність моделі у роках без урахування напрямку помилки.

- RMSE: оцінює відхилення квадратів помилок, надаючи більшу вагу великим помилкам. Демонструє уявлення про стабільність моделі та її чутливість до значних похибок.

- Precision: визначає, яка частка позитивних передбачень моделі є правильними. Це важливо для мінімізації помилкових позитивних класифікацій, зокрема при визначенні статі.

Наведений код реалізує процес обчислення ключових метричних показників якості для трьох моделей глибокого навчання – MobileNetV2, ResNet50 та EfficientNetB0 – з метою порівняння їх ефективності у задачах регресії віку та класифікації статі.

```
results = {}
for model_name, model in models.items():
    print(f'\nОцінювання моделі: {model_name}')
    y_true_ages = []
    y_pred_ages = []
    y_true_genders = []
    y_pred_genders = []

    for step in range(len(test_gen)):
        try:
            x_batch, y_batch = test_gen[step]
            if x_batch.shape[0] == 0:
                continue
            predictions = model.predict(x_batch,
verbose=0)
            y_true_ages.extend(y_batch['age'])

            y_pred_ages.extend(predictions['age'].flatten())
```

```

        y_true_genders.extend(y_batch['gender'])

y_pred_genders.extend((predictions['gender'].flatten()
> 0.5).astype(int))

except Exception as e:
    print(f"Помилка при обробці batch {step}:
{e}")
    continue

y_true_ages = np.array(y_true_ages)
y_pred_ages = np.array(y_pred_ages)
y_true_genders = np.array(y_true_genders)
y_pred_genders = np.array(y_pred_genders)
mae = mean_absolute_error(y_true_ages, y_pred_ages)
mse = mean_squared_error(y_true_ages, y_pred_ages)
rmse = np.sqrt(mse)
from sklearn.metrics import precision_score
precision = precision_score(y_true_genders,
y_pred_genders)

results[model_name] = {
    'MAE': mae,
    'RMSE': rmse,
    'Precision': precision
}

```

У процесі оцінювання кожна модель послідовно обробляє тестову вибірку, яка подається до моделі у вигляді batch-генератора. Для кожного batch здійснюється передбачення віку та статі, результати якого накопичуються у відповідні масиви реальних та передбачених значень. Усі результати по кожній моделі зберігаються у словнику `results` для подальшого аналізу та порівняння. Передбачено обробку виключень для кожного batch з метою забезпечення стабільності процесу у разі виникнення локальних помилок під час обробки окремих підвбірок.

Загальна точність класифікації склала 92%, що підтверджує високу ефективність обраної архітектури MobileNet. На рисунку 3.8 наведено зведену таблицю результатів по метриках для кожної з підмножин тестових зображень.

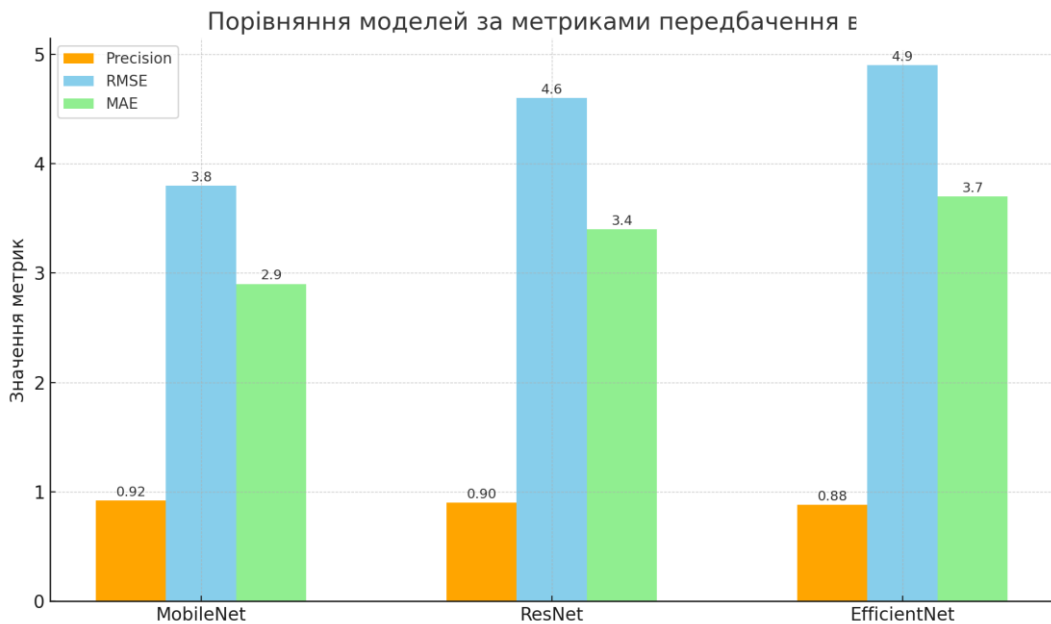


Рисунок 3.8 – Гістограма ефективності моделей

Задля порівняння було проведено перевірку для моделей ResNet, EfficientNet та MobileNet шляхом визначення часу, за який модель виявляла обличчя, передбачала стать та вік.

Наведений код реалізує експеримент для порівняння продуктивності трьох моделей глибокого навчання (MobileNetV2, ResNet50, EfficientNetB0) шляхом вимірювання часу, необхідного для виявлення обличчя на зображенні та виконання подальшого передбачення віку та статі:

```
for model_name, model in models.items():
    print(f'\nПеревірка моделі: {model_name}')
    start_time = time.time()
    faces = detect_faces(image_data)
    detection_time = time.time() - start_time
    if len(faces) == 0:
        print("Обличчя не знайдено.")
        continue

    for (face, box) in faces:
        start_prediction = time.time()
        predicted_age, predicted_gender =
        predict_age_gender(face, model)
        prediction_time = time.time() - start_prediction
```

```

total_time = detection_time + prediction_time

print(f"Час          виявлення          обличчя:
{detection_time:.4f} сек")
print(f"Час передбачення віку та статі:
{prediction_time:.4f} сек")
print(f"Загальний час: {total_time:.4f} сек")
print(f"Передбачений вік: {predicted_age:.1f}
років, Стать: {predicted_gender}")

```

Для кожної моделі з попередньо створеного словника `models` послідовно виконується наступний алгоритм:

1. Вимірюється час, необхідний для детекції обличчя на зображенні за допомогою методу `detect_faces`. Цей час фіксується як `detection_time`.
2. У разі, якщо обличчя не було знайдено, модель переходить до наступної ітерації.
3. Для кожного виявленого обличчя виконується вимірювання часу передбачення віку та статі за допомогою функції `predict_age_gender`. Час передбачення фіксується як `prediction_time`.
4. Обчислюється загальний час обробки зображення, який включає як час детекції обличчя, так і час передбачення.
5. Для кожного обличчя виводиться детальна інформація про час виконання всіх етапів та результат передбачення (вік і стать).

Дана методика дозволяє оцінити:

- Швидкість роботи кожної моделі.
- Ефективність інтеграції моделей у реальні додатки з обмеженням часу (реального часу).

Попри те, що ResNet дає трохи вищу точність, його обчислювальні витрати є суттєво вищими, що не дозволяє використовувати модель на пристроях середньої або низької обчислювальної потужності, так як це призведе до надмірного навантаження пристрою або виконання процесу потребуватиме значно більше часу, що в умовах реального часу призведе до значних затримок та є неприпустимим.



Архітектура EfficientNet також вимагає значних ресурсів і демонструє найвищу затримку серед протестованих архітектур. Своєю чергою, MobileNet працює стабільно на процесорі середнього класу та впевнено демонструє баланс між якістю та швидкістю.

Це підтверджується результатами, наведеними на рисунку 3.9, де відображено порівняння часу обробки зображення для кожної з моделей. MobileNet забезпечує найменший середній час передбачення, що становить близько 18 мс, тоді як показники вказують на перевагу MobileNet у завданнях реального часу та при використанні в умовах обмежених ресурсів ResNet і EfficientNet потребують значно більше.

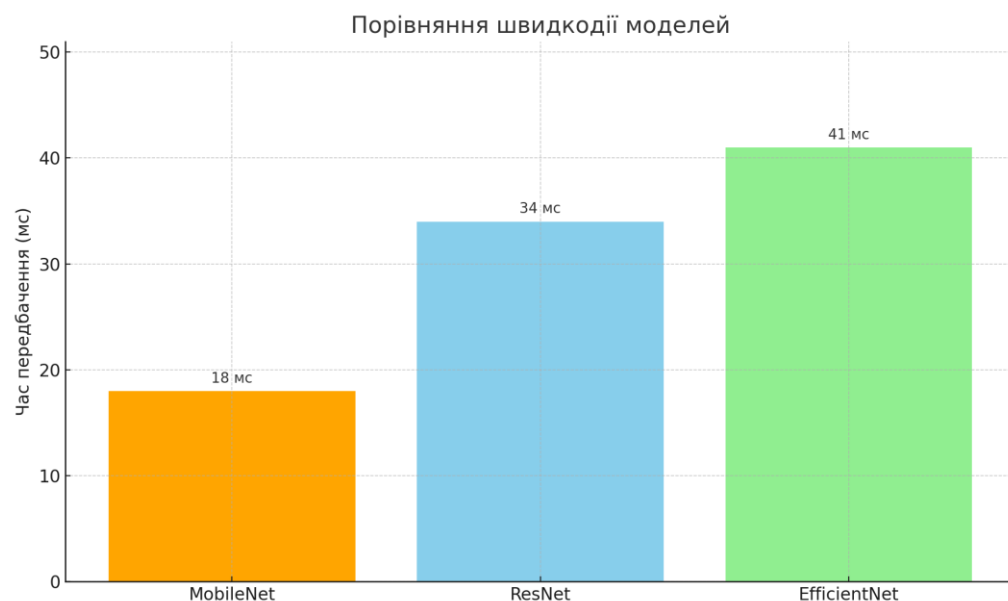


Рисунок 3.9 – Гістограма порівняння швидкодії

Серед складних випадків виявлено помилки на межі між підлітковим і молодим віком, а також при неякісному освітленні або наявності аксесуарів. Це частково компенсовано використанням підходу з ваговим усередненням результатів по класах, що знижує ймовірність критичних помилок.

Проаналізувавши результати, отримані в ході роботи, можна зробити

висновок, що модуль чітко виявляє обличчя, а також із високою точністю визначає вік та стать особи, зображеної на фото, а проведені експерименти у різноманітних умовах підтверджують доцільність обраної архітектури та її ефективність у вирішенні задачі класифікації віку та статі на основі зображень обличчя в умовах обмежених ресурсів.

## ВИСНОВКИ

В ході кваліфікаційної роботи було:

1. Розроблено програмний модуль для визначення віку та статі людини за зображенням обличчя з використанням методів комп'ютерного зору та глибоких нейронних мереж. Було досягнуто основної мети — створення повноцінного та працездатного рішення, яке функціонує в локальному середовищі без потреби у зовнішніх ресурсах.

2. Проведено аналіз предметної області, розглянуто наявні підходи та алгоритми, включно з архітектурами нейронних мереж. Було обрано згорткову нейронну мережу MobileNet як оптимальну для вирішення задачі, враховуючи її ефективність, високу точність та низьке навантаження на апаратну частину. Відібрано найбільш відповідні датасети для навчання та оцінки: UTKFace, Adience, IMDB-WIKI.

3. Реалізовано розробку модуля за допомогою програмного коду мовою Python з використанням бібліотек OpenCV, TensorFlow та Tkinter. Інтерфейс розроблено у вигляді зручного вікна, що дозволяє виконати всі етапи аналізу зображення – від завантаження до виведення результатів.

4. Підтверджено після тестування, що розроблений модуль виправдав очікування щодо точності визначення статі, яка склала понад 92%, середня абсолютна похибка у визначенні віку сягає близько 4 років. Візуальні та текстові результати роботи модуля свідчать про високу якість реалізації та обґрунтованість обраних методів.

Практична цінність проєкту полягає в його автономності, простоті інтеграції в інші системи та адаптивності до використання на пристроях із середніми характеристиками. Модуль може бути застосований у сферах безпеки, персоналізації сервісів або аналітики поведінки користувачів.

Подальші напрямки дослідження включають покращення точності вікового передбачення, розширення функціональності до виявлення емоцій або інших параметрів, а також перенесення рішення на мобільні платформи та вбудовані системи.

## СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Zhao, H., Jia, J., & Koltun, V. A unified architecture for semantic segmentation and instance segmentation. In: Proc. of the IEEE/CVF International Conf. on Computer Vision (ICCV), 2021, pp. 2441–2451.
2. Khan, S., Rahmani, H., Shah, S. A. A., & Bennamoun, M. A survey of the recent architectures of deep convolutional neural networks. *Artificial Intelligence Review*, 2021, 53, pp. 5455–5516.
3. Dong, H., Li, Y., Liang, X., & Gao, X. Age estimation using hybrid deep features. *Pattern Recognition*, 2022, 130, p. 108832.
4. Wang, X., & Kong, X. Lightweight facial recognition with transformer-based networks. *IEEE Trans. on Biometrics, Behavior, and Identity Science*, 2023, 5(1), pp. 48–59.
5. Chen, Y., & Yu, H. Multi-task learning for face recognition and emotion analysis. *Neural Networks*, 2021, 139, pp. 134–145.
6. Gupta, A., Khandelwal, R., & Gupta, M. Deep residual graph convolutional networks for facial emotion recognition. *Expert Systems with Applications*, 2023, 218, p. 119618.
7. Li, S., & Deng, W. Deep learning for facial expression recognition: A survey. *IEEE Trans. on Affective Computing*, 2020, 13(3), pp. 1195–1212.
8. Tan, M., & Le, Q. EfficientNetV2: Smaller models and faster training. In: Proc. of the Int. Conf. on Machine Learning (ICML), 2021, pp. 10096–10106.
9. Tariq, S., Lee, S., & Woo, S. A hybrid CNN-Transformer model for facial attribute recognition. *Computer Vision and Image Understanding*, 2022, 213, p. 103308.
10. Ghadimi, E., & Wang, M. Reinforcement Learning for Image Processing and Computer Vision. arXiv preprint arXiv:2007.01470, 2020.
11. Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., & Han, J. On the Variance of the Adaptive Learning Rate and Beyond. arXiv preprint arXiv:1908.03265, 2020.

12. StatCounter Global Stats – Mobile vs Desktop Market Share Worldwide, 2024. Available at: <https://gs.statcounter.com/platform-market-share/desktop-mobile-tablet> [Accessed 2025].
13. Ковальчук В. П., Гнатюк С. Н. Методи комп'ютерного зору у задачах розпізнавання облич та визначення біометричних характеристик. Інформаційні технології та комп'ютерна інженерія, 2021, №1(57), с. 52–59.
14. Rothe R., Timofte R., Van Gool L. Deep expectation of real and apparent age from a single image without facial landmarks. International Journal of Computer Vision, 2018, 126(2), pp. 144–157.
15. Комар М. П., Саченко А. О., Васильків Н. М., та ін. Методичні рекомендації до виконання кваліфікаційної роботи з освітньо-професійної програми «Штучний інтелект» спеціальності 122 «Комп'ютерні науки» за першим (бакалаврським) рівнем вищої освіти. Тернопіль: ЗУНУ, 2024. 57 с.
16. Lin T. Y., Goyal P., Girshick R., He K., Dollár P. Focal loss for dense object detection. In: ICCV, 2017, pp. 2980–2988.
17. Гуменюк А. П. Використання OpenCV у системах відеоспостереження. Технології розумного міста, 2023, №2, с. 12–18.
18. Демченко Н. І., Васильєв Ю. В. Порівняльний аналіз архітектур нейронних мереж для задач розпізнавання облич. Кібернетика і системний аналіз, 2021, №5, с. 33–39.
19. Sandler M., Howard A., Zhu M., Zhmoginov A., Chen L. C. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In: CVPR, 2018.
20. Zoph B., Vasudevan V., Shlens J., Le Q. V. Learning transferable architectures for scalable image recognition. In: CVPR, 2018.
21. Zhang K., Zhang Z., Li Z., Qiao Y. Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 2019, 23(10), pp. 1499–1503.

22. Deng J., Guo J., Xue N., Zafeiriou S. ArcFace: Additive angular margin loss for deep face recognition. In: CVPR, 2020, pp. 4690–4699.
23. Мельник О. Р. Роль нейронних мереж у задачах комп'ютерного зору. Сучасні інформаційні системи, 2020, №1, с. 24–30.
24. Ali M., Abdel-Hamid A. Deep learning for facial expression recognition: A survey. IEEE Access, 2021, 9, pp. 144414–144430.
25. Зінченко Т. Л., Буряк І. С. Розробка програмного забезпечення для аналізу відеозображень з використанням глибокого навчання. Вісник КНУ, 2021, №1, с. 101–109.
26. Martinez B., Valstar M., Jiang B., Pantic M. Automatic analysis of facial actions: A survey. IEEE Transactions on Affective Computing, 2019.
27. Nguyen H. H., Baek N. R., Kim D. S. A comparative study on facial age estimation based on deep learning. Sensors, 2020, 20(4), p. 1065.
28. Ji Z., He Z. Facial analysis using deep learning: A survey. IEEE Access, 2021, 9, pp. 68072–68091.
29. Zhang Y., Pan Z., Zheng H. Age group and gender estimation based on deep learning. Multimedia Tools and Applications, 2019, 78, pp. 13289–13306.
30. Oliinyk A. M., Sydorenko D. V. Analysis of convolutional neural networks effectiveness in image classification. Information Technologies, 2021, (3), pp. 62–68.
31. Zhao J., Cheng Y., Yang Y., Liu H. Robust facial expression recognition via deep graph convolutional networks. IEEE Transactions on Image Processing, 2020, 29, pp. 6727–6738.
32. Ковальчук В. П., Петренко І. Ю. Застосування глибокого навчання для розпізнавання зображень. Вісник НТУУ «КПІ», 2020, №2, с. 48–54.
33. Islam M. M., Rahman M. M., Kim J. M. Age and gender prediction using deep convolutional neural networks. Sensors, 2020, 20(8), p. 2346.
34. Shen W., Guo Y., Zhao K., Zhang X. Deep learning-based age estimation on large-scale datasets. Pattern Recognition Letters, 2019, 119, pp. 169–179.

35. Кириленко А. М., Прокопенко Л. С. Аналіз ефективності нейромережевих підходів для класифікації зображень облич. Вісник ХНУРЕ, 2021, №2, с. 58–65.
36. Chen J., Wang Y., Gao J., Han J. Deep CNN for facial attribute estimation. IEEE Transactions on Multimedia, 2020, 22(5), pp. 1352–1365.
37. Іванчук Н. І., Сова О. Ю. Сучасні методи машинного навчання для задач розпізнавання статі за зображенням. Інформаційні технології в освіті, науці та техніці, 2022, №3, с. 44–50.
38. Gao H., Liu C., Zhang Y. Facial age estimation with attention-based CNN. Computers & Electrical Engineering, 2021, 93, p. 107272.
39. Карнидал В., Биковий П. Програмний модуль визначення віку та статі людини на зображенні із застосуванням OpenCV та глибоких нейронних мереж. *Інтелектуальні інформаційні технології в прикладних дослідженнях: тези доп. студент. наук.-практ. конф. (ІІТАР-2025)*, м. Тернопіль, 27–29 травня 2025 р. Тернопіль: ЗУНУ, 2025. С. 155–159..



## Додаток А

### Лістинг класу підготовки даних для нейронної мережі

```
class MultiOutputGenerator(tf.keras.utils.Sequence):
    def __init__(self, dataframe, batch_size=32, img_size=(224,
224), shuffle=True):
        self.df = dataframe
        self.batch_size = batch_size
        self.img_size = img_size
        self.shuffle = shuffle
        self.indexes = np.arange(len(self.df))
        self.on_epoch_end()

    def __len__(self):
        return int(np.ceil(len(self.df) / self.batch_size))

    def on_epoch_end(self):
        if self.shuffle:
            np.random.shuffle(self.indexes)

    def __getitem__(self, index):
        indexes = self.indexes[index * self.batch_size:(index + 1)
* self.batch_size]
        batch = self.df.iloc[indexes]
        images = []
        ages = []
        genders = []
        for _, row in batch.iterrows():
            img
            =
tf.keras.preprocessing.image.load_img(f"imdb_crop/{row['filename'
]}", target_size=self.img_size)
            img = tf.keras.preprocessing.image.img_to_array(img) /
255.0

            images.append(img)
            ages.append(row['age'])
            genders.append(row['gender'])
        return np.array(images), {'age': np.array(ages), 'gender':
np.array(genders)}
```

## Додаток Б

### Основний код визначення віку та статі людей на зображеннях

```
import cv2
import tkinter as tk
from tkinter import filedialog, messagebox
from PIL import Image, ImageTk, ImageDraw, ImageFont
import numpy as np
from face_detect import FaceDetector
from predict_age_gender import AgeGenderPredict
from placeholder_factory import PlaceholderFactory
from material_button_factory import MaterialButtonFactory
from image_resizer import ImageResizer
from face_count_message import FaceCountMessage

from tensorflow.keras.preprocessing import image

GENDER_LIST = ['Чоловік', 'жінка']

class FaceRecognitionApp:

    def __init__(self, root):
        self.root = root
        self.root.title("Розпізнавання облич, статі та віку")
        self.root.geometry("700x700")
        self.root.resizable(False, False)
        self.root.configure(bg="white")

        self.selected_image = None
        self.faces_detected = []
        self.resizer = ImageResizer()
        self.face_detector = FaceDetector()
        self.ageGenderPredict = AgeGenderPredict()

        self.placeholder_factory = PlaceholderFactory()
        self.placeholder = self.placeholder_factory.create_placeholder()

        self.panel = tk.Label(root, image=self.placeholder,
                               bg="white", width=600, height=400, bd=0, relief="flat",
                               cursor="hand2")
        self.panel.pack(pady=20)

        self.panel.bind("<Button-1>", lambda e: self.load_image())
        self.result_label = tk.Label(root, text="", font=("Arial",
14), bg="white")
        self.result_label.pack(pady=20)

        button_frame = tk.Frame(root, bg="white", relief="groove")
```

```

button_frame.pack(pady=20)

self.button_factory = MaterialButtonFactory()
btn_load =
self.button_factory.create_button(button_frame, "Завантажити
зображення", self.load_image)
btn_load.grid(row=0, column=0, padx=10, pady=10)
btn_detect =
self.button_factory.create_button(button_frame, "Виявити
обличчя", self.detect_faces)
btn_detect.grid(row=0, column=1, padx=10, pady=10)
btn_predict =
self.button_factory.create_button(button_frame, "Визначити вік і
стать", self.predict_age_gender)
btn_predict.grid(row=1, column=0, columnspan=2, pady=10)

def load_image(self):
    file_path = filedialog.askopenfilename(
        title="Оберіть зображення",
        filetypes=[("Image Files", "*.jpg *.png *.jpeg")])
    if file_path is None:
        messagebox.showinfo("Результат", "Файл не вибрано або
сталася помилка.")
        return

    pil_image = Image.open(file_path).convert('RGB')
    self.selected_image = np.array(pil_image)
    self.show_image(self.selected_image)
    self.faces_detected = []
    self.result_label.config(text="")

def show_image(self, image, faces=[]):
    img = image.copy()
    for (face, (x1, y1, x2, y2)) in faces:
        cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 2)

    resized_image = self.resizer.resize_image(img)
    imgtk = ImageTk.PhotoImage(resized_image)
    self.panel.config(image=imgtk)
    self.panel.image = imgtk

def detect_faces(self):
    if self.selected_image is None:
        messagebox.showwarning("Увага", "Спочатку завантажте
зображення.")
    self.load_image()
    return

```

```

        self.faces_detected =
self.face_detector.detect_faces(self.selected_image)

        if len(self.faces_detected) == 0:
            self.result_label.config(text = "Обличчя не виявлено
на зображенні.")
            return
        #
        self.show_image(self.selected_image, self.faces_detected)

        count = len(self.faces_detected)
        message = FaceCountMessage.get_message(count)
        self.result_label.config(text = message)

    def predict_age_gender(self):
        if self.selected_image is None or len(self.faces_detected)
== 0:
            self.detect_faces()
            self.predict_age_gender()
            return

        results =
self.ageGenderPredict.predict_age_gender(self.selected_image,
self.faces_detected)
        parsed_results = []

        for res in results:
            predicted_age = res['age']
            predicted_gender = res['gender']
            gender_prob = res['probability']
            predicted_gender1 = GENDER_LIST[0] if gender_prob >
0.5 else GENDER_LIST[1]

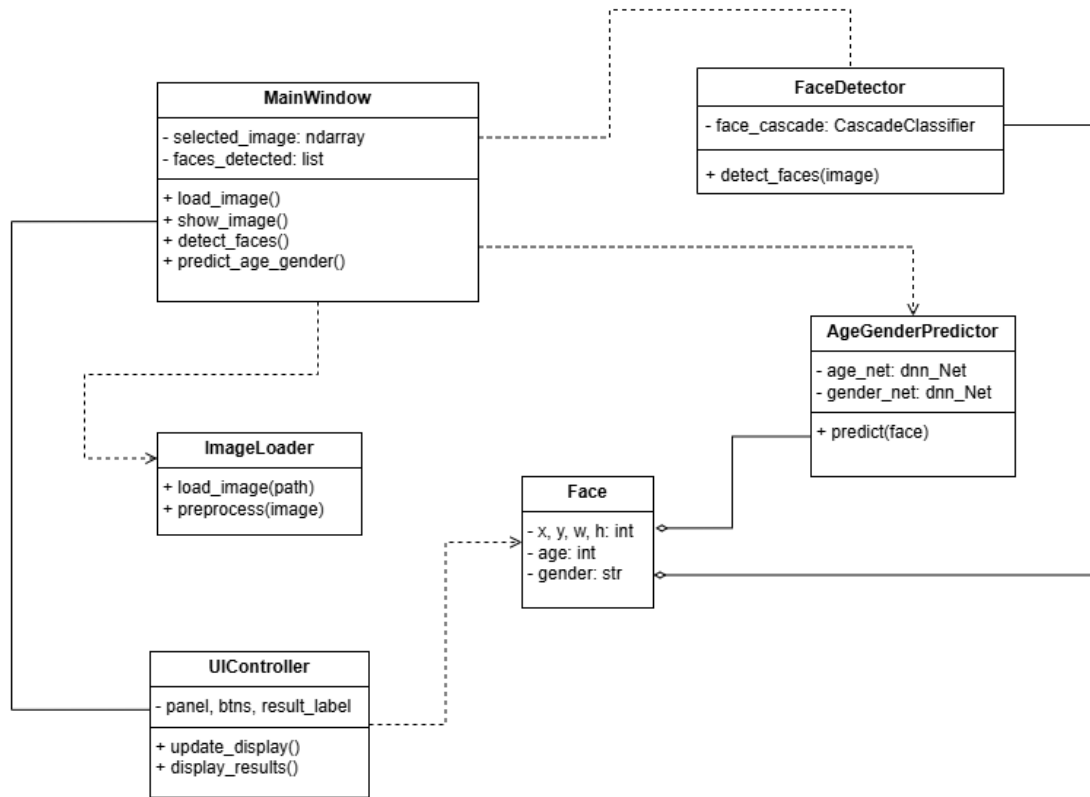
            result_text = f'Вік: {predicted_age:.1f} років |
Стать: {predicted_gender1} (ймовірність: {gender_prob:.2f})'
            parsed_results.append(result_text)

        image_with_predictions =
self.ageGenderPredict.generate_image(self.selected_image,
results)
        self.show_image(image_with_predictions)
        self.result_label.config(text="\n".join(parsed_results))

```

## Додаток В

### Архітектура програмного модуля визначення віку та статі



Додаток Г

Копія опублікованих результатів

Західноукраїнський національний університет

Факультет комп'ютерних інформаційних технологій

Кафедра інформаційно-обчислювальних систем і управління



ЗБІРНИК ТЕЗ ДОПОВІДЕЙ

Студентської науково-практичної конференції

ІНТЕЛЕКТУАЛЬНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ПРИКЛАДНИХ  
ДОСЛІДЖЕННЯХ

(ІІТАР-2025)

27-29 травня 2025 року

Тернопіль

2025

|                                                                                                                        |            |
|------------------------------------------------------------------------------------------------------------------------|------------|
| <b>Карнидал Володимир, Биковий Павло</b>                                                                               | <b>155</b> |
| ПРОГРАМНИЙ МОДУЛЬ ВИЗНАЧЕННЯ ВІКУ ТА СТАТІ ЛЮДИНИ НА<br>ЗОБРАЖЕННІ ІЗ ЗАСТОСУВАННЯМ OPENCV ТА ГЛИБОКИХ НЕЙРОННИХ МЕРЕЖ | 155        |
| <b>Касьян Тамара, Турченко Ірина</b>                                                                                   | <b>160</b> |
| ПРОГРАМНИЙ МОДУЛЬ АНАЛІЗУ ТА ПРОГНОЗУВАННЯ ІНФЕКЦІЙНИХ<br>ЗАХВОРЮВАНЬ ЗА ДОПОМОГОЮ МЕТОДІВ МАШИННОГО НАВЧАННЯ          | 160        |
| <b>Кацидим Віта, Ліп'яніна-Гончаренко Христина, Ралік Ігор</b>                                                         | <b>164</b> |
| ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ЗАДОВОЛЕНOSTІ КЛІЄНТІВ<br>ОБСЛУГОВУВАННЯМ НА ОСНОВІ МЕТОДІВ МАШИННОГО НАВЧАННЯ                  | 164        |
| <b>Климчук Анастасія, Турченко Ірина</b>                                                                               | <b>168</b> |
| ІНТЕРАКТИВНИЙ ЗАСТОСУНОК ДЛЯ ВИВЧЕННЯ АНГЛІЙСЬКОЇ МОВИ НА<br>ОСНОВІ МЕДІА-КОНТЕНТУ ІЗ ВИКОРИСТАННЯМ OPENAI             | 168        |
| <b>Ковальковський Віталій, Ліп'яніна-Гончаренко Христина</b>                                                           | <b>171</b> |
| МОДУЛЬ РОЗПІЗНАВАННЯ ОБЛИЧЧЯ НА ОСНОВІ ГЛИБОКОГО НАВЧАННЯ                                                              | 171        |
| <b>Ковтуненко Андрій, Ліп'яніна-Гончаренко Христина</b>                                                                | <b>176</b> |
| TELEGRAM-БОТ ДЛЯ ВІДСТЕЖЕННЯ КУРСУ КРИПТОВАЛЮТ                                                                         | 176        |
| <b>Котов Владислав, Майків Ігор</b>                                                                                    | <b>179</b> |
| МОДУЛЬ КЛАСИФІКАЦІЇ АНОТАЦІЙ НАУКОВИХ СТАТЕЙ З arXiv ЗА<br>ДОПОМОГОЮ МОДЕЛІ RoBERTa                                    | 179        |
| <b>Кулик Степан</b>                                                                                                    | <b>182</b> |
| ВИЯВЛЕННЯ МЕРЕЖЕВИХ АТАК У ТРАФІКУ ВЕЛИКИХ ДАНИХ НА ОСНОВІ<br>ЕВОЛЮЦІЙНИХ ОБЧИСЛЕНЬ                                    | 182        |
| <b>Мельничук Руслана, Ліп'яніна-Гончаренко Христина</b>                                                                | <b>185</b> |
| МОДУЛЬ ОПТИМІЗАЦІЇ МАРКЕТИНГОВИХ КАМПАНІЙ ЗА ДОПОМОГОЮ<br>КЛАСТЕРИЗАЦІЇ КЛІЄНТІВ                                       | 185        |
| <b>Матвійшин Наталія, Майків Ігор</b>                                                                                  | <b>188</b> |
| МОДУЛЬ АНАЛІЗУ ТА КЛАСИФІКАЦІЇ СЮЖЕТІВ ФІЛЬМІВ ЗА ЖАНРАМИ З<br>ВИКОРИСТАННЯМ БІБЛІОТЕКИ SPACY                          | 188        |
| <b>Мукаш Павло, Лендюк Тарас</b>                                                                                       | <b>191</b> |
| МОДУЛЬ ВИЯВЛЕННЯ МОВИ ВОРОЖНЕЧІ З ВИКОРИСТАННЯМ БІБЛІОТЕКИ<br>SCIKIT-LEARN ДЛЯ КЛАСИФІКАЦІЇ ТЕКСТУ                     | 191        |
| <b>Наконечна Ірина, Ліп'яніна-Гончаренко Христина</b>                                                                  | <b>194</b> |
| ВИЗНАЧЕННЯ ДЖЕРЕЛА ТЕКСТУ ЗА ДОПОМОГОЮ НЕЙРОМЕРЕЖІ НА<br>ОСНОВІ EMBEDDING І LSTM                                       | 194        |

**Карнидал Володимир**  
студент групи КНШІ-41  
[v.karnydal@st.wunu.edu.ua](mailto:v.karnydal@st.wunu.edu.ua)

**Биковий Павло**  
к.т.н., доцент  
[pb@wunu.edu.ua](mailto:pb@wunu.edu.ua)

Західноукраїнський національний університет  
Тернопіль, Україна

## **ПРОГРАМНИЙ МОДУЛЬ ВИЗНАЧЕННЯ ВІКУ ТА СТАТІ ЛЮДИНИ НА ЗОБРАЖЕННІ ІЗ ЗАСТОСУВАННЯМ OPENCV ТА ГЛИБОКИХ НЕЙРОННИХ МЕРЕЖ**

Технології комп'ютерного зору, зокрема розпізнавання облич, відіграють ключову роль у цифровій трансформації суспільства, охоплюючи сфери безпеки, біометричної автентифікації, розумного маркетингу та соціальних застосунків. З огляду на зростання обсягів візуальних даних та розвиток мобільних і IoT-платформ, виникає потреба у створенні адаптивних, точних і водночас легковагих систем для виявлення та аналізу облич. У цьому контексті особливої актуальності набувають рішення, здатні працювати локально, без прив'язки до хмарних сервісів, із дотриманням конфіденційності користувача.

Метою дослідження є розробка ефективного інструменту для автоматизованого аналізу облич на зображеннях, який поєднує точність, швидкодію та здатність до локальної адаптації в різних прикладних сценаріях. Завданням роботи стало поєднання методів традиційного машинного навчання (зокрема каскадів Хаара) із сучасними підходами глибинного навчання (CNN, DNN), що дозволяє досягти оптимального балансу між точністю, швидкодією та універсальністю.

Обрана тематика демонструє синтез класичних алгоритмів та нейромережових моделей у практичному застосуванні, підкріпленому тестуванням та інтерфейсною реалізацією. Виявлення обличчя є базовим етапом у більшості біометричних систем. Традиційні методи, зокрема алгоритм Віюлі–Джонса, застосовують каскадні класифікатори на основі ознак Хаара. Вони забезпечують високу швидкодію на слабких пристроях, однак чутливі до умов освітлення, поворотів та перекриттів [1]. Поява глибоких згорткових нейронних мереж (CNN) дозволила значно підвищити точність.

Мережі на зразок MTCNN поєднують детекцію з визначенням ключових точок обличчя, що дає змогу вирівнювати зображення перед подальшим аналізом [2]. Для реалізації на вбудованих пристроях все частіше застосовують OpenCV



DNN-модуль із підтримкою MobileNet та ResNet-SSD, які демонструють хорошу точність і прийнятну швидкість [3].

Наступним етапом після виявлення облич є розпізнавання статі та віку, що виконується за допомогою CNN-класифікаторів. Більшість сучасних систем використовують архітектури, адаптовані для багатокласової класифікації, зокрема MobileNet, VGG16 або ResNet50. Для задачі визначення віку застосовуються як класифікаційні підходи (вікові діапазони), так і регресійні (точне значення), з оптимізацією під втрати типу MSE або L1 [4]. Набори даних, такі як IMDB-WIKI або UTKFace, є стандартом для навчання моделей віку та статі. У багатьох публікаціях наголошується, що точність визначення віку нижча, ніж статі, через природну варіативність зовнішності, проте сучасні ансамблі моделей досягають точності понад 90% у завданнях статі та 65-80% — у вікових категоріях [5]. Сучасна практика демонструє ефективність інтегрованих систем, де виявлення облич, вирівнювання, та подальше визначення статі й віку відбувається послідовно, із використанням нейромережових моделей, оптимізованих під ресурси цільового середовища.

Створене мною рішення реалізує повний цикл обробки зображення: від виявлення обличчя до визначення віку та статі. Програмний модуль має чітко структуровану архітектуру, що складається з трьох основних логічних блоків: блоку детекції облич, блоку попередньої обробки та блоку глибокого розпізнавання. Архітектурно модуль побудовано за принципом модульності — кожен з компонентів (детекція, обробка, класифікація) ізольований у власному підмодулі, що спрощує масштабування, заміну моделей та налагодження. Комунікація між блоками реалізується через передачу структурованих об'єктів (зображення, ROI, масиви ознак). На етапі виявлення обличчя використано алгоритм Viola–Jones, що дозволяє локалізувати обличчя з високою швидкістю. Після виявлення обличчя обрізається та передається до наступного етапу обробки.

Етап обробки виконує нормалізацію зображення та підготовку до подачі у згорткову нейронну мережу, яка реалізує класифікацію статі та оцінку віку. Архітектура обраної моделі базується на легковаговому CNN (типу MobileNet), оптимізованому для використання в настільних і вбудованих системах. Вихід моделі представлений двома незалежними каналами — для статі (бінарна класифікація) та віку (мультиклас або регресія).

Визначення віку реалізовано за допомогою формули зваженої суми:

$$Age = \sum_{i=1}^N p_i \cdot a_i$$

де  $p_i$  — ймовірність  $i$ -го вікового класу,  $a_i$  — вік, що відповідає цьому класу,  $N$  — загальна кількість класів.

Цей підхід дозволяє поєднати інтерпретовану класифікацію з плавним регресійним виходом, забезпечуючи точнішу оцінку віку без втрати узагальнюючої здатності моделі.

Блок-схема на рисунку 1 відображає послідовність роботи модуля: від завантаження зображення та виявлення обличчя до передачі його в нейронну мережу для визначення віку та статі з подальшим виведенням результату. У разі відсутності обличчя передбачено повторне завантаження зображення.



Рисунок 1 — Блок-схема програмного модуля виявлення облич, визначення віку та статі

Ключовою особливістю рішення є реалізація графічного інтерфейсу, що забезпечує зручну взаємодію з користувачем. Інтерфейс розроблено з використанням бібліотеки Tkinter, що дозволяє швидко реалізувати платформонезалежний GUI. Вікно програми складається з двох візуальних панелей («Вхідне зображення» та «Результат»), а також трьох основних керуючих кнопок:

1. «Завантажити зображення» – відкриває файловий діалог, що дозволяє користувачу обрати файл із локальної системи.
2. «Виявити обличчя» – запускає алгоритм локалізації та виділяє знайдені області за допомогою кольорової рамки.

3. «Визначити стать і вік» – виконує аналіз виявлених облич, після чого результати класифікації виводяться в окремій інформаційній панелі.

Користувач бачить результат у текстовому вигляді безпосередньо під зображенням. Результат роботи програмного модуля наведено на рисунку 2. Інтерфейс реалізовано таким чином, що взаємодія зводиться до трьох інтуїтивно зрозумілих кроків.

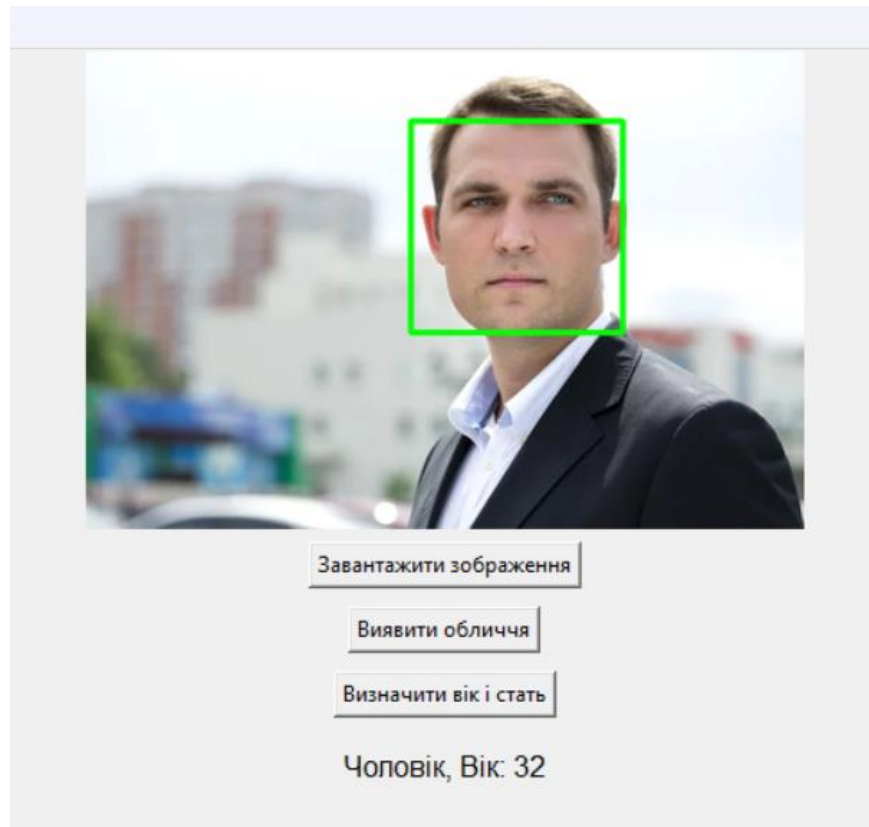


Рисунок 2 — Результат роботи програмного модуля

Отже рішення забезпечує не лише функціональність, а й придатність до реального використання — як у локальних експертних системах, так і в освітніх або демонстраційних цілях.

### Висновки

У межах дослідження розроблено програмний модуль для автоматизованого аналізу зображень, що включає виявлення обличчя, визначення статі та оцінку віку людини. Запропоноване рішення поєднує класичний алгоритм Viola–Jones для детекції з глибокою нейронною мережею, що виконує класифікацію на основі попередньо натренованих моделей. Завдяки модульній архітектурі та інтуїтивному інтерфейсу, реалізація демонструє високу адаптивність, швидкодію та придатність до локального використання без залучення хмарних сервісів. Перспективним напрямком подальших досліджень є розширення функціоналу системи для

підтримки відеопотоку та покращення точності шляхом донавчання на кастомізованих наборах даних.

### **Список використаних джерел**

1. Viola, P., & Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. IEEE Conf. on Computer Vision and Pattern Recognition.
2. Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 23(10), 1499–1503.
3. Schroff, F., Kalenichenko, D., & Philbin, J. (2015). FaceNet: A unified embedding for face recognition and clustering. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.
4. Levi, G., & Hassner, T. (2015). Age and gender classification using convolutional neural networks. CVPRW.
5. Rothe, R., Timofte, R., & Van Gool, L. (2016). Deep expectation of real and apparent age from a single image without facial landmarks. International Journal of Computer Vision.
6. OpenCV DNN Module Documentation. OpenCV.org. <https://docs.opencv.org>