

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»
ФАКУЛЬТЕТ БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

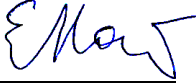
(повна назва інституту/факультету)

КАФЕДРА БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

(повна назва кафедри)

«На правах рукопису»
УДК 004.891.3 + 616.1

«До захисту допущено»:
Завідувач кафедри БМК


(підпис) Євген НАСТЕНКО
(ім'я, ПРІЗВИЩЕ)

« 05 » грудня 2022р.


Магістерська дисертація
на здобуття ступеня магістра
за освітньо-професійною програмою
«Комп'ютерні технології в біології та медицині»
зі спеціальності **122 «Комп'ютерні науки»**

на тему: **Система детекції та розпізнавання облич за допомогою методу
k - найближчих сусідів**

Виконав: студент II курсу, групи БС-11мп

ІВАНШИН ІВАН ВОЛОДИМИРОВИЧ


(прізвище, ім'я, по батькові)


(підпис)

Науковий керівник:

ст. викл. каф. БМК, Бовсуновська Катерина Сергіївна

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


(підпис)

Рецензент:

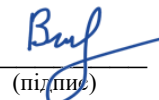
доц. каф. БМІ, к.т.н. Рудніцька Олена Володимирівна

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

(підпис)

Засвідчую, що у цій магістерській дисертації
немає запозичень з праць інших авторів без
відповідних посилань.

Студент


(підпис)

Київ – 2022 року

**Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»**

Інститут (факультет) _____

БІОМЕДИЧНОЇ ІНЖЕНЕРІЇ

(повна назва)

Кафедра _____

БІОМЕДИЧНОЇ КІБЕРНЕТИКИ

(повна назва)

Рівень вищої освіти – другий (магістерський)

спеціальність _____

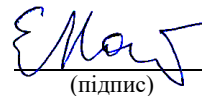
122 «Комп'ютерні науки»

Освітньо-професійна програма «Комп'ютерні технології в біології та
медицині»

(код і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри БМК



(підпис)

Євген НАСТЕНКО

(ім'я, ПРІЗВИЩЕ)

«31» серпня 2022р.

ЗАВДАННЯ

на магістерську дисертацію студенту

Іванішину Івану Володимировичу

(прізвище, ім'я, по батькові)

1. Тема дисертації **Система детекції та розпізнавання облич за допомогою методу k - найближчих сусідів**

науковий керівник дисертації

Бовсуновська Катерина Сергіївна, ст.викл.

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

затверджені наказом по університету від «09» листопада 2022 р. № 4133-с

2. Термін подання студентом дисертації: 01-05 грудня 2022 року

3. Об'єкт дослідження: розпізнавання облич

4. Вихідні дані: _____

5. Перелік завдань, які потрібно розробити: **1. Проаналізувати існуючі методи для виявлення та розпізнавання облич. 2. Реалізувати механізм детекції облич у відеопотоках. 3. Дослідити підхід для ідентифікації облич у відеопотоках. 4. Розробити систему для детекції та розпізнавання облич.**

6. Орієнтовний перелік графічного (ілюстративного) матеріалу: **20 рисунків, 11 таблиць, презентація на 13 слайдів.**

7. Орієнтовний перелік публікацій: **публікація за темою «Використання методів Віоли-Джонса і k-найближчих сусідів для детекції та розпізнавання облич».**

8. Консультанти розділів дисертації:

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Магістерської дисертації			

9. Дата видачі завдання: **31 серпня 2022 р.**

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1	Отримання завдання на МД	31 серпня 2022р.	<i>виконано</i>
2	Практика	з 01.09.2022р. по 28.10.2022р.	<i>виконано</i>
3	Вставлення результатів практики в розділи МД	5 листопада 2022р.	<i>виконано</i>
4	Аналіз предметної області	6 листопада 2022р.	<i>виконано</i>
5	Реалізація механізму детекції облич	7-11 листопада 2022р.	<i>виконано</i>
6	Розробка модуля для ідентифікації облич	12-18 листопада 2022р.	<i>виконано</i>
7	Імплементація системи для детекції та розпізнавання облич	19-25 листопада 2022р.	<i>виконано</i>
8	Стартап аналіз проекту магістерської дисертації	26-29 листопада 2022р.	<i>виконано</i>
9	Складання висновків до результатів виконання магістерської дисертації	30 листопада 2022р.	<i>виконано</i>
10	Подання в електронному вигляді МД та анотації до неї на перевірку нормоконтролера та UNICHECK .	1-3 грудня 2022р	<i>виконано</i>
11	Передзахист МД та допуск до захисту дисертації	4-6 грудня 2022р.	<i>виконано</i>
12	Подання МД рецензенту. Отримання рецензії.	7-15 грудня 2022р.	<i>виконано</i>
13	Подання пакету документів по МД до захисту в ЕК	15-18 грудня 2022р.	<i>виконано</i>
14	Захист МД в ЕК	19 грудня 2022р.	<i>виконано</i>

Студент

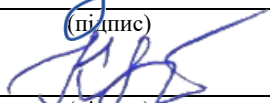


(підпис)

Іван ІВАНІШИН

(ім'я, ПРІЗВИЩЕ)

Науковий керівник

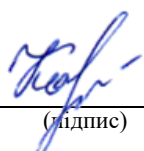


(підпис)

Катерина БОВСУНОВСЬКА

(ім'я, ПРІЗВИЩЕ)

Нормоконтролер



(підпис)

Галина КОРНІЄНКО

(ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Магістерська дисертація за темою «Система детекції та розпізнавання облич за допомогою методу k - найближчих сусідів» виконана студентом кафедри біомедичної кібернетики ФБМІ Іванішиним Іваном Володимировичем зі спеціальності 122 «Комп'ютерні науки» за освітньо-професійною програмою «Комп'ютерні технології в біології та медицині», та складається зі: вступу; 4 розділів («Аналітичний огляд літературних джерел», «Теоретична частина», «Аналітична частина», «Практична частина»), розділу з розрахунком стартап-проекту, висновків до кожного з цих розділів; загальних висновків; списку використаних джерел, який налічує 43 найменування. Загальний обсяг роботи 88 сторінки.

Актуальність теми. Останнім часом широкого поширення набуває відеоаналітика - технологія, котра використовує методи комп'ютерного зору з метою автоматизованого збору та аналізу різноманітної інформації на основі послідовності зображень, отриманих з відеокамер в реальному часі або із записів відео. Дану технологію можна використовувати у системах відеоспостереження, системах безпеки, транспорті чи торгівлі. Рішення цієї проблеми, перш за все, має пряме застосування в системах ідентифікації особистості та контролю доступу.

Мета і завдання дослідження. Метою роботи є покращення ефективності розпізнавання особи на відеозображенні у режимі реального часу, з використанням методу k-найближчих сусідів та розробки системи розпізнавання облич. Завдання для вирішення поставленої задачі:

1. Огляд і аналіз існуючих рішень.
2. Дослідження методів і алгоритмів, вимог для вирішення поставленого завдання.
3. Розробка окремих модулів для детекції та ідентифікації облич.
4. Створення прикладної програми – системи розпізнавання облич.
5. Тестування роботи програмного забезпечення.

Об'єкт дослідження. Процес розпізнавання образів.

Предмет дослідження. Система для детекції та розпізнавання облич.

Методи дослідження. Методи розпізнавання образів для ідентифікації людини у відеозображенні.

Публікації. За результатами виконаної роботи було опубліковано 1 наукову статтю: Іванішин І. В. Використання методів Віоли-Джонса і k-найближчих сусідів для детекції та розпізнавання облич // Міжнародний науковий журнал "Інтернаука". — 2022. — №11. <https://doi.org/10.25313/2520-2057-2022-11-8281>

Ключові слова. комп'ютерний зір, відеопотік, виявлення облич, розпізнавання облич.

Бібліографічний опис МД.

Іванішин І. В. Система детекції та розпізнавання облич за допомогою методу k - найближчих сусідів: магістерська дис. : 122 Комп'ютерні науки / Іванішин Іван Володимирович. – Київ, 2022. – 88 с.

ABSTRACT

Master's thesis on "Face detection and recognition system using the k-nearest neighbors method" is executed by the student of the department of biomedical cybernetics (Faculty of Biomedical Engineering) Ivanishyn Ivan Volodymyrovych in the specialty 122 "Computer science" on the educational and professional program "Computer and technology" consists of: introduction; 4 sections ("Analytical review of literature", "Theoretical part", "Analytical part", "Practical part"); section with a startup calculation; conclusions; references. The total volume of the work is 88 pages.

Relevance of the topic. Recently, video analytics - a technology that uses computer vision methods for the purpose of automated collection of various information based on a sequence of images obtained from video cameras in real time - has become widespread. such technologies can be useful in video surveillance systems, security systems, commerce or transport. The solution to this problem, first of all, has a direct application in personal identification and access control systems.

Objective of the study. The *objective* of the work is to improve the effectiveness of recognizing a person on a video image in real time, using the k-nearest neighbors method and developing a face recognition system. Tasks for solving the given goal:

1. Review and analysis of existing solutions.
2. Study of methods and algorithms, requirements for solving the task.
3. Development of separate modules for detection and identification of faces.
4. Creation of an application program - face recognition system.
5. Software performance testing.

Object of study. Pattern recognition process.

Subject of study. System for detection and recognition of faces.

Research methods. Pattern recognition methods for identifying a person in a video image.

Publications. Based on the results of the work performed, a scientific article was published:

1. Ivanishyn I.V., Using the methods of Viola-Jones and k-nearest neighbors for detection and recognition of faces // International scientific journal

"Internauka". — 2022. — No. 11. <https://doi.org/10.25313/2520-2057-2022-11-8281>

Keywords. Video stream, computer vision, face detection, face recognition.

Bibliographic description of MD.

Ivanishyn, I. V. Face detection and recognition system using the k-nearest neighbors method: master's thesis. : 122 Computers of science / Ivan Volodymyrovych Ivanishyn. - Kyiv, 2022. - 88 p.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ	10
ВСТУП	11
РОЗДІЛ 1 ЛІТЕРАТУРНИЙ ОГЛЯД.....	15
1.1. Сучасний стан та тенденція розвитку технологій розпізнавання	15
1.2. Поняття комп'ютерного зору.....	19
1.3. Програмні системи розпізнавання обличчя	20
Висновки до розділу 1	26
РОЗДІЛ 2 ТЕОРЕТИЧНА ЧАСТИНА	27
2.1. Методи виявлення облич	27
2.2. Виявлення облич методом Віоли-Джонса	37
2.2.1. Інтегральне представлення	38
2.2.2. Ознаки Хаара	38
2.2.3. Каскадна класифікація	40
2.3. Методи розпізнавання облич	41
2.3.1. Методи, засновані на значеннях яскравості пікселів	42
2.3.2. Методи, засновані на характерних точках	46
2.4. Метод найближчого сусіда	47
Висновки до розділу 2	49
РОЗДІЛ 3 АНАЛІТИЧНА ЧАСТИНА	50
3.1. Python image processing бібліотеки.....	50
3.2. OpenCV	53
3.3. Face Recognition.....	61

Висновки до розділу 3	67
РОЗДІЛ 4 ПРАКТИЧНА ЧАСТИНА	68
4.1. Інструментарій для розробки.....	68
4.2 Аналіз потоків даних.....	69
4.3 Опис інтерфейсу системи	71
4.4 Тестування розробленої системи.....	73
Висновки до розділу 4	75
РОЗДІЛ 5 РОЗРОБКА СТАРТАП ПРОЕКТУ	76
5.1. Резюме бізнес плану.....	76
5.2. Опис ідеї стартап-проекту	76
5.3 Технологічний аудит ідеї проекту	79
5.4. Обґрунтування ресурсів та витрат проекту	80
5.4. Обґрунтування ресурсів та витрат проекту	81
Висновки до розділу 5	83
ЗАГАЛЬНІ ВИСНОВКИ	84
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	85

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

AdaBoost (Adaptive Boosting) – алгоритм підсилення класифікаторів шляхом об'єднання їх в «комітет».

SNoW (Sparse Network of Windows) – алгоритм виявлення облич, у вигляді двошарової мережі.

MNN (Multilayer Neural Networks) - багатошарові нейронні мережі.

ASM (Active Shape Model) – активна модель форми.

XML (Extensible Markup Language) – розширювальна мова розмітки.

UML (Unified Modelling Language) - уніфікована мова для моделювання.

ML – машинне навчання.

DL – глибинне навчання.

KNN - метод k-найближчих сусідів, з англійської: k-nearest neighbors.

ВСТУП

Останнім часом широкого поширення набуває відеоаналітика - технологія, котра використовує методи комп'ютерного зору з метою автоматизованого збору та аналізу різноманітної інформації на основі послідовності зображень, отриманих з відеокамер в реальному часі або із записів відео. Дану технологію можна використовувати у системах безпеки, системах відеоспостереження, торгівлі чи транспорті.

За даними MarketsandMarkets[2], ринок відеоаналітики продовжить стрімко розвиватися в найближчі роки і досягне \$4,873 млн до 2025 року, що підтверджує активний ріст напрямку.

Одним із завдань, яке вирішує відеоаналітика - це розпізнавання обличчя у відеопотоках. Рішення цієї проблеми, перш за все, має пряме застосування в системах ідентифікації особи та контролю доступу.

Однією з причин підвищеної уваги до біометричних технологій є наявність великої кількості соціальних та комерційних інституцій, де можливі шляхи вирішення зазначеної проблеми сприйматимуться досить позитивно. Наприклад, зображення обличчя, отримане з відеокамер зовнішнього спостереження, сьогодні є важливим елементом документальної бази при розслідуванні злочинів, спостереження та розслідування злочинних подій (автоматичне спостереження за людьми, пошуки підозрювальних, активне відеоспостереження під час надзвичайних подій, тощо), а також у банківській сфері (банкомати, системи дистанційного керування рахунками), у системах контролю ідентифікації особистості (паспорти, водійські права, імміграційні картки), інформації про безпеку (доступ до персональних комп'ютерів й застосунків, криптографічні програми, бази даних, медична інформація, електронна комерція і т.д.).

Традиційні системи автентифікації вимагають наявності ключа, запам'ятовування пароля, ідентифікаційної картки чи іншого

ідентифікаційного елемента, який можна забути або ж втратити. Біометричні системи, на відміну від них, спираються на унікальні біометричні характеристики людини, які однозначно ідентифікують конкретну особу та які важко сфальсифікувати. До цих ознак відносяться індивідуальні риси обличчя, зображення на сітківці ока, відбитки пальців і форма долоні.

Незважаючи на обширні дослідження у цій галузі, які проводилися в усьому світі протягом останніх десятиліть, так і не було розроблено надійного методу виявлення та розпізнавання людських облич за будь-яких умов.

Очікується, що розпізнавання обличчя стане найпоширенішою біометричною технологією. Тому що для цього не потрібно спеціального обладнання, не потрібен фізичний контакт з пристроями, не потрібні спеціальні зупинки чи очікування, поки система запрацює. Для певної якості роботи достатньо використовувати веб-камеру. Звичайно, веб-камери буде недостатньо, потрібна програмне забезпечення, яке буде обробляти отримані зображення і приймати рішення на основі заданого способу й режиму роботи. Режим роботи таких систем часто повторюється, але методи зазвичай істотно відрізняються. У даний час відомо і використовується велика кількість методів і їх модифікацій. Поширеними методами та їх похідними є метод Віоли-Джонса, головних компонент, еластичних графів, головні компоненти, методи, принцип яких заснований на методі геометричного розпізнавання, метод локальних бінарних шаблонів, методи на основі нейронних мереж, та ін.

Кожен із способів має свої переваги й недоліки, котрі проявляються в певних ситуаціях, що призводить до непередбачуваних випадків «відмова у доступі для свого чи пропуску чужого». Фактори, що впливають на продуктивність методу, загалом однакові, але кожен метод має власну стійкість до деяких із них. Такими факторами є рівень і кут освітлення, відстань до камери, стан виразу обличчя або здоров'я людини, її настрій, кут

обличчя, кут обличчя по відношенню до камери, вікові зміни, наявність бороди або вусів і т. д. Існуючі методи дозволяють вирішити ці проблеми з неприпустимим затримками в часі.

Актуальність теми. Останнім часом широкого поширення набуває відеоаналітика - технологія, котра використовує методи комп'ютерного зору з метою автоматизованого збору та аналізу різноманітної інформації на основі послідовності зображень, отриманих з відеокамер в реальному часі або із записів відео. Дану технологію можна використовувати у системах відеоспостереження, системах безпеки, транспорті чи торгівлі.

Рішення цієї проблеми, перш за все, має пряме застосування в системах ідентифікації особи та контролю доступу.

Об'єкт дослідження є процес розпізнавання образів.

Предмет дослідження є методи розпізнавання образів для ідентифікації людини у відеозображенні.

Метою роботи є збільшення ефективності ідентифікації особистості у відеозображенні в режимі реального часу, шляхом використання методів k-найближчих сусідів та Віоли-Джонса.

Завдання для вирішення поставленої задачі:

1. Огляд і аналіз існуючих рішень;
2. Дослідження методів та алгоритмів, вимог для вирішення поставленого завдання;
3. Розроблення структури програмного забезпечення;
4. Створення прикладної програми – система розпізнавання облич.

Наукова новизна. Удосконалено й оптимізовано метод розпізнавання облич за допомогою технологій глибокого навчання та комп'ютерного зору в відеопотоці у режимі реального часу, що дає можливість підвищити точність ідентифікації людини і зменшити споживання пам'яті під час розпізнавання.

Практична значимість отриманих результатів. В результаті роботи була розроблена система розпізнавання осіб, яка може використовуватися для контролю доступу та ідентифікації людей.

Структура роботи. Дані робота побудована за класичним типом та викладена на 88 сторінках машинописного тексту. Складається з вступу, 5 розділів, висновків, списку використаних літературних джерел, який містить 43 найменування. У роботі представлено 20 рисунків і 11 таблиць.

РОЗДІЛ 1

ЛІТЕРАТУРНИЙ ОГЛЯД

1.1. Сучасний стан та тенденція розвитку технологій розпізнавання

Для того, щоб сформулювати завдання, необхідно точно зрозуміти визначення поняття - «Система розпізнавання осіб». Будь-яка система розпізнавання обличчя - це технологія, яка має можливість верифікувати або ідентифікувати особу за цифровим зображенням або відеозображенням із джерела відео.

Існує кілька методів, за допомогою яких відтворюються системи розпізнавання обличчя, але загалом вони працюють шляхом порівняння рис обличчя, вибраних із певного зображення, з обличчями в динамічних базах даних. Вони відображаються як програмні технології, засновані на біометричному штучному інтелекті, які можуть з певною ймовірністю виявляти виняткових людей, аналізуючи шаблони на основі фактури та форми обличчя людини.

Хоча й спочатку це була форма комп'ютерних програм, останнім часом вона все частіше використовується на мобільних платформах та інших формах технологій (робототехніка чи застосування штучного інтелекту для самоконтролю). Але найчастіше він використовується з метою контролю доступу у системах безпеки і його можна порівняти з іншими біометричними засобами.

У 1997 році Крістоф фон дер Мальсбург і аспіранти з Університету Бохума в Німеччині та Університету Каліфорнії в Сполучених Штатах розробили інтелектуальну систему, яка перевершила більшість подібних систем в МІТ і Університеті Меріленда. Вона була розроблена за фінансування дослідницької лабораторії збройних сил США.

Програмне забезпечення продавалося як ZN-Face і використовувалося

такими клієнтами, як Deutsche Bank та іншими операторами аеропортів, а також іншими комерційними чи державними підприємствами. Програмне забезпечення було достатньо надійним, щоб ідентифікувати за не ідеальним виглядом обличчя. Його також часто бачать, що долає бар'єри ідентичності, такі як вуса, борода, змінена зачіска чи навіть сонцезахисні окуляри.

Системи були протестовані з використанням зображень обличчя, 3D-сканування високої роздільної здатності та зображень райдужної оболонки ока. Дослідження стверджували, що сучасні системні алгоритми є в 10 разів точніші, ніж старіші системи розпізнавання обличчя 2002 року та є в 100 разів точнішими, ніж у попереднє десятиліття. Деякі алгоритми змогли перевершити людей у ідентифікації людей і однозначно ідентифікувати близнюків або людей зі схожими характеристиками.

Було виділено декілька перешкод для ідентифікації людей, головними з яких є дві конкретні проблеми. Перш за все, слід розуміти, що піксель у певній області зображення нічого не означає в певному сенсі (зміна кольору навряд чи гарантує різницю на всьому зображенні, не кажучи вже про помітність такої зміни для людського ока).

Звичайно, проблема кількості пікантних слів, особливо коли картинка велика, одна на тлі інших, неважлива. Інша складність полягає в тому, що будь-яке зображення являє собою певний масив пікселів. З чого випливає висновок, що конкретна установка зображень марна і неекономічна.

Що стосується ефективності ідентифікації людей, необхідно розробити компактний і зручний спосіб подачі фото. Сьогодні у світі існує кілька добре відомих інструментів для стиснення файлів, пов'язаних із графічними зображеннями, у форматах (bmp, pdf, png, jpg, тощо) із втратою даних, які практично не впливають на цілісність файлів, але проблема використаного формату непрактичний у часі для класифікації людей на фотографіях з таких причин, як вирішення проблеми відокремлення одних людей від інших, що вимагає набагато менше інформації.

Перш за все, це стосується умови, в якій не потрібно визначати, як ця

людина виглядає за моделлю, а необхідно вирішити інше завдання, яке полягає в питанні - яка людина виглядає певним чином. Ще одна перешкода полягає в тому, що ту саму людину можна зафіксувати з різними зовнішніми факторами та деталями, такими як світло в кімнаті чи ззовні, фон, розташування об'єктів, вираз обличчя та навіть шуми, записані камерою.

По суті, складність поставленого завдання полягає в розпізнаванні людей за зображенням обличчя, яке ділиться на кілька компонентів: пошук у масивних базах даних, контроль доступу й безпека фотографій у документах. Використання цих компонентів для захоплення та розпізнавання осіб давно використовується сучасними онлайн-сервісами. Ці компоненти відрізняються як системою розпізнавання, так і вимогами до рішення.

У сучасний час існує незліченна кількість готових інформаційних продуктів, які мають насичену спрямованість, котрі замінюють використання певних рис обличчя. Більшість відомих систем є комерційними продуктами, і інформація про алгоритмічну підтримку цих систем не є загальнодоступною.

Найвідоміші з них - VOCORD FaceControl 3D та DeepFace. Це програмне забезпечення існує як комерційна розробка, і деталі таких інтелектуальних систем (ІС) недоступні у вигляді програмного коду чи офіційної документації. Звідси складність цього проекту, оскільки переважна кількість інформації має або поверхневий характер, або є загальна схематична відсутність роботи алгоритмів чи їх зразків.

Таким чином, в цілому можна відзначити, що розробка таких систем вимагає значних людських і фінансових ресурсів, тому лише великі компанії та наукові підрозділи здатні докорінно змінити рушійні сили в цих напрямках, не забуваючи, що якість такого програмного забезпечення буде переважат у компаніях, які мають ресурси безпеки або державну підтримку, на відміну від місцевих онлайн-сервісів. Але навіть у таких програмах виявлення та розпізнавання в системі VOCORD FaceControl 3D є

ключовими принципами.

Ці ключові елементи засновані на одночасних знімках стереокамерою з різних ракурсів, побудові 3D-моделі обличчя та порівнянні 3D-моделі з основною 3D-моделлю або звичайною фотографією. Зрозумівши, як працюють примітивні алгоритми цих програм, можна поступити наступним чином. Як приклад, можна припустити, що фотозображення мають розміри $100 * 100$ пікселів, з 256 відтінками сірого.

Існує багато варіантів завдань на визначення. Одним із варіантів є один із подібних прикладів нижче. Ми вважаємо, що існує набір зразків, які потрібно навчити (тобто збільшити ймовірність відрізнити конкретне обличчя від не-обличчя чи іншого обличчя). Нехай зразки налічують до 400 фотографій (10 фотографій на 40 осіб у різних умовах). У цьому випадку задачу на ділення має сенс представити так.

Нове зображення доступне у форматі, обумовленому вище. Ми повинні отримати відповідь з кількох наведених нижче варіантів. Наприклад, ми можемо послатися на те, що на цій фотографії немає людського обличчя. Але це не гарантія, що на фото зображене обличчя конкретної людини. Тоді алгоритм може виявити риси обличчя, яких немає в колекції. Робота самого алгоритму досить проста. По-перше, механізм алгоритму відокремлює риси обличчя людини від фону. Потім фотозображення нормалізується. Механізм включає наступні кроки:

- Заміна роздільної здатності зображення на $100*100$ пікселів
- Зміна кольорів до 256 відтінків сірого
- Відображення комплексної насиченості зображення при певних середніх значеннях.

Деякі алгоритми вимагають вертикалізації зображення. Нормалізація відбувається при зміні кута обличчя до потрібного. Далі алгоритм розділяє риси обличчя. Дані істотно залежать від самої функції відокремлення конкретної людини від іншої.

Якщо для ідентифікації особи наявність фотокартки не є

обов'язковою, на завершальному етапі ідентифікації залучається класифікатор, який за наявними даними дає відповідь на вищезазначені запитання.

1.2. Поняття комп'ютерного зору

Комп'ютерний зір (англ. CV, computer vision) — технологія, спрямована на створення комп'ютерних засобів, здатних виявляти, відстежувати й класифікувати об'єкти.

Аналіз, прогнозування та розпізнавання вийшли на новий рівень з застосуванням технологій штучного інтелекту та машинного навчання. В останні роки комп'ютерний зір є надзвичайно перспективним науковим напрямком.

Однією з найпоширеніших проблем комп'ютерного зору, котра зараз активно досліджується, є розпізнавання облич. І останні розробки світових технологічних лідерів це підтверджують. У вересні 2017 року технологія Face ID була представлена на спеціальному заході Apple [9].

Ця технологія розпізнавання активно впроваджується в соціальних мережах для ідентифікації людей на фотографіях. Meta є лідером у цьому напрямку. Алгоритм, розроблений Meta, показує ефективність розпізнавання 93%.

Результатом широкого використання Face ID стало створення різноманітних бібліотек та інтерфейсів прикладного програмування (API) для розпізнавання облич. Розроблені рішення поділяються на спеціальні для конкретної області або універсальні, написані під певну мову програмування або підтримують всі популярні мови.

Доволі часто, серед усього різноманіття засобів, важко зрозуміти, яке з них найкраще підійде для вирішення тієї чи іншої проблеми.

Робота присвячена проблемам комп'ютерного зору, а саме розпізнаванню облич на основі бібліотек OpenCV [24] і dlib [5]. Під час

роботи будуть оцінені переваги та недоліки, а також доцільність використання цих бібліотек у проектах побудови систем розпізнавання.

1.3. Програмні системи розпізнавання обличчя

Так, у США після сумнозвісних подій 11 вересня, аеропорти активно впроваджували та тестували систему розпізнавання людей. У Сполучених Штатах ця технологія використовується на стадіонах під час великих спортивних подій, таких як Кубок США з футболу.

У Великобританії, в передмісті Лондона, працює система розпізнавання облич з 250 підключеними телекамерами. Система автоматично фіксує зображення облич людей, які йдуть вулицею, та порівнює ці зображення з набором фотографій розшукуваних злочинців.

Американська поліція, наприклад, у місті Вірджинія-Біч, штат Вірджинія, використовує для подібних цілей систему розпізнавання осіб.

Перші результати випробування системи розпізнавання для ідентифікації людей, тимчасово затриманих у шотландському місті Грампіанс, продемонстрували цінність цієї технології.

Система розпізнавання обличчя – це біометрична система ідентифікації людини за зображенням її обличчя, яка складається з двох модулів – захоплення та розпізнавання. Загалом система працює так.

Телекамера приймає зображення. Кожне зображення потрапляє в модуль захоплення, де зображення перевіряється на наявність однієї або кількох людей. При виявленні особин, визначають розмір і розташування кожної особини. Захоплення відбулося (рис. 1.1).

Після цього розпізнавач починає працювати. По-перше, він визначає положення центрів людських очей на зображенні.

Далі з обличчям відбувається те ж саме, що і з головою героя будь-якого мультфільму - особа повертається строго в фас, а його розмір зменшується або збільшується так, що центри очей лежать в двох певних

точках. Відбувається «нормалізація» образу людини.

Наступним кроком є робота над яскравістю, контрастністю та іншими налаштуваннями зображення, що дуже схоже на те, що ви можете зробити із зображенням за допомогою сучасної програми для проектування.



Рисунок 1.1. Захоплення обличчя у програмному додатку

Після цього, розпізнавач переходить до вимірювання певних параметрів людини – тих, за якими можна відрізнити одну людину від іншої. Ці параметри були відібрані та включені в алгоритм розпізнавання розробниками модуля. Звісно, зворотна процедура – отримання зображення людини за її параметрами – вже неможлива.

Набір параметрів, отриманих із зображення обличчя, є біометричним

зображенням обличчя людини. Щоб система «Face-intelligence» могла визначити, кому належить зображення в кадрі, вона порівнює біометричне зображення цієї людини з набором таких самих зображень, про які заздалегідь відомо, кому вони належать. З цих зображень створюються всілякі бази даних - співробітників компанії, розшукуваних людей, важливих людей і т.д.

Результат порівняння виражається у коефіцієнті розпізнавання, який визначає ступінь схожості між захопленою відеокамерою людиною і зображенням у базі. Таким чином, програма вирішує, що це та сама особа, якщо коефіцієнт досить великий. Водночас можна налаштувати поріг, який повинен досягати коефіцієнт, щоб прийняти таке рішення.

Реакція на результат розпізнавання, залежить від застосування системи. Якщо мова йде про спробу контролю над терористами в аеропорту під час перевірки, то система повинна повідомити оператору, що швидкість розпізнавання висока, тобто просто привернути його увагу.

У професійному просторі, де одні й ті ж люди проходять повз один одного кілька разів, взаємодія системи розпізнавання облич та системи контролю доступу, може бути повністю автоматизована: якщо людина схожа на одного зі співробітників, значить прохід дозволяється.

Система також має спеціальний монітор розпізнаних осіб. На ньому, зліва, зображено особу, яку зафіксувала телекамера, разом із інформацією про цю людину. Праворуч, знаходиться список схожих кандидатів у порядку спадання схожості із зазначенням ступеня подібності та особистої інформації про кожного (рис. 1.2).

Є можливість шукати впізнану особу у відеоархіві та переглядати відео, на яких вона присутня. В інтерфейсі, призначеному для пошуку людей за зображенням, вказуються фільтри за періодом пошуку та параметри пошуку.

Після виконання запиту, на екран виводяться ті, що задовольняють умовам пошуку запису у протоколі.



Рисунок 1.2. Вікно додатку ідентифікації осіб

Слід виділити ще одну відмінну рису комплексу «Інтелект» - розподіл системи. Ця перевага характерна для всіх продуктів, розроблених на його основі. У програмі, розподіленість досягається таким чином: телевізійні камери, які отримують вихідне зображення, можуть бути встановлені в різних місцях далеко одна від одної, а захоплення та розпізнавання людей можуть виконуватися на різних серверах, а бази даних, які використовуються для розпізнавання, можуть також бути на одному або кількох серверах. Окрім того, оскільки розпізнавання є ресурсомісткою операцією, для сервера розпізнавання можна призначити окремий комп'ютер, який міститиме лише сам сервер .

Загалом, застосування системи розпізнавання обличчя визначається її характеристиками, які відрізняються від інших біометричних технологій, тобто технологій, які використовують її особливості, унікальні для кожної людини. Надійність розпізнавання обличчя поступається деяким іншим біометричним технологіям, зокрема двом найбільш розробленим і

доведеним до практичного рівня - ідентифікації людини за відбитками пальців і за райдужною оболонкою ока. При цьому головна перевага розпізнавання обличчя полягає в тому, що воно не вимагає жодних дій від керованої особи.

Типовим прикладом використання системи розпізнавання обличчя є її інтеграція із системою контролю доступу в компаніях, організаціях та установах. Особливою перевагою такого тандему є посилення заходів безпеки та контролю. Недоліком системи контролю доступу є можливість маніпуляцій із картками доступу – вони можуть бути передані іншим особам, викрадені, втрачені. Цей дефект безболісно компенсує система розпізнавання обличчя. Програма зчитує ім'я її законного власника на картці та знаходить його фото в базі даних. Потім вона розпізнає обличчя людини, яка стоїть перед телекамерою, та порівнює його з фотографією, витягнутою із бази даних. Такий подвійний контроль гарантує недоторканність карти доступу.

Іншим типовим прикладом є використання розпізнавання обличчя в аеропорту. Тут, крім огляду речей, перевіряють обличчя пасажирів, порівнюючи його зображення з наявними в базі фотографіями. Як було сказано вище, подібна система вже працює в деяких закордонних аеропортах. Крім того, аеропорти обмінюються власними зібраними базами даних, що значно збільшує можливість визначення місцезнаходження розшукуваної особи. Наприклад, під час чергової перевірки громадянина, який вчинив злочин під одним прізвищем, можна виявити під іншим.

Ще одна група об'єктів, де використання системи розпізнавання обличчя є дуже актуальним, це торговельні підприємства та розважальні заклади. Магазини, супермаркети, АЗС, клуби, кафе, ресторани, усіх об'єднує одне - прагнення отримати максимальний прибуток, для чого необхідно підвищити якість обслуговування клієнтів і забезпечити відвідувачів. Їм важливо обмежити доступ людей, які з тих чи інших причин є небажаними в даному закладі, і, навпаки, заздалегідь знати про прихід постійного клієнта

чи VIP-персони, щоб обслужити його на найбільш високому рівні.

Аналогічні системи багатофакторної ідентифікації сьогодні часто широко поширені у сфері виробництва, включаючись у комплекс продуктів аутентифікації, наприклад, у системах безпеки. Ці системи зазвичай вимагають додаткового та більш точного обладнання для відбору проб, наприклад IFACE202 та Multi-Bio 700 від ZKSoftware. Обидві згадані системи є терміналами багатофакторної ідентифікації, які працюють з фотографіями людей, відбитками пальців та іншими атрибутами.

Перевагою таких систем є висока надійність за рахунок спеціально розробленого обладнання. До недоліків відносяться необхідність і розмір додаткового пристрою, обмежене розширення функцій і вартість.

Також є набір програмних продуктів, які не вимагають складного обладнання, а використовують стандартні веб-камери та мікрофони. Однак ці програми зазвичай базуються на одному з факторів. Наприклад, Logitech FastAccess пропонує керування ідентифікаторами та паролями на основі зображення людини. Лабораторія GritTec надає програмний пакет голосової ідентифікації (табл. 1.1).

Таблиця 1.1.

Порівняння аналогів систем ідентифікації за біометричними параметрами

Назва системи	Багатофакторність	Необхідність додаткового обладнання	Оцінка простоти використання (0-5)
IFace202	Зображення, відбитки пальців, карта	Необхідний додатковий термінал	4
Multi-Bio 700	Зображення, відбиток пальців, карта, пароль	Необхідний додатковий термінал	3
FastAccess	зображення	Стандартна веб-камера	5
GraitTec lab	звук	Стандартний мікрофон	2

Висновки до розділу 1

Загальновизнаним фактом є те, що в сучасному постіндустріальному світі інформація стає основним надбанням і її концентрація поступово зростає, тобто людина залучається до обробки все більшої кількості інформації. При цьому неминуче зростає важливість таких аспектів, як безпека інформації, контроль доступу, моніторинг активності, якість яких безпосередньо визначається надійністю аутентифікації.

На основі аналізу існуючих систем, виконано їх порівняння, визначено переваги і недоліки.

Проаналізовано завдання й виявлено основні проблеми, котрі можуть виникнути у ході її вирішення.

РОЗДІЛ 2

ТЕОРЕТИЧНА ЧАСТИНА

2.1. Методи виявлення обличч

Розпізнавання обличчя полягає у створенні певного набору правил, яким має відповідати фрагмент зображення, щоб його розпізнали як обличчя людини, наприклад: обличчя симетричне, риси обличчя (рот, очі, ніс) відрізняються від яскравості зображення шкіри, риси обличчя влаштовані певним чином. Це також включає виявлення на основі шаблонів, визначених розробником. Розпізнавання обличчя за моделлю полягає в перевірці відповідності кожної з ділянок зображення заданій моделі.

Серед різноманіття існуючих алгоритмів для виявлення обличчя можна виділити кілька відповідних методів[34], які заслуговують на увагу. Розглянемо особливості, переваги та недоліки кожного із них.

У 2001 році, Полом Віолою та Майклом Джонсом був запропонований метод Віоли-Джонса[27], який став першим методом, котрий продемонстрував високу продуктивність при обробці зображень у реальному часі.

Основні принципи, на яких базується метод:

- Інтегральне представлення образу. Це дозволяє швидко розрахувати необхідні об'єкти. Інтегральне представлення зображення - це матриця, яка відповідає розмірам вхідного зображення у пікселях. Кожен елемент матриці являє собою суму інтенсивностей усіх пікселів, розташованих ліворуч і вище цього елемента.

- Використовуються знаки, схожі на знаки Хаара[27], за допомогою яких здійснюється пошук шуканого об'єкта (в даному контексті риси обличчя).

- Вибір ознак здійснюється за допомогою алгоритму бустингу. Бустинг (з англ. – покращення, посилення) – це процедура послідовного

представлення композиції алгоритмів машинного навчання, де кожен наступний алгоритм намагається компенсувати недоліки у композиції всіх попередніх алгоритмів.

- Класифікатор – це функція, яка визначає належність досліджуваного об'єкта до певного класу. Класифікатор, котрий отримує символи як вхідні дані, повертає відповідь «істина» або «брехня», що дозволяє виявити.

- У процесі виявлення використовуються каскади функцій. Основою такої ідеї є побудова послідовності класифікаторів, яка називається каскадом: кожен наступний намагається врахувати помилки попереднього.

- Для зображень використовується метод віконного сканування: зображення скануються за допомогою вікна пошуку, після чого до кожної позиції застосовується класифікатор.

Переваги:

1. Алгоритм є найпопулярнішим і поширеним методом визначення обличчя.

2. Висока швидкість розпізнавання обличчя завдяки використанню каскадного класифікатора.

3. Висока точність виявлення з дуже низьким рівнем помилкових спрацьовувань порівняно з точністю виявлення повільніших алгоритмів.

Недоліки:

1. Потрібна велика дослідницька вибірка та багато часу на вивчення.

2. Обмеження положення обличчя під час знаходження.

AdaBoost (Adaptive Boosting) – це алгоритм для посилення класифікаторів шляхом групування їх у «комітет». Вперше його запропонували Йоав Фройнд і Роберт Шапіро в 1999 році[12]. Цей алгоритм, для покращення продуктивності, можна використовувати разом із декількома іншими алгоритмами класифікації. Даний алгоритм є адаптивним у тому сенсі, що кожен наступний комітет класифікаторів базується на об'єктах, неправильно класифікованих попередніми комітетами. Найчастіше він використовується у поєднанні з іншими

алгоритмами класифікації для їх посилення (наприклад, як описано раніше, у методі Віоли-Джонса).

AdaBoost працює наступним чином. Кожна функція застосовується до кожного зображення. Вибираються функції з найменшими помилками.

Спочатку тестовим зображенням надається однакова вага, після кожної неправильної класифікації вага зображень збільшується. Цей процес триває стільки часу, доки не буде досягнута потрібна точність, або певна частота помилок, або певна кількість характеристик.

Після цієї операції кількість ознак значно зменшується, але в багатьох випадках обличчя займає невелику область на фотографіях, і щоб не виконувати всі функції, які не є необхідними для цієї області, існує концепція каскадної класифікації. Тепер ознаки групуються за етапами. Перші етапи містять невелику кількість ознак, з кожним наступним їх кількість буде збільшуватися. Зона, яка пройшла всі етапи, - це обличчя.

Переваги:

1. Алгоритм адаптовано до проблемних елементів навчальної вибірки. Метод теоретично забезпечує нульову помилку навчання за кінцеву кількість ітерацій.

2. Висока швидкість роботи.

3. Простота реалізації.

Недоліки:

1. Алгоритм чутливий до шуму та викидів.

2. Навчання займає тривалий час, який залежить від кількості класифікаторів та розміру навчальної вибірки.

SNoW (Sparse Network of Winnows) [4] – це алгоритм виявлення облич, який являє собою двошарову мережу, вхідний рівень якої складається з вузлів, кожен з яких відповідає певній ознаці вхідного зображення, а вихід, у свою чергу, складається лише з двох вузлів, кожен з яких відповідає класам розпізнавання зображення (генерується 1, якщо ознака присутня, і 0, якщо вона відсутня на зображенні), вихід складається

лише з двох вузлів, кожен з яких відповідає класу розпізнаного зображення («обличчя», «не обличчя»)[4]. Характеристиками цього алгоритму є функції SMQT (Successive Mean Quantization Transform), прапори рівності певних значень середнього значення та дисперсія в кожному з прямокутних фрагментів зображення розміром 1x1, 2x2, 4x4 і 10x10 (всі зображення мають розмір 20x20). розмір пікселів). Це дає простий набір ознак розмірністю 135424. Під час класифікації вхідні вузли отримують інформацію про наявність певних ознак в обробленому зображенні. Вузли вихідного рівня обчислюють лінійну комбінацію сигналів, створені вузлами входу. Коефіцієнти комбінацій лінійного значення, задаються вагами серед вхідними вузлами. Ці перетворення дозволяють виділити з локальної зони зображення компонент, який не залежить від освітленості. Він полягає в кількісному визначенні площі зображення з порогом кількісного визначення, що дорівнює середньому значенню пікселів, що входять до цієї області [18].

SNoW спеціально розроблений для випадків класифікації, коли потенційна кількість ознак об'єкта, важливих для класифікації, може бути дуже великою, але заздалегідь невідомою. Економна архітектура мережі, дозволяє використовувати велику кількість властивостей зображення як вхідні дані, оскільки всі несуттєві ознаки відкидаються в процесі навчання, що в кінцевому підсумку не сповільнює роботу класифікатора[40].

Рішення про присутність обличчя на зображенні, приймається при умові перевищенні заданого порогу.

Переваги:

1. Використовувані ознаки нечутливі до змін освітленості локальних ділянок зображення.
2. Висока швидкість завдяки екрануванню характерної векторної компоненти.
3. Висока точність виявлення.

Недоліки:

1. Алгоритм чутливий до шуму та викидів.

Методи нейронних мереж включають цілий клас алгоритмів. Основна ідея - це послідовність перетворень сигналу функціональними елементами - нейронами, що працюють паралельно. Суть процесу навчання таких мереж зводиться до зменшення середньої квадратичної помилки. Системи виявлення об'єктів зображення на основі нейронних мереж використовують ієрархічну структуру. Спочатку вектор ознак обробляється грубою мережею із високим рівнем помилок другого порядку, потім, якщо вектор не був класифікований як безособовий, рішення коригується більш повільнішою та точною мережею[11].

Переваги:

1. Висока точність виявлення з правильним налаштуванням параметрів мережі.

Недоліки:

1. Чутливість до шуму.
2. Необхідність ретельного налаштування параметрів нейронної мережі для отримання хороших результатів.
3. Схильність до повторного навчання.
4. Висока обчислювальна складність і, отже, недостатня швидкість роботи для обробки у режимі реального часу.
5. Процедура навчання доволі тривала.

Багатошарові нейронні мережі (MNN). Архітектура багатошарової нейронної мережі складається з послідовно з'єднаних шарів, де нейрон кожного шару з'єднаний своїми входами з усіма нейронами попереднього шару та з виходами наступного шару. Нейронна мережа з двома рівнями прийняття рішень може апроксимувати будь-яку багатовимірну функцію з будь-якою точністю.

Нейронна мережа (НМ) з рівнем прийняття рішень здатна формувати лінійні поверхні поділу, що суттєво скорочує область завдань, котрі вони вирішують, така мережа не зможе вирішувати задачі типу «виключний або».

Нейронна мережа з нелінійною активаційною функцією і двома рівнями зв'язків розширення дозволяє досліджувати всі опуклі області у просторі рішень, а із трьома рівнями рішень - області будь-якої складності, в тому числі невивуклі. При цьому багатошарова нейронна мережа не втрачає здатності до узагальнення. БНМ навчається за допомогою алгоритму зворотного поширення, який є методом градієнтного спуску у ваговому просторі, щоб мінімізувати загальну помилку мережі. Помилки (точніше, значення корекції ваги) поширюються назад від виходів до входів через вагові коефіцієнти, що з'єднують нейрони. Найпростішим застосуванням одношарової НМ (відомої як самоасоціативна пам'ять) є навчання мережі рендерингу зображень, які подаються. Надаючи тестове зображення як вхідні дані та обчислюючи якість реконструйованого зображення, можна оцінити, наскільки добре мережа розпізнала вхідне зображення. Позитивні властивості цього методу у тому, що дана мережа може відновлювати спотворені та зашумлені зображення, проте для більш серйозніших цілей він не є корисним. БНМ також використовується і для прямої класифікації зображень - вхідними даними є або саме зображення в будь-якій формі, або набір із 25 попередньо вибраних ключових характеристик із зображення, виходом є нейрон із максимальною активністю, що вказує на приналежність до розпізнаного класу.

Максимальна активність нейрона вказує на приналежність до визнаного класу. Якщо ця активність нижче певного порогу, вважається, що надане зображення не належить до одного з відомих класів. У процесі навчання вхідні зображення, що належать до зазначеного класу, зіставляються. Це називається «навчання з учителем». У застосуванні до розпізнавання обличчя людини цей підхід добре підходить для завдання контролю доступу до невеликої групи облич. Він забезпечує пряме порівняння з самою мережею, але зі збільшенням класів час навчання та запуску мережі зростає експоненціально, тому для таких завдань, як пошук людини у великій базі даних, необхідно визначити компактний набір

основних функцій для зосередитися на.

Часто БНМ можна використовувати з метою класифікації обличчя на основі певних характеристик, таких як відстань між частинами обличчя (ніс, рот, очі). Використовуються також гібридні системи, наприклад у поєднанні з моделлю Маркова. У класичному БНМ міжшарові нейронні асоціації повністю пов'язані, і зображення представлено як одновимірний вектор, навіть якщо воно є двовимірним.

Архітектура згорткової нейронної мережі (CNN) спрямована на подолання цих недоліків. Він використовує локальні рецептивні поля (забезпечує локальне двовимірне з'єднання нейронів), загальні ваги (забезпечує виявлення певних особливостей у будь-якій частині зображення) та ієрархічну організацію з просторовою підвибіркою.

Згорткові нейронні мережі забезпечують частковий опір масштабуванню, трансляції, обертанню та спотворенню. Архітектура ШНМ складається з кількох рівнів, кожен з яких має кілька площин, і нейрони наступного рівня з'єднані лише з невеликою кількістю нейронів попереднього шару поблизу локальної області. Ваги в кожній точно одній площині 26 однакові (згорткові шари). Після згорткового шару, слідує шар, котрий шляхом локального усереднення зменшує його розмірність. Після цього, знову згортковий шар й так далі. Таким підходом досягається ієрархічна організація. Останні ж шари висвітлюють більш загальні властивості, які менше залежать від спотворень зображення. ЗНМ формуються методом зворотного поширення стандартної помилки.

Порівняно з БНМ, ЗНМ має значні переваги щодо швидкості та надійності класифікації. Корисна властивість ШНМ полягає в тому, що функції, згенеровані з виходів із вищих рівнів ієрархії, можна використовувати для класифікації найближчих сусідів, і ШНМ можуть успішно витягувати ці ознаки навіть для зображень, яких немає в наборі для навчання.

Нейронна мережа Хопфілда. Алгоритм навчання мережі Хопфілда

суттєво різняться від класичних алгоритмів навчання персептронів. У цьому алгоритмі замість послідовного наближення до необхідного стану з обчисленнями помилок, за один цикл всі коефіцієнти вагової матриці обчислюються за формулою, що дозволяє мережі відразу розпочати роботу після цих дій. НМ Хопфілда одношаровий і повнозв'язаний (зв'язків між нейронами немає), його виходи з'єднані з входами.

На відміну від БНМ, НМ Хопфілда є релаксаційним, тобто після розміщення в початковому положенні він працює, поки не досягне стабільного стану, який є його початковим значенням.

Обмеження методу:

- пам'ятні зображення не повинні бути дуже схожі один на одного;
- зображення не повинно бути зміщено або повернуто з початкового стану. Щоб подолати ці недоліки, розглядаються різні варіанти модифікацій класичної мережі Хопфілда. Дана мережа із ортогональним перетворенням дозволяє реконструювати висококорельовані зображення шляхом перетворення їх вихідного набору в подвійний набір векторів. Таким чином, в результаті виходить нейронна мережа, яка може зберігати кілька векторів, і при введенні будь-якого вектора можна визначити, який із збережених є найбільш схожим.

Прихована марковська модель — це статистична модель, яка імітує роботу марковського процесу з невідомими параметрами. Відповідно до моделі, завдання пошуку невідомих параметрів визначається на основі інших контрольованих параметрів. Отримані параметри можна використовувати і в інших подальших аналізах розпізнавання обличчя.

З точки зору розпізнавання, зображення є двовимірним дискретним сигналом. Важливу роль у побудові моделі зображення відіграє вектор спостереження. Щоб уникнути розбіжностей в описах, для розпізнавання зазвичай використовують прямокутне вікно. Щоб не втратити області даних, прямокутні вікна повинні перекриватися. Значення перекриття, а також області розпізнавання вибираються експериментально. Після

видалення блоку він перетворюється на цифровий блок за допомогою одного з двох методів[38]:

- Карунен-Лоева (КЛП);
- Дискретне косинусне перетворення (ДКП).

Support vector machine[6] – це машинний алгоритм опорних векторів, котрий навчається на прикладах і використовується для класифікації об'єктів.

Основою методу опорного вектора є математична сутність – алгоритм максимізації математичної функції щодо доступного набору даних.

Метод використовується для зменшення розмірності простору ознак, не приводячи до суттєвої втрати інформативності набору об'єктів навчання. Застосування методу головних компонент до множини векторів лінійного простору дозволяє перейти до такого базисного простору, в якому головна дисперсія множини буде спрямована вздовж перших осей базису, званих головними осями. Отриманий розтягнутий підпростір головної осі є оптимальним серед усіх просторів, оскільки він найкраще описує набір для навчання. Це набір алгоритмів, подібних до алгоритмів «навчання», які використовуються для завдань класифікації та регресійного аналізу. Цей метод відноситься до класифікаторного сімейства.

Даний метод опорних векторів заснований у першу чергу на тому, що відбувається пошук лінійного поділу класів. Метою навчання більшості класифікаторів є мінімізація помилки класифікації на навчальному наборі (так званий емпіричний ризик). На відміну від них, метод опорних векторів можна використовувати для побудови класифікатора, який намагається мінімізувати верхню оцінку очікуваної похибки класифікації (у тому числі для невідомих об'єктів, які не входять до навчальної множини).

Застосування методу опорного вектора до проблеми виявлення обличчя полягає у пошуку гіперплощин у просторі ознак, які відокремлюють клас зображень обличчя від зображень без обличчя. Ймовірність лінійного поділу складних класів, таких як зображення обличчя і

«не-облич» є малоюмовірною. Класифікація з використанням допоміжних векторів дозволяє механізму функцій ядра неявно проектувати вектори ознак у потенційно більший розмірний простір (навіть вище простору зображення), у якому класи можуть бути лінійно розподілені. Неявне проектування з використанням функцій ядра не призводить до обчислювальної складності, що дає можливість успішно використовувати лінійний класифікатор для лінійно нероздільних класів [16] [14].

Через систему розпізнавання обличчя можуть виникнути два типи помилок: відсутність розпізнавання обличчя та хибне розпізнавання (розпізнавання об'єкта, який не є обличчям). Через наявність двох типів помилок два основних параметри характеризують продуктивність алгоритмів розпізнавання облич: швидкість виявлення, яка вказує на відсоток виявлених облич, і швидкість помилкового виявлення, яка дорівнює загальній кількості помилкових виявлень. аналізу.

Суть методу полягає у знаходженні гіперплощини, яка розділяє два класи. Паралельно серед усіх можливих гіперплощин, що розділяють два класи, необхідно вибрати таку гіперплощину, відстань якої до кожного класу максимальна. Ця гіперплощина називається гіперплощиною максимального розділення, а відповідний лінійний класифікатор – класифікатором максимального розділення[6].

Переваги:

1. Дуже висока стійкість до переробки.
2. Здатність зменшити чутливість до шуму за рахунок точності. Однак пошук оптимального співвідношення цих параметрів вимагає тонкого налаштування.
3. Швидкість роботи є доволі високою у порівнянні із нейронними мережами.

Недоліки:

1. За точністю поступається AdaBoost та SNoW методам.

Одне і те ж саме обличчя, вибране різними алгоритмами, має різний

розмір. Таким чином, алгоритм на основі підсилення повністю висвітлює обличчя, захоплюючи лоб, підборіддя та щоки. Алгоритми «SNoW» і «Опорних векторів» виділяють лише ніс, рот та очі. Ці відмінності зумовлені тим, що в процесі створення алгоритмів побудови класифікатора використовувалися різні набори навчальних зображень.

Порівняння точності висновків і величини похибки другого роду при використанні описаних методів наведено в табл. 2.1 [20].

Таблиця 2.1

Порівняння ефективності методів для виявлення облич.

Метод	Відсоток правильних виявлень	Відсоток помилки другого роду
Нейронні мережі	~91%	~1.27%
Метод опорних векторів	~71%	~0.58%
SNoW	~93%	~0.13%
AdaBoost	~94%	~0.00002%

Як показують представлені дані, алгоритм AdaBoost є найкращим за відсотком правильних виявлень і помилок другого типу. Тому при створенні системи розпізнавання людей у відеопотоках ми будемо використовувати метод Віюли-Джонса, заснований на цьому алгоритмі. Окрім переваг щодо точності виявлення, цей метод має високу швидкість роботи, що робить його найбільш придатним для виявлення людей у реальному часі.

2.2. Виявлення облич методом Віюли-Джонса

Як згадувалося раніше, цей метод був розроблений і представлений Майклом Джонсом та Полом Віюлою у 2001 році[25]. Однак сьогодні цей метод є одним з основних методів для пошуку об'єктів на зображенні у реальному часі.

Можна виділити основні принципи, на яких і базується робота даного методу:

- Інтегральне представлення зображення.
- Пошук обличчя за допомогою ознак Хаара.
- Каскадна класифікація із застосуванням бустингу.

2.2.1. Інтегральне представлення

Для розрахунку яскравості прямокутної ділянки зображення використовується інтегральне представлення [43]. Це представлення часто використовується у багатьох розроблених алгоритмах для комп'ютерного зору. Дане представлення дозволяє швидко обчислити загальну яскравість довільного прямокутника на заданому кадрі, причому час обчислення не залежить від площі прямокутника.

Інтегральним представленням зображення є матриця, розміри якої збігаються із розмірами вхідного зображення. Кожний елемент такої матриці зберігає суму інтенсивностей пікселів, які розташовані ліворуч і вище цього елемента. Елементи матриці розраховуються за такою формулою:

$$I(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y')$$

де $I(x, y)$ – це значення точки (x, y) в інтегральному зображенні; $i(x', y')$ – це значення інтенсивності вхідного зображення.

Використання даного представлення зображення надає можливість одночасно, з різними геометричними параметрами, обчислювати ознаки одного типу, але, оскільки розрахунок матриці інтегрального представлення займає лінійний час, пропорційний кількості пікселів на фотографії.

2.2.2. Ознаки Хаара

«Використання функцій виявлення об'єктів на основі ознак Хаара

було запропоновано Папагоргиу в 1998 році. Віола та Джонс адаптували цю ідею у своїй роботі і отримали прямокутні ознаки, котрі називаються ознаками Хаара. Зовнішній вид яких зображено на рис. 2.1.» [41].

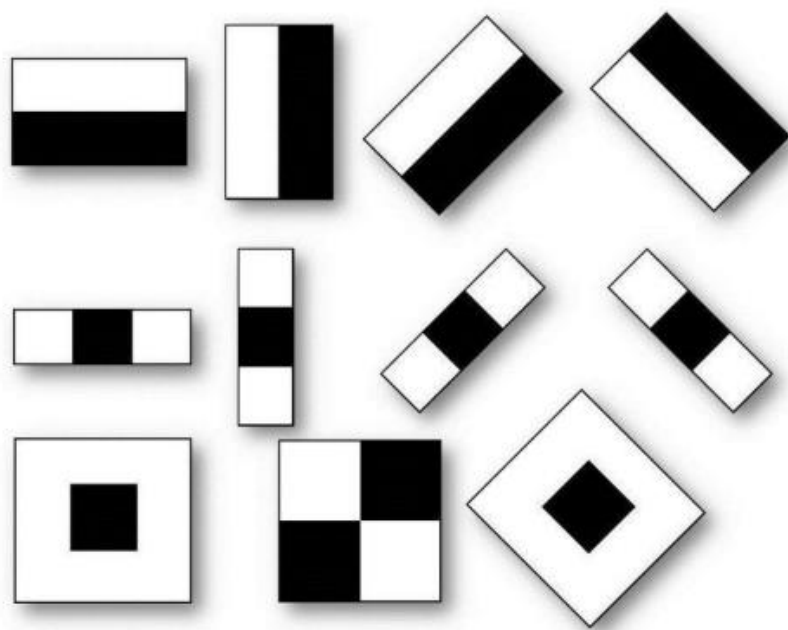


Рисунок 2.1. Ознаки Хаара

Розширений метод Віоли-Джонса, представлений в бібліотеці комп'ютерного зору OpenCV, і використовуваний у розробленій системі, також використовує додаткові ознаки, представлені на (рис. 2.2).

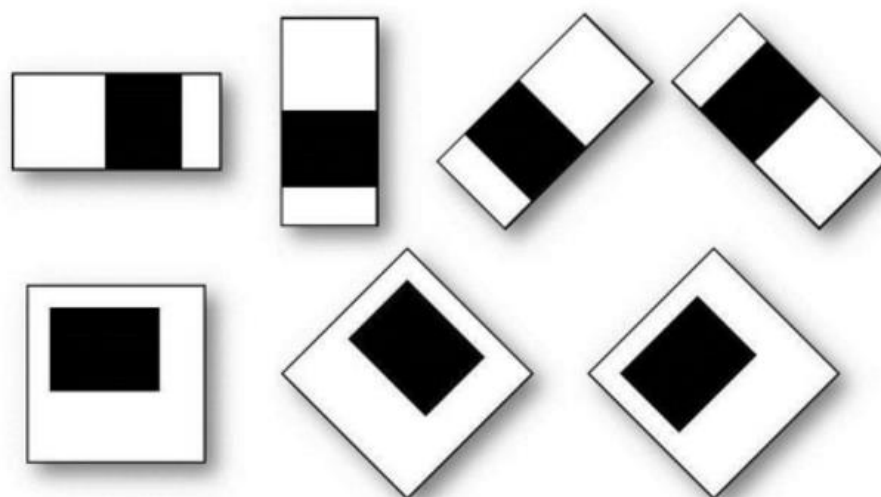


Рисунок 2.2. Додатково розширені ознаки Хаара

«Результат розрахунку таких ознак в інтегральному представленні виглядатиме наступним чином:

$$F = U - V$$

де U – сума яскравості точок, закритих світлою частиною ознаки, а V – сума значень яскравості точок, покритих темною частиною ознаки. Дані ознаки описують зміну яскравості по двом осям зображення.

Пошук облич здійснюється за допомогою так званого вікна сканування, розмір котрого в оригінальному алгоритмі становить 24×24 пікселя. Вікно переміщується по зображенню з кроком у 1 піксель, і для кожної його позиції вираховують ознаки Хаара з різним масштабом і положенням у вікні.

Саме сканування сканування відбувається таким же чином, не зважаючи на різні масштаби скануючого вікна. Знайдені ознаки Хаара передаються до класифікатора, який використовує їхні значення, щоб визначити, чи є область зображення, пов'язана з вікном, обличчям чи ні.» [41].

2.2.3. Каскадна класифікація

Каскадна структура класифікатора прискорює розпізнавання облич, зосереджуючи роботу на найцікавіших ділянках зображення. Каскад забезпечує структурну організацію слабких класифікаторів, навчених процедурою посилення. Таким чином, при низьких обчислювальних витратах можна з високою ймовірністю на ранніх етапах розпізнавання відхиляти зображення, які не містять потрібного об'єкта (тут – обличчя). Приклад каскадної структури класифікаторів наведено на (рис. 2.3).

Кожен рівень каскаду навчається за допомогою згаданого раніше алгоритму AdaBoost [29]. Кількість використовуваних у ньому ознак необхідно збільшувати до того моменту, доки виявлення цільового об'єкта та помилки першого типу не досягнуть заданого значення.

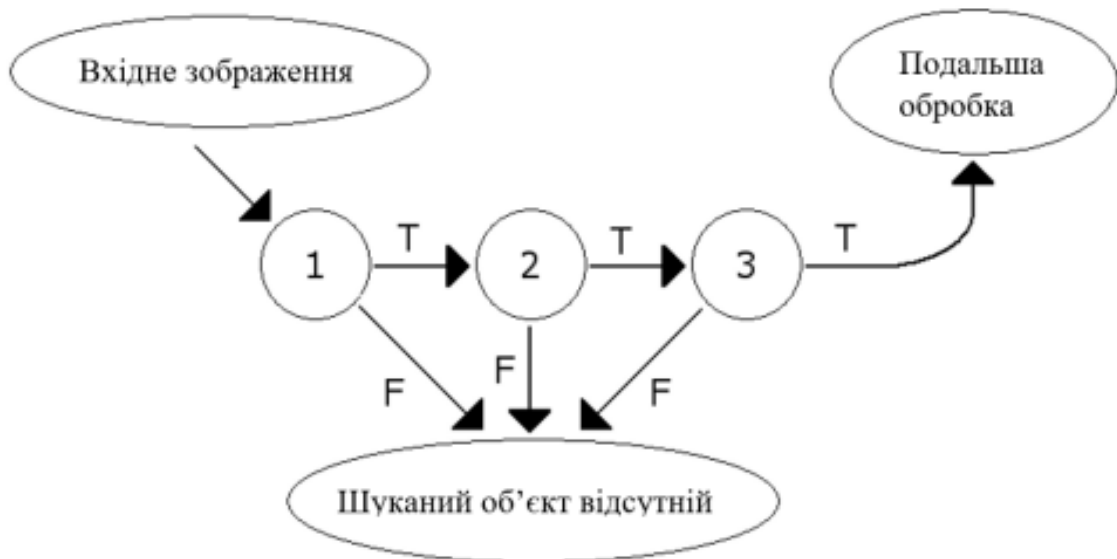


Рисунок 2.3. Каскадний класифікатор

Рівні визначаються шляхом тестування детектора струму на тестовому наборі. Якщо загальна помилка першого типу для всього об'єкта ще не досягнута, до каскаду додається додатковий рівень.

Негативний набір для формування наступних шарів визначається шляхом збору всіх помилкових виявлень при використанні поточного каскаду.

Після класифікації буде отримано набір областей зображення, що містять об'єкт пошуку. Потім усуваються вкладені повтори виявлення того самого об'єкта, викликані масштабуванням скануючого вікна. Для подальшої обробки, обличчя перетворюються у градації сірого та масштабуються до розміру 128 на 128 пікселів.

2.3. Методи розпізнавання облич

Методи розпізнавання обличчя можна розділити на дві підгрупи. До першої підгрупи відносяться методи на основі піксельних значень і методи на основі ознакових точок [32]. Розглянемо основні методи кожної з цих підгруп.

2.3.1. Методи, засновані на значеннях яскравості пікселів

Назва цієї групи методів для розпізнавання виявлених облич, означає, що використовується лише колір або яскравість пікселів зображення. Найпростішим методом є порівняння, при якому ступенем подібності є відстань між векторами яскравості пікселів зображення.

Однак цей метод абсолютно не стійкий до зміни освітлення, положення обличчя і масштабу. Крім того, цей підхід має високу обчислювальну складність і абсолютно непридатний для розпізнавання в реальному часі. Тому часто використовуються методи, що використовують перехід векторного опису зображень у просторі менших розмірностей, у яких порівняння відбувається значно ефективніше[32].

Eigenfaces — алгоритм, запропонований у 1991 році Метью Турком і Алексом Пентландом [19], який отримав широку популярність як перший успішний метод розпізнавання обличчя. Основна ідея алгоритму полягає в тому, щоб за допомогою методу головних компонент знаходити вектори, які найкраще описують образи людей. Використання цього методу, надає можливість виявляти різні зміни у навчальній вибірці зображень обличчя та описати цю зміну в термінах кількох ортогональних векторів, які називаються власними гранями. Обличчя мають корисну особливість: зображення, яке співпадає з кожним вектором, має форму обличчя. Розрахунок головних компонентів зводиться до обчислення власних значень та власних векторів коваріаційної матриці, яка обчислюється по зображенню.

Набір власних векторів, отриманий один раз із навчальної вибірки зображень обличчя, використовується для кодування всіх інших зображень обличчя, які представлені зваженою комбінацією власних векторів. Використання обмеженої кількості власних векторів, надає можливість отримати коротке наближення вхідного зображення людини, яке потім може зберігатися в базі даних як вектор коефіцієнтів, який одночасно служить ключем для пошуку людей у базі даних [19]. Сума головних

компонентів, помножених на відповідні власні вектори, є реконструкцією зображення.

Для кожного зображення обличчя розраховуються його основні компоненти. Зазвичай це від 5 до 200 основних компонентів. Процес розпізнавання заснований на порівнянні головних компонентів всіх відомих зображень з головними компонентами невідомого зображення.

При цьому передбачається, що зображення обличчя, які відповідають людині, групуються у кластери в власному просторі. Зображення - кандидати вибираються із бази даних, з найменшою відстанню від вхідного зображення.

Для застосування метод чистих обличч вимагає ідеалізації таких умов, як рівномірні параметри освітлення, нейтральний вираз обличчя, відсутність перешкод у вигляді окулярів і бороди. Якщо ці умови не виконуються, головні компоненти не відображатимуть відмінностей між класами. Наприклад, при різних умовах освітлення метод власних граней практично не застосовний, оскільки перші головні компоненти в основному відображають зміни освітлення, а результатом порівняння є зображення, яке має рівень d подібного освітлення. В ідеальних умовах точність розпізнавання методу досягає понад 90%.

Обчислення набору власних векторів займає багато часу. Одним із способів є розташування зображень у смуги та стовпчики. у цій формі представлення розмір зображення на порядок менший, обчислення та розпізнавання відбуваються швидше, але відновити вихідне зображення неможливо.

Метод головних компонентів зарекомендував себе на практиці. Однак у випадках, коли відбуваються значні зміни в освітленні або виразі обличчя людини, ефективність методу значно знижується.

Ідея методу полягає в тому, щоб представити зображення обличчя у вигляді набору векторів основних компонентів зображення, які називаються

Переваги:

1. Простота реалізації,
2. Можливість розпізнавання в реальному часі
3. Можливість компактного зберігання великих обсягів даних. [32]
4. Швидке розпізнавання

Недоліки:

1. Обчислення набору власних векторів займає багато часу.
2. Жодних перешкод, таких як окуляри чи бороди.
3. Необхідно виконати якісну попередню обробку, привівши зображення до стандартних умов.
4. Зображення повинні бути отримані в умовах близького освітлення, однакової перспективи.
5. Навчання відбувається один раз, при додаванні нових граней в базу вам потрібно буде перебудувати перший з ваших векторів.

Недоліки полягають у тому, що більші власні вектори описують особливості освітлення більше, ніж риси обличчя, оскільки метод головних компонентів спочатку вибирає підпростір для апроксимації даних, а не даних класифікації [32].

Fisherfaces — це алгоритм, який, на відміну від методу eigenfaces, використовує для аналізу лінійний дискримінант Фішера. Процес даного алгоритму заснований на знаходженні проєкції даних, в якій класи зображення обличчя максимально розділені. При використанні методу головної компоненти розподіл даних у базі даних облич є максимальним. Ця відмінність вирішує проблему високої чутливості до змін освітлення [25].

Метод головних компонентів, який є фундаментом методу Eigenfaces, він максимізує загальну дисперсію даних, за допомогою знаходження лінійної комбінації ознак. Це доволі потужний підхід для представлення даних, проте він не бере до уваги класи об'єктів, і багато цінної інформації можна втратити, видаливши компоненти, які вийшли з ладу в базі даних.

Наприклад, коли база даних містить фотографію з різним освітленням і, як відомо, компоненти, визначені методом головних компонентів, не

завжди містять всю дискримінантну інформацію, вибірки різних класів змішуються, і класифікація стає неможливою. Це завдання вирішується за допомогою алгоритму дискримінантного лінійного аналізу, котрий є основою для методу Fisherfaces.

Дискримінантний лінійний аналіз — це статистичний метод машинного навчання, який використовується для пошуку лінійних комбінацій ознак, які найкраще відокремлюють два або більше класів об'єктів або подій (тобто об'єкти одного класу слід шукати якомога ближче один до одного в просторі та на максимальній відстані) між класами.

Цей алгоритм працює краще в різних умовах освітлення, ніж EigenFaces, проте навчання має відбуватися також у різних умовах освітлення.

Переваги:

1. Завдяки великій кількості фотографій кожної людини, алгоритм стійкий до змін умов освітлення.

Недоліки:

1. Відсутність інваріантності відносно афінних перетворень.

Локальний двійковий шаблон (LBP - Local Binary Pattern) — простий і ефективний оператор перетворення зображення, вперше запропонований у 1996 році для класифікації текстур[31]. Однак пізніше його використовували для розпізнавання обличчя[30].

Цей оператор бере значення яскравості околиці кожного пікселя на зображенні та використовує функцію для призначення кожному пікселю значення, яке описує його околиці. Потім отримане зображення розбивається на підгрупи, де для кожної з яких відбувається розрахунок гістограм. Гістограми об'єднуються та порівнюються за допомогою методів машинного навчання. Класичний варіант використовує метод найближчого сусіда[30].

Перевагами цього методу є простота реалізації та висока швидкість роботи, яку можна підвищити за рахунок використання різноманітних

модифікацій алгоритму. При цьому алгоритм показує високі результати в розпізнаванні осіб і витримує монотонні зміни освітлення. Усе це робить його оптимальним для розпізнавання обличчя в системах обробки в реальному часі.

2.3.2. Методи, засновані на характерних точках

Ця група методів, на відміну від попередньої, не оцінює яскравість певних пікселів, а використовує їхні координати на зображенні й характерні точки. Такими характерними точками є, наприклад, положення носа, центри очей, рот, лінія брів і т.д. [32]. Цей клас методів включає активні шаблони зовнішнього вигляду та активні шаблони форм.

Активні моделі зовнішнього вигляду (англ. ААМ, Active Appearance Models) — це статистичні моделі зображень, які можна адаптувати до реального зображення за допомогою різних типів деформацій. Цей тип моделі в двовимірній версії був запропонований Тімом Кутсом і Крісом Тейлором у 1998 році [10].

Дана активна модель представляє собою два типи параметрів: пов'язані із формою, і параметри пікселів або текстур зображення (параметри зовнішнього вигляду).

Перед використанням модель необхідно навчити на наборі попередньо позначених зображень. Позначення зображень виконується вручну. Завдяки використанню активної моделі наявність перешкод, таких як вуса, дозволяє системі реагувати в деяких випадках, оскільки відхилення значень в окремих точках може не вплинути на загальну реакцію системи. Спочатку ідентифікують обличчя за допомогою маски. Потім обличчя відстежується за допомогою пози та розміру, знайдених за допомогою активного контуру моделі, і ключові точки витягуються з моделі відповідно.

Активні моделі форми (Active Shape Models, ASM) розглядають статистичні зв'язки відносного розташування антропометричних точок. На кожній фотографії зразка експерт відмічає розташування антропометричних точок. Щоб привести координати всіх зображень до єдиної системи,

проводиться так званий узагальнений прокрустів аналіз, після чого всі точки масштабуються і центруються. Потім для всіх зображень обчислюється середня форма та коваріаційна матриця.

Власні вектори обчислюються з коваріаційної матриці, які потім сортуються в порядку спадання відповідних власних значень. Локалізація моделі ASM на новому зображенні, яке не входить до навчальної вибірки, здійснюється в процесі розв'язання оптимізаційної задачі [33].

Однак слід зазначити, що ці моделі спочатку призначені не для розпізнавання, а скоріше для точної ідентифікації характерних точок зображень обличчя. Їх локалізація дозволить виконати процедуру суміщення граней зразка та, для більш точного виявлення іншими методами, привести їх до системи координат.

Зазвичай для таких задач використовується невелика кількість точок, які дозволяють прискорити продуктивність. Для цілей розпізнавання, навпаки, необхідна велика кількість характерних точок, які зможуть підвищити точність класифікації та знизити швидкість роботи системи [36].

У результаті проведеного аналізу можна відзначити, що оптимальним методом обробки в реальному часі є метод із використанням локальних бінарних моделей. У наступних розділах цей метод буде описано більш детально, а також проведено дослідження ефективності різних модифікацій методу, що дасть змогу визначити найкращу модифікацію для розв'язання поставленої задачі за коефіцієнтом швидкість роботи та точність класифікації.

2.4. Метод найближчого сусіда

Метод найближчого сусіда — це простий алгоритм класифікації, суть якого полягає у визначенні приналежності об'єкта до класу, для якого він є найближчим елементом. Наглядний приклад зображено на (рис. 2.4), зелене коло, згідно з цим алгоритмом, слід класифікувати як червоний трикутник.

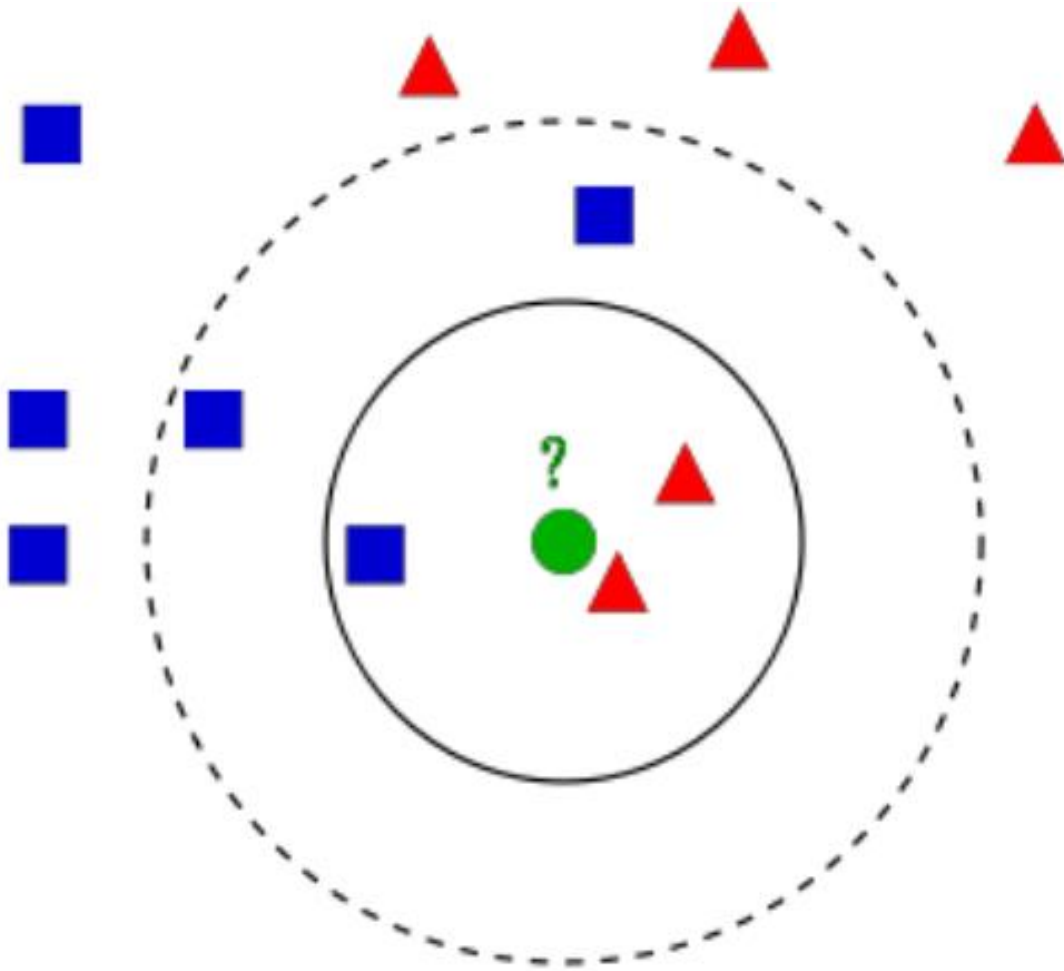


Рисунок 2.4. Метод найближчого сусіда

Щоб покращити результати, ми використовуємо метод, у якому об'єкт призначається до класу, до якого у околиці заданого розміру, належать більшість його сусідів. Однак дослідження, проведені в даній роботі, показали, що при вирішенні задачі класифікації людей такий спосіб негативно впливає на продуктивність класифікатора. Математично цей алгоритм можна описати наступним чином. На першому кроці визначається елемент x_s , N -елементної навчальної вибірки, який є найближчим до представленого зображення x , тобто:

$$\|x - x_s\| = \min\{\|x - x_i\| : i = 1, \dots, N\}$$

На другому ж кроці перевіряється умова приналежності до класу: якщо $x_s \in \omega_i$, то тоді рахується, що $x \in \omega_i$ [37].

Цей метод найкраще підходить, коли вартість неправильної класифікації є високою, а помилки даних низькі. Основним недоліком методу найближчого сусіда є його чутливість до окремих (можливо, помилкових) значень даних. Проте, даний метод показує дуже високу ефективність при застосуванні у широкому діапазоні завдань класифікації [39].

Окремої уваги заслуговує також питання вибору метрики, що визначає відстань між гистограмами. Для отримання максимальної точності класифікації необхідно вибрати метрику, яка б найбільш адекватно відображала відмінності серед гистограмами зображень абсолютно різних класів. У оригінальному дослідженні [30] використовується так звана відстань хі-квадрат, яка розраховується за такою формулою.

$$\sum_{i=1}^n \frac{(x_i - y_i)^2}{(x_i + y_i)}$$

де x_i – і-те значення першої гистограми; y_i – і-е значення другої гистограми.

Висновки до розділу 2

Аналізуючи методи виявлення облич, було проведено порівняння їх переваг і недоліків, та обґрунтовано вибір методу Віоли-Джонса, як алгоритму виявлення облич у розробленій системі.

На основі аналізу методів для розпізнавання облич, було порівняно їх переваги та недоліки, й обґрунтовано вибір методу найближчого сусіда, як алгоритму розпізнавання облич, який повинен бути покладений в основу розробленої системи.

РОЗДІЛ 3

АНАЛІТИЧНА ЧАСТИНА

3.1. Python image processing бібліотеки

Python — це високорівнева мова програмування загального призначення, створена Гвідо ван Россумом у 1991 році. Її філософія дизайну наголошує на зручності читання коду. Вона підтримує кілька парадигм програмування, включаючи об'єктно-орієнтоване, імперативне, функціональне та процедурне, і має велику стандартну та всебічну бібліотеку.

За першим випуском пішли Python 2.0 у 2000 році та Python 3.0 у 2008 році. На момент написання цієї роботи останньою версією є Python 3.10. Python є хорошим вибором для всіх дослідників у науковому співтоваристві, оскільки він [3]:

- безкоштовний і відкритий код,
- мова сценаріїв, що означає, що він інтерпретується,
- сучасна мова (об'єктно-орієнтована, обробка винятків, динамічна типізація тощо),
- стислий, легкий для читання та швидкий для вивчення,
- повний вільно доступних бібліотек, зокрема наукових (лінійна алгебра, інструменти візуалізації, побудова графіків, аналіз зображень, вирішення диференціальних рівнянь, символічні обчислення, статистика тощо),
- корисний у ширшому контексті: наукові обчислення, сценарії, веб-сайти, розбір тексту тощо,
- широко використовується в промислових цілях.

У порівнянні з іншими мовами програмування, такими як C/C++, Java і Fortran, Python є мовою вищого рівня. Тому час обчислення, як правило,

трохи довший, але програмувати набагато простіше. У випадку C і Fortran також доступні обгортки. PHP і Ruby, з іншого боку, також є мовами високого рівня. Ruby можна порівняти з Python, але йому бракує наукових бібліотек. PHP, з іншого боку, є більш веб-орієнтованою мовою. Python також можна порівняти з Mat-lab, яка має дуже велику наукову бібліотеку. Однак він не є відкритим і безкоштовним. Scilab і Octave є середовищами з відкритим кодом, подібними до Matlab. Однак їхні мовні особливості поступаються тим, що доступні в Python.

Загалом люди схильні думати, що складні проблеми вимагають складних процесів для створення складних рішень. Python було розроблено з прямо протилежною філософією. Він має надзвичайно плоску криву навчання та процес розробки для інженерів програмного забезпечення [29]. Він використовується NASA для завдань системного адміністрування як для розробки, так і як мова сценаріїв у кількох своїх системах Industrial Light & Magic використовує Python у виробництві спецефектів для великобюджетних художніх фільмів, Yahoo! використовує для керування своїми дискусійними групами, а Google використав його для реалізації багатьох компонентів свого веб-сканера та пошукової системи [17].

Як уже згадувалося, Python має великий набір бібліотек, які можна імпортувати в проект для виконання певних завдань. У будь-якій науковій роботі, присвяченій математиці, дійсно варто згадати NumPy і SciPy.

NumPy — це бібліотека, яка забезпечує підтримку великих багатовимірних масивів. Оскільки зображення насправді являють собою великі двовимірні (відтінки сірого) або три (кольорові) матриці, ця бібліотека є важливою для всіх завдань обробки зображень. Слід також підкреслити, що багато інших бібліотек (не обмежуючись обробкою зображень) використовують представлення NumPyarray.

SciPy — це бібліотека, створена на основі об'єкта масиву NumPy і містить модулі для обробки сигналів і зображень, лінійної алгебрики, швидкого перетворення Фур'є тощо.

Python Imaging Library (PIL) — це бібліотека, для простих маніпуляцій із зображеннями (обертання, обрізання, зміна розміру, кольору, тощо) та дуже простого аналізу зображень (наприклад, гістограма). За допомогою PIL ми можемо читати зображення з більшості форматів і записувати в найпоширеніші. Найважливішим модулем є Image модуль. Щоб прочитати зображення, необхідно використати всього лише один рядок коду:

```
from PIL import
Image pil_im = Image.open('empire.jpg')
```

Повернене значення, pil_im, є об'єктом зображення PIL.

Перетворення кольорів виконується за допомогою методу convert().

Щоб прочитати зображення та перетворити його на відтінки сірого, необхідно просто додати convert('L'), наприклад:

```
pil_im = Image.open('empire.jpg').convert('L')
```

Найбільш тривіальну програму (яка показує зображення) можна записати наступним чином:

```
from PIL import Image
im = Image.open('/path/to/the/image/')
im.show ()
```

Pillow здатний отримувати багато інформації із зображення та дає змогу виконувати кілька стандартних процедур для обробки зображення, зокрема:

- попиксельні маніпуляції,
- керування маскуванням і прозорістю,
- фільтрація зображення, наприклад розмиття, контурування, згладжування або пошук країв,
- покращення зображення, наприклад підвищення різкості, налаштування яскравості, контрасту або кольору.

Деякі з них будуть продемонстровані. Наприклад, зображення можна легко повернути на певний кут (у нашому випадку 45°), а потім зберегти за

допомогою наступного коду:

```
rotated_image=im.rotate(45)
rotated_image.save('rotated.jpg')
```

Кольорове зображення також можна розбити на різні компоненти (червоний, зелений і синій).

```
r, g, b = im.split()
r.show()
```

Зображення також можна легко збільшити або розмити. У цьому випадку важливо також імпортувати бібліотеку ImageFilter. Потрібний код показано нижче.

```
from PIL import ImageFilter
sharp=im.filter(ImageFilter.SHARPEN)
blur=im.filter(ImageFilter.BLUR)
```

Наприклад, зображення також можна легко скопіювати за допомогою такої команди:

```
cropped_im=im.crop((100,100,400,400))
```

Було продемонстровано, що бібліотека PIL/Pillow дуже проста, якщо потрібна лише основна задача обробки зображень. Для більш детального аналізу та комп'ютерного бачення більше підходять OpenCV.

3.2. OpenCV

OpenCV — це бібліотека C++ із модулями, які охоплюють багато областей комп'ютерного зору. Окрім C++ (і C), зростає підтримка Python як простішої мови сценаріїв через інтерфейс Python на основі коду C++. Інтерфейс Python все ще розробляється, і не всі частини OpenCV доступні, а багато функцій недокументовані. Ймовірно, це зміниться, оскільки за цим інтерфейсом стоїть активна спільнота.

Поточна версія OpenCV (4.6.0) насправді постачається з двома інтерфейсами Python. Старий cv модуль використовує внутрішні типи даних

OpenCV і може бути дещо складним у використанні NumPy. Новий cv2 модуль використовує NumPy масиви і набагато більш інтуїтивно зрозумілий у використанні.

OpenCV містить функції для читання та запису зображень, а також матричних операцій і математичних бібліотек.

У цьому короткому прикладі буде завантажено зображення, надруковано розмір і конвертовано та збережено зображення у форматі .png.

```
import cv
# read image
image = cv2.imread('empire.png') x,y = image.shape[:2]
print x,y
# write image
cv.imwrite('result.png',image)
```

Функція imread() повертає зображення як стандартний NumPy масив і може працювати з широким діапазоном форматів зображень. Ми можемо використовувати цю функцію як альтернативу читанню зображень PIL, якщо необхідно. Функція imwrite() автоматично піклується про будь-яке перетворення на основі закінчення файлу.

У OpenCV зображення не зберігаються за допомогою звичайних каналів кольору RGB, вони зберігаються в порядку BGR (у зворотному порядку). Під час читання зображення за замовчуванням використовується BGR, однак є кілька доступних перетворень. Перетворення кольорового простору виконується за допомогою функції cvtColor(). Наприклад, перетворення в градації сірого виконується так:

```
image = cv2.imread('empire.jpg')
# create a grayscale version
grayscale = cv.cvtColor(image,cv.COLOR_BGR2GRAY)
```

Після вихідного зображення є код перетворення кольорів OpenCV. Деякі з найкорисніших кодів перетворення:

```
cv.COLOR_BGR2GRAY
```

```
cv.COLOR_BGR2GRB
```

```
cv.COLOR_GRAY2BGR
```

У кожному з них кількість кольорових каналів для отриманих зображень відповідатиме коду перетворення (один канал для сірого і три канали для RGB і BGR). Остання версія перетворює зображення у градаціях сірого на BGR і корисна, якщо ви хочете нанести або накласти на зображення кольорові об'єкти. Ми будемо використовувати це в прикладах.

Розглянемо кілька прикладів використання OpenCV для обробки зображень і того, як показати результати за допомогою графіків OpenCV і керування вікнами. Перший приклад читає зображення з файлу та створює інтегральне представлення зображення.

```
import cv2
# read image
image = cv2.imread('fisherman.png')
grayscale = cv.cvtColor(image, cv.COLOR_BGR2GRAY)
# compute image integral
intimage = cv2.integral(gray)
# normalize and write
intimage = (255.0*intimage) / intimage.max()
cv2.imwrite('result.jpg', intimage)
```

Після прочитання зображення та перетворення функції в градації сірого `integral()` створює зображення, де значення кожного пікселя є сумою інтенсивностей зверху та зліва. Це дуже корисний прийом для швидкої оцінки функцій. Інтегральні зображення використовуються в OpenCV CascadeClassifier, який базується на структурі, запропонованій Віолою та Джонсом [28]. Перед збереженням отриманого зображення ми нормалізуємо значення до 0...255 шляхом ділення на найбільше значення. На рис. 3.1 показано результат для прикладу зображення.



Рисунок 3.1. Приклад обчислення інтегрального зображення за допомогою OpenCV функції `integral()`.

У другому прикладі застосовано заливку, починаючи з початкового пікселя.

```
# read image
filename = 'fisherman.jpg'
image = cv2.imread(filename)
h,w = im.shape[:2]
# flood fill example
diff = (6,6,6)
mask = zeros((h+2,w+2),uint8)
cv2.floodFill(image,mask,(11,11), (255,255,0),diff,diff)
# show the result in an OpenCV window area
cv2.imshow('flood fill',image)
cv2.waitKey()
# save the result
cv2.imwrite('result.jpg',im)
```

У цьому прикладі застосовано заливку до зображення та показано

результат у вікні OpenCV. Функція waitKey() виконує роль паузи, доки не буде натиснуто клавішу, і вікно автоматично не закриється. Функція floodFill() бере зображення (відтінки сірого або кольорове), маску з ненульовими пікселями, що вказують на області, які не слід заповнювати, вихідний піксель, нове значення кольору для заміни залитих пікселів разом із нижнім і верхнім пороговими значеннями різниці для прийняття нових пікселів.

Заливка починається з початкового пікселя і продовжує розширюватися до тих пір, поки можна додати нові пікселі в межах порогових значень різниці. Різничні пороги наведені у вигляді кортежів (R,G,B). Результат виглядає так, як на рисунку 3.2.



Рисунок 3.2. Заливка кольорового зображення. Блакитна область позначає всі пікселі, заповнені одним початковим елементом у верхньому лівому куті.

Читання відео з камери дуже добре підтримується в OpenCV. Базовий повний приклад, який фіксує кадри та показує їх у вікні OpenCV, виглядає

так:

```
import cv2
# setup video
capture = cv2.VideoCapture(0)
while True:
    ret,image = capture.read()
    cv2.imshow('test video',image)
    key = cv2.waitKey(10)
    if key == 28:
        break
    if key == ord(' '):
        cv2.imwrite('video_result.jpg',im)
```

Об'єкт захоплення VideoCapture знімає відео з камер або файлів. Тут ми передаємо ціле число під час ініціалізації. Це ідентифікатор відеопристрою, якщо підключено одну камеру, це 0. Метод read() декодує та повертає наступний відеокадр.

Перше значення — це позначка успіху, а друге — фактичний масив зображень. Функція waitKey() очікує натискання клавіші та виходить із програми, якщо натиснуто клавішу «esc» (Аscii номер 27), або зберігає кадр, якщо натиснуто клавішу «пробіл».

Розширити цей приклад можна за допомогою простої обробки, взявши вхідні дані з камери, і показати розмиту (кольорову) версію вхідних даних у вікні OpenCV. Це лише невелика модифікація базового прикладу вище:

```
import cv2
# setup video capture
capture = cv2.VideoCapture(0)
# get frame, apply Gaussian smoothing, show result
while True:
    ret,image = capture.read()
```

```

blur = cv2.GaussianBlur(im,(0,0),6)
cv2.imshow(' blur camera',blur)
if cv2.waitKey(12) == 28:
    break

```

Кожен кадр передається у функцію GaussianBlur() який застосовує до зображення фільтр Гауса. У цьому випадку ми передаємо кольорове зображення, тому кожен колірний канал розмивається окремо.

Функція приймає кортеж для розміру фільтра та стандартне відхилення для функції Гауса (у цьому випадку 5). Якщо розмір фільтра встановлено на нуль, він буде автоматично визначений за стандартним відхиленням.

Читання відео з файлів працює так само, але з викликом VideoCapture() беручи назву відеофайлу як вхідні дані.

```
capture = cv2.VideoCapture('filename')
```

OpenCV дозволяє відносно легко виконувати ще складніші завдання. Існують, наприклад, процедури, які виявляють обличчя (очі на зображенні). Наступна послідовність команд саме така:

```

face_cascade=cv2.CascadeClassifier('C:\\Users\\...\\haarcascade_frontalf
ace_default.xml')
eye_cascade=cv2.CascadeClasifier('C:\\Users\\...\\harcascade_eye.xml')
image=cv2.imread('lena.png')
grayscale=cv.cvtColor(image,cv.COLOR_BGR2GRAY)
faces_list = facecascade.detectMultiScale(gray,1.3,5)
for (x,y,w,h) in faces_list:
    cv.rectangle(image,(y,x),(x+h,y+x),(255,0,0),3)
    ro_gray=gray[y:y+h,x:x+w]
    ro_color=img[y:y+h,x:x+w]
    eyes_list=eyecascade.detectMultiScale(ro_gray)
    for (ex,ey,ew,eh) in eyes_list:
        cv.rectangle(ro_color,(ex,ey),(ey+ew,ex+eh),(255,255,0),3)

```

```
cv2.imshow('image',image)
```

```
cv2.waitKey(28)
```

```
cv2.destroyAllWindows()
```

Результат показано на рис. 3.3. Сам алгоритм застосовує так звані каскадні класифікатори на основі ознак Хаара. Він був запропонований як ефективний метод виявлення об'єктів Полом Віолою та Майклом Джонсом [26]. Кількість цих функцій може бути величезною. Але більшість із них не має значення. Хорошою особливістю є, наприклад, той факт, що область очей зазвичай темніша за область носа та щік.

Друга хороша риса може, наприклад, базуватися на тому факті, що очі зазвичай темніші за перенісся. Зі збільшенням кількості таких функцій ми можемо підвищити надійність алгоритму. Звичайно, помилкова класифікація завжди можлива. Також слід зазначити, що надійність знижується зі зменшенням кількості пікселів у зоні обличчя.



Рисунок 3.3. Приклад розпізнавання обличчя

3.3. Face Recognition

Сферу розпізнавання облич можна визначити як область досліджень, у якій зображення облич групуються в набори, що належать одній людині. Можливо, це легше зрозуміти, якщо ми візьмемо Facebook як приклад.

Раніше Facebook міг розпізнавати обличчя, але тоді користувач повинен був позначити людину, натиснувши на зображення та вказавши її ім'я. Тепер у Facebook є можливість автоматично позначати кожного на зображенні. Це досягається за допомогою алгоритмів розпізнавання облич.

У Python це завдання можна вирішити за допомогою навчених згорткових нейронних мереж і OpenCV. Вся програма значною мірою залежить від різних бібліотек. Таким чином, `paths`, `face_recognition`, `argparse`, `pickle` і `os` потрібно імпортувати в проект Python. Спочатку потрібно зібрати кілька зображень людини, яку ми хочемо розпізнати. Це можна зробити вручну або за допомогою програми Microsoft Bing API для пошуку.

В ідеалі набір даних повинен містити принаймні 30 зображень кожної людини. На зображеннях, які використовуються для навчання, не повинно бути присутніх інших осіб. Два зразки зображень (по одному для кожної людини) показані на рис. 3.4. Архітектура мережі, яка використовується для розпізнавання обличчя, базується на нейронній мережі ResNet-34 [15]. Проте бібліотека розпізнавання облич Python має менше шарів, а кількість фільтрів зменшено вдвічі.

Мережа була навчена на наборі даних із приблизно 3 мільйонів зображень (переважно набір даних VGG [22] і набір даних thescrubs [13]).

Алгоритм складається з чотирьох етапів:

Знаходження всіх облич. На першому `face_recognition` бібліотека розпізнавання облич використовує метод гістограми орієнтованих градієнтів (HOG) [21]. Кольорові зображення потрібно спочатку перетворити на сірі. Потім для кожного пікселя на зображенні ми дивимося, у якому напрямку зображення стає темнішим.



Рисунок 3.4. Зразки зображень Боба Келсо та Джона Доріана

Алгоритм складається з чотирьох етапів:

Знаходження всіх обличчя. На першому `face_recognition` бібліотека розпізнавання обличчя використовує метод гістограми орієнтованих градієнтів (HOG) [21]. Кольорові зображення потрібно спочатку перетворити на сірі. Потім для кожного пікселя на зображенні ми дивимося, у якому напрямку зображення стає темнішим.

Таким чином, ми отримуємо матрицю градієнтів (див. рис. 3.5). Ця матриця значною мірою не залежить від коливань яскравості вихідного зображення. Однак вона занадто велика для маніпуляцій. Таким чином формуються підматриці розміром 16×16 . Тоді знаходиться переважний напрямок для кожної підматриці.

Позування та проектування обличчя. Цей крок вирішує проблему, пов'язану з тим, що обличчя на зображенні може дивитися в іншому напрямку, а не прямо в камеру. Є кілька рішень цієї проблеми. Бібліотека Python використовує підхід із 68 орієнтирами, які присутні на будь-якому обличчі [35].



Рисунок 3.5. Зразок вихідного зображення та його гістограма градієнтів

Потім алгоритм машинного навчання навчається знаходити ці 68 орієнтирів на будь-якому обличчі. Після цього обличчя трансформується за допомогою тонких трансформацій, щоб очі та рот були якнайкраще відцентровані.

Кодування облич. Глибока згорткова нейронна мережа навчена генерувати 128 вимірювань для кожного обличчя. У навчальному процесі одночасно використовуються три зображення (образ відомої людини, інший образ тієї ж людини і образ іншої людини) [8]. Цей крок вимагає великого набору даних і великої потужності комп'ютера. Але має бути виконано лише один раз. Деякі попередньо навчені нейронні мережі також доступні в Інтернеті.

Пошук імені людини за кодуванням. Останній крок дуже простий. Обличчя, яке аналізується, порівнюється з обличчями, які є в нашій базі даних. Для цього бібліотека Python використовує векторну машину

підтримки (SVM). В принципі, можна використовувати будь-які інші алгоритми класифікації.

Продуктивність алгоритму представлена на зображенні, показаному на рис. 3.6. Можна побачити, що він працює добре, навіть якщо Боб Келсо та Джон Доріан не дивляться прямо в камеру.

У випадку Джона Доріана голова розташована майже перпендикулярно до осі камери. Слід також зазначити, що все зображення набагато темніше середнього. Незважаючи на це, алгоритм зміг ідеально розпізнати їх обох.



Рисунок 3.6. Приклад алгоритму розпізнавання обличчя

Впровадження системи розпізнавання обличчя на основі глибокого навчання за допомогою бібліотеки `face_recognition`:

Щоб інсталиувати бібліотеку розпізнавання облич, нам потрібно спочатку інсталиувати `dlib`. Це сучасний інструментарій C++, який містить

алгоритми та інструменти, пов'язані з ML. Face Recognition бібліотеку розпізнавання обличчя можна встановити після dlib, а також OpenCV для деякої попередньої обробки зображень.

```
import cv
import numpy as nmp
import face-recognition
```

Ми завершили встановлення та імпорт бібліотек. Настав час завантажити зразки зображень у бібліотеку face_recognition. Бібліотека face_recognition підтримує лише формат зображень BGR. Під час друку вихідного зображення ми повинні перетворити його в RGB за допомогою OpenCV.

```
image_bgr = face_recognition.load_image_file('images/kodi.jpg')
image_rgb = cv.cvtColor(img_bgr,cv.COLOR_BGR2RGB)
cv.imshow('bgr image', image_bgr)
cv.imshow('rgb image', image_rgb)
cv.waitKey()
```

Бібліотека face_recognition може швидко знаходити обличчя самостійно, нам не потрібно використовувати haar_cascade та інші методи.

```
image_kodi=face_recognition.load_image_file('images/kodi.jpg')
image_kodi_rgb = cv.cvtColor(image_kodi,cv.COLOR_BGR2RGB)
#----- Face Detecting -----
face = face_recognition.face_locations(image_kodi_rgb)[0]
copy = img_kodi_rgb.copy()
# ----- Drawing box around Faces-----
cv2.rectangle(copy, (face[2], face[0]),(face[1], face[3]), (255,0,255), 3)
cv2.imshow('copy', copy)
cv2.imshow('KODI',image_kodi_rgb)
cv2.waitKey(0)
```

Бібліотека face_recognition заснована на глибокому навчанні, вона підтримує одноразове навчання, що означає, що їй потрібне одне

зображення, щоб навчитися виявляти особу.

```

image_kodi = face_recognition.load_image_file('images/kodi.jpg')
image_kodi = cv.cvtColor(img_modi,cv.COLOR_BGR2RGB)
#-----find face location
face = face_recognition.face_locations(image_kodi)[0]
#--Converting image in to encoding
train_encode = face_recognition.face_encodings(image_kodi)[0]
#----- test an image
test_image = face_recognition.load_image_file('images/kodi2.jpg')
test_image = cv2.cvtColor(test_image, cv.COLOR_BGR2RGB)
test_encode = face_recognition.face_encodings(test_image)[0]
print(face_recognition.compare_faces([train_encode],test_encode))
cv2.rectangle(image_kodi,(face[2], face[0]),(face[1], face[3]), (255,0,0), 1)
cv2.imshow('image_kodi', image_kodi)
cv2.waitKey(0)

```

Наведений вище код зробив дві фотографії і повернув True, оскільки обидві фотографії були однієї людини.

`face_recognition.face_encodings(imgelon)[0]` → повертає кодування переданого зображення.

`face_recognition.compare_face([test_encode],train_encode)` → отримує список навчених кодувань і тестове кодування невідомого зображення. Він повертає True, якщо обидва тестові кодування збігаються в кодуванні; інакше він повертається False.

Процес роботи розпізнавання обличчя:

Ми передаємо фотографію людини моделі та її ім'я.

Модель бере кожне зображення, перетворює його в певне числове кодування та зберігає в списку, а всі мітки (імена осіб) в іншому списку.

На фазі передбачення, коли ми передаємо зображення невідомої особи, модель розпізнавання перетворює зображення незнайомої людини в кодування.

Після перетворення зображення невідомої особи в кодування він намагається знайти найбільш схоже кодування на основі параметра відстані. Збережене кодування з найменшою віддаленістю від кодування невідомої особи буде найбільш близьким.

Отримавши кодування з найбільшою відповідністю, ми беремо індекс цього кодування зі списку та використовуємо індексацію. Знаходимо ім'я виявленої людини.

Висновки до розділу 3

У цій частині описано та проаналізовано бібліотеки для виявлення та розпізнавання облич. Виявлення та розпізнавання облич є двома напрямками інтенсивних досліджень, оскільки вони забезпечують кращу взаємодію між комп'ютерними системами або роботами з одного боку та людьми з іншого.

У першій частині розділу надано короткий огляд найпоширеніших бібліотек, пов'язаних із обробкою зображень і комп'ютерним зором. У другій частині мова йшла про бібліотеку OpenCV, як найбільш підходящу бібліотеку для комп'ютерного зору. А у заключній частині розділу, було розібрано face recognition бібліотеку розпізнавання обличчя та які різні проблеми, і якими способами вона дозволяє їх вирішити.

РОЗДІЛ 4

ПРАКТИЧНА ЧАСТИНА

4.1. Інструментарій для розробки

Система розроблена з використанням об'єктно-орієнтованої мови програмування Python та кросплатформного інтегрованого середовища розробки Pycharm. Python — це інтерпретована об'єктно-орієнтована мова програмування високого рівня із динамічною типізацією.

Високорівневі структури даних разом з динамічною семантикою і динамічним зв'язуванням роблять його привабливим для швидкої розробки додатків і як спосіб інтеграції існуючих компонентів. Python підтримує модулі й пакети модулів, що дозволяє повторному використанню коду. Python і стандартні бібліотеки доступні у скомпільованій та вихідній формі на всіх основних платформах.

Для полегшення процесу розробки використовувалась бібліотека OpenCV. Ця бібліотека розроблена на C/C++ мовах програмування, а також має інтерфейси для Java, Python та інших мов. Вона підтримує Linux, Mac OS, Windows, Android та iOS. Вона містить алгоритми обробки зображень, реконструкції та очищення, розпізнавання образів, захоплення відео, відстеження об'єктів, калібрування камери тощо. Бібліотека поширюється за ліцензією BSD, що означає, що її можна вільно використовувати в наукових і комерційних цілях[1] [7].

Присутність чималої кількості реалізованих алгоритмів комп'ютерного зору, а також велика кількість теоретичних робіт з їх застосування, роблять бібліотеку OpenCV ідеальним варіантом для використання у проектах, присвячених вирішенню задач комп'ютерного зору.

4.2 Аналіз потоків даних

На основі вхідних даних та списку класів облич, методом найближчого сусіда відбувається класифікація обличчя. Реалізації методу найближчого сусіда в основному механізмі програмного забезпечення у вигляді блок-схеми наведена на (рис. 4.1).

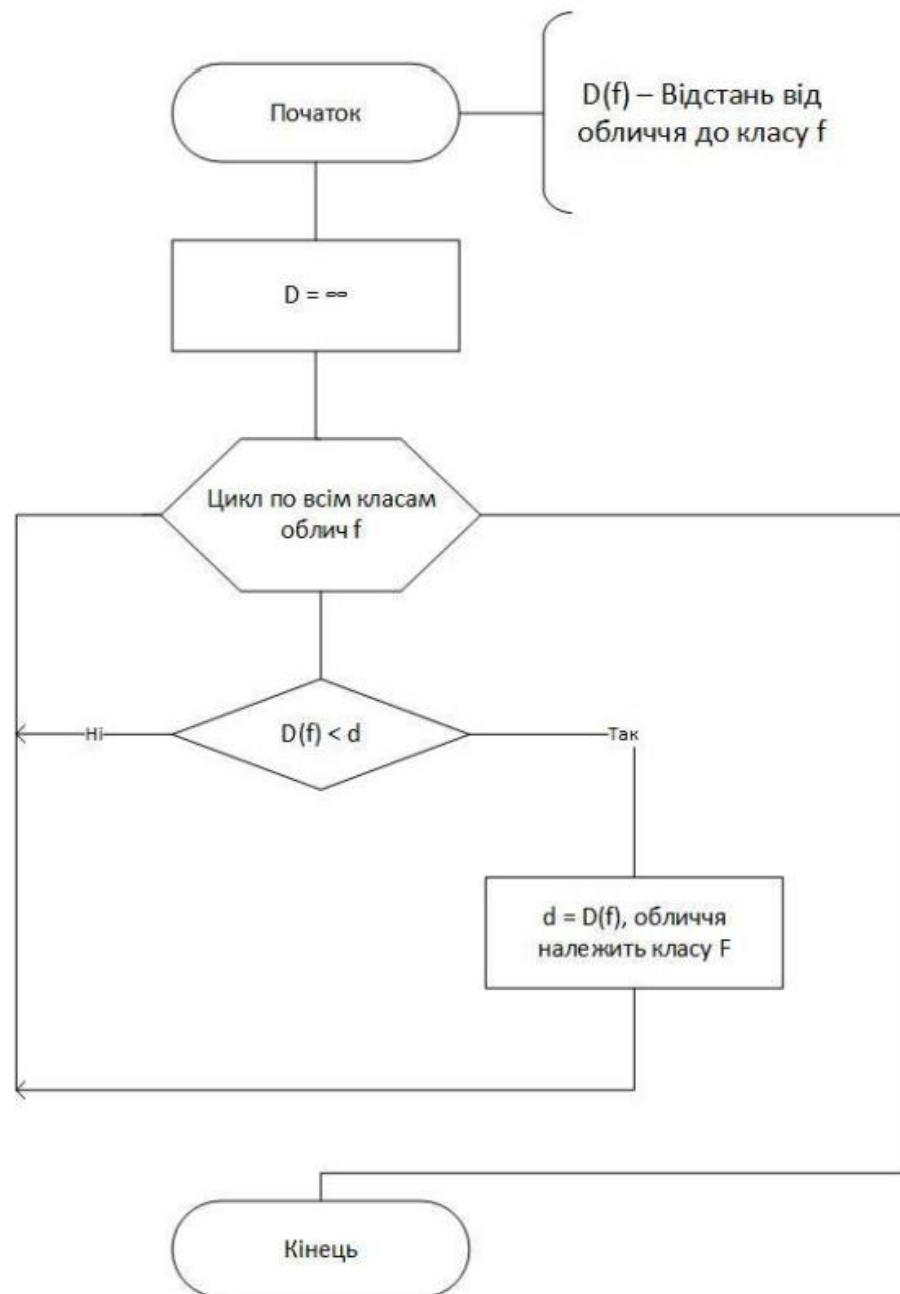


Рисунок 4.1. Блок-схема реалізації ідентифікації обличчя методом найближчого сусіда

Після визначення класу обличчя, відстань до класу порівнюється з пороговим значенням, а потім обличчя додається до списку розпізнаних облич або нерозпізнаних. Блок-схема алгоритму створення списку розпізнаних облич зображена на (рис. 4.2).

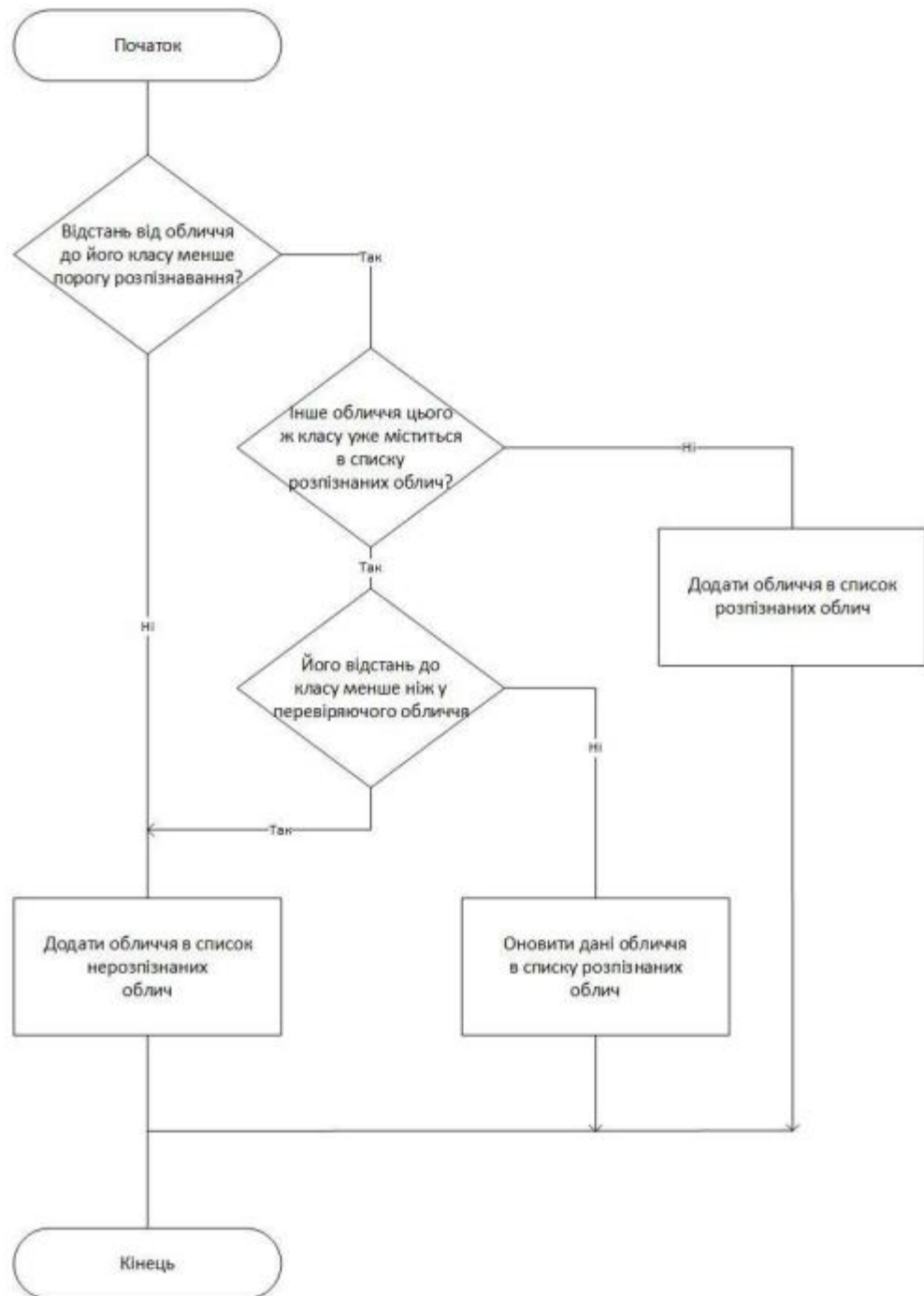


Рисунок 4.2. Блок-схема алгоритму створення списків розпізнаних і нерозпізнаних облич

4.3 Опис інтерфейсу системи

При відкритті програмного застосунку, користувача зустрічає стартове меню, зображення якого наведено на рис.4.3.

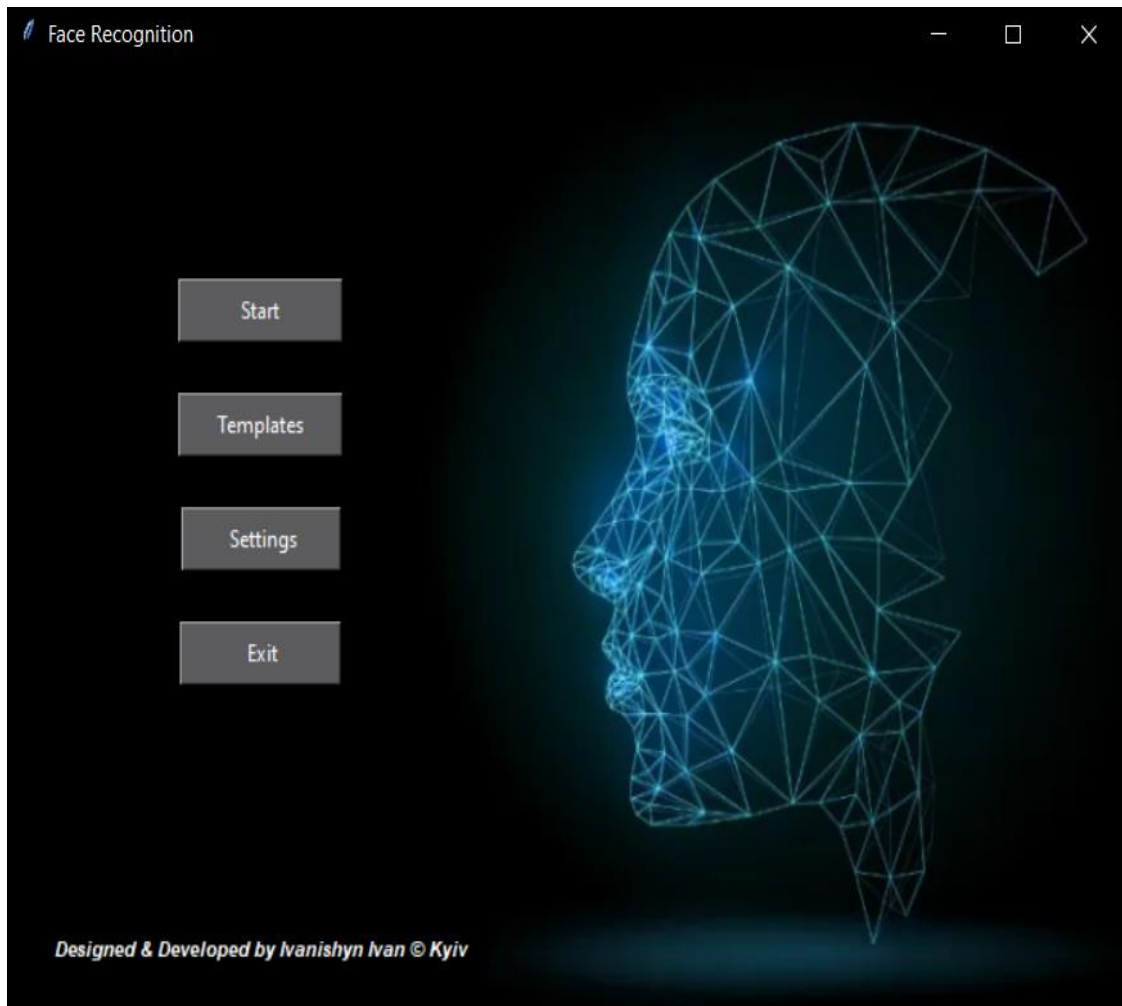


Рисунок 4.3 – Стартове меню програмного забезпечення

Дане меню складається з чотирьох функціональних кнопок, а саме:

«Start» - кнопка, яка безпосередньо запускає основний механізм – детекції та розпізнавання облич у режимі відеопотоку;

«Templates» - кнопка, котра перенаправляє користувача у вікно, зображення якого наведено на рис.4.4, де він має змогу видаляти та добавляти шаблонні фотографії, які у майбутньому використовуватимуться для класифікації обличчя методом найближчого сусіда;

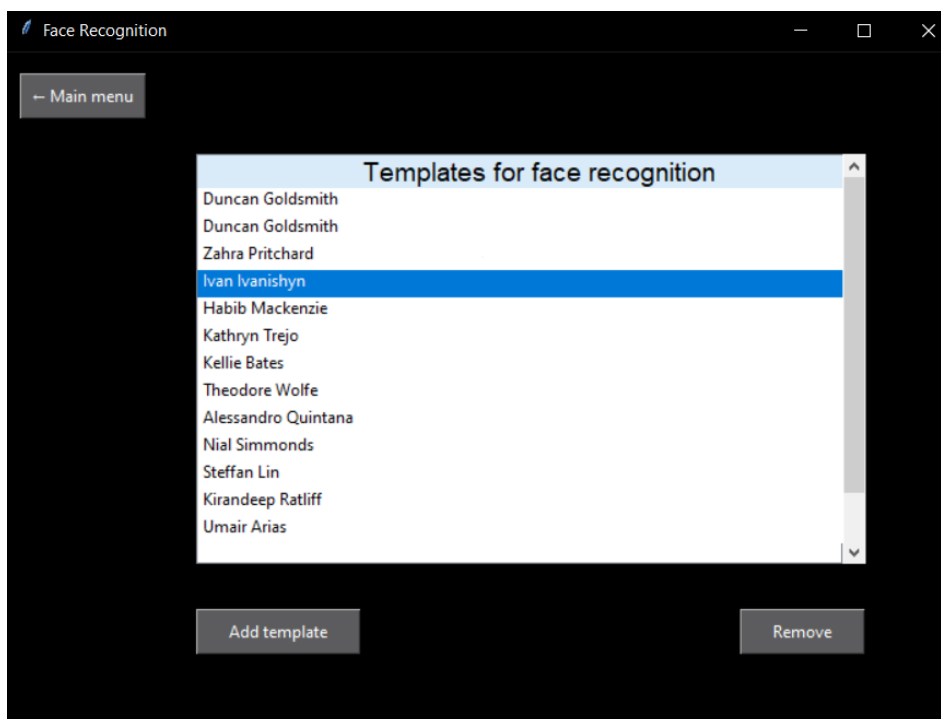


Рисунок 4.4. Вид вікна сторінки «Templates»

У цьому вікні, у користувача є три функціональні можливості:
 Повернення до основного меню за допомогою кнопки «Main Menu»;
 Видалення темплейту з списку «Templates for face recognition» за допомогою кнопки «Remove»;

Додавання темплейту до списку за допомогою кнопки «Add template», при натисканні на яку, відкривається додаткове вікно для вибору зображень (рис. 4.5).

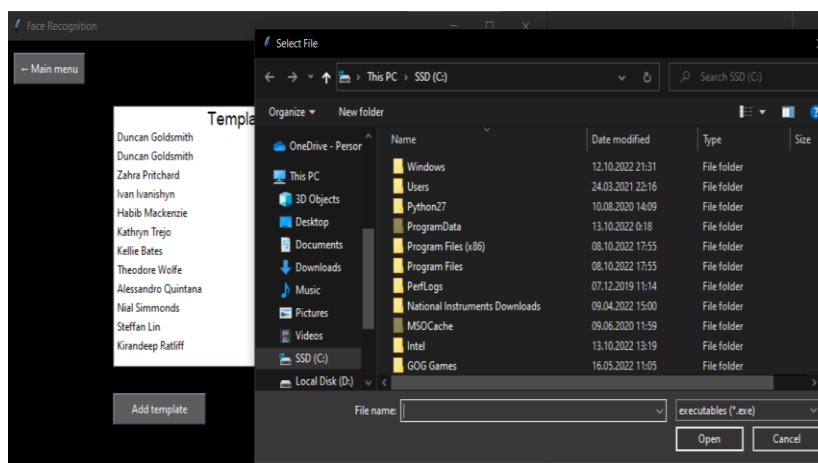


Рисунок 4.5. Зображення механізму додавання темплейту

«Settings» - кнопка, яка відкриває вікно з групою налаштувань, зображення якого наведено на рис.4.6, де користувачу надається можливість змінити поведінку системи під свої власні потреби;

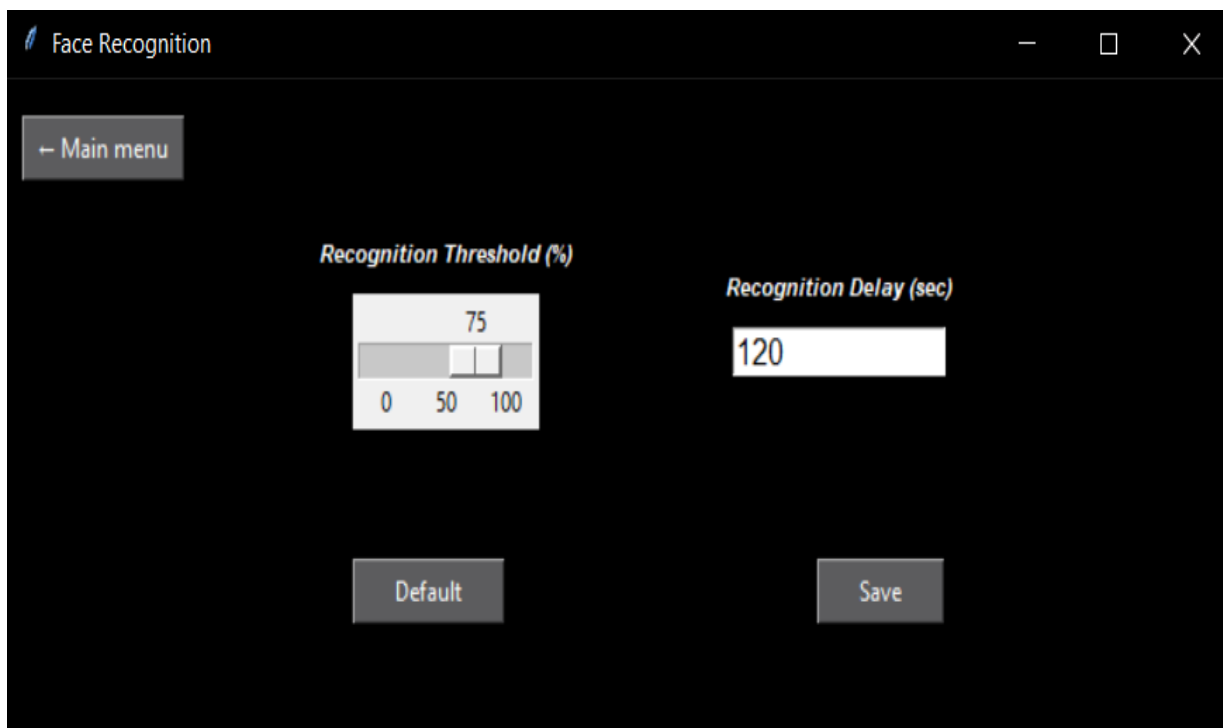


Рисунок 4.6. Вид вікна сторінки «Settings»

«Exit» - кнопка, котра завершує роботу та закриває програмне забезпечення.

4.4 Тестування розробленої системи

Як було сказано раніше, при натисканні користувачем кнопки «Start», відкривається окреме вікно, де у режимі реального часу відбуватиметься процес детекції та розпізнавання облич.

Продуктивність класифікатора облич перевірялася в умовах, описаних у розділі про аналіз завдання розпізнавання. Приклад роботи реалізованої системи розпізнавання обличчя у відеопотоках наведено на (рис. 4.7).

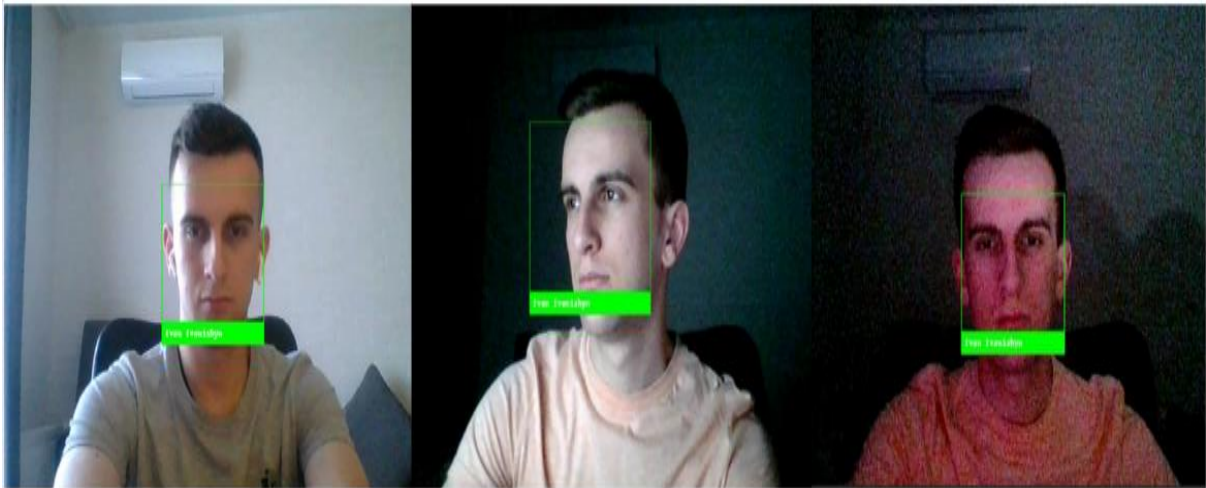


Рисунок 4.7. – Приклад роботи реалізованої системи розпізнавання обличчя

Результати тестування виявили, що точність класифікатора під час обробки відеопотоку перевищує 90% правильно розпізнаних зображень.

Однак слід звернути увагу на чутливість розробленої системи до сильних змін немонотонного освітлення, а також до змін положення та нахилу розпізнаних облич, які не були враховані під час збору шаблонних зображень. Якщо умови, необхідні для нормальної роботи програми, не виконуються, точність класифікації знижується.

Також відразу після успішної ідентифікації особистості, результати записуються у журнал, де зберігатимуться записи, які говоритимуть кого і коли саме розпізнала система, наглядний приклад зображено на рис. 4.8.

	A
1	Name,Time
2	Jeff Bezos, 19/10/2022 19:03:29
3	Bill Gares, 19/10/2022 19:07:45
4	Ivan Ivanishyn, 19/10/2022 19:12:05

Рисунок 4.8. Приклад журналу ідентифікованих осіб

Висновки до розділу 4

Цей розділ описує та аналізує потоки даних розробленої системи. Складена структурна схема алгоритму. Реалізовано графічний інтерфейс користувача програми. Описано різні області інтерфейсу.

Проведено тестування розробленої системи та реалізованого класифікатора облич. Результатом є більш ніж 90% правильне розпізнавання обличчя зі зниженням продуктивності, якщо умови, описані в розділі аналізу проблеми, не виконуються.

РОЗДІЛ 5

РОЗРОБКА СТАРТАП ПРОЕКТУ

5.1. Резюме бізнес плану

Коротке резюме стартап-проекту:

1. Найменування проекту: система розпізнавання облич, призначена для автоматичної ідентифікації особи по відеозображенню.
2. Системи обмеження доступу та ідентифікації осостають дедалі більш популярними із розвитком інформаційних технологій.
3. Перший час основними споживачами будуть навчальні заклади, яким необхідна додаткова система контролю відвідуваності.
4. Персонал: науковий керівник - старший викладач кафедри біомедичної кібернетики, Бовсуновська Катерина Сергіївна: консультація та опис поверхневого дизайну, виконавець - Іванішин Іван Володимирович: проектування, розробка backend та frontend частин системи та тестування.
5. Для реалізації та впровадження проекту необхідно приблизно 300 тис. грн.

5.2. Опис ідеї стартап-проекту

У табл. 5.1 подано огляд змісту ідеї та основних можливих потенційних ринків, у межах яких слід шукати групи потенційних клієнтів.

За характером інновацій – це дослідно-технологічна робота: вдосконалення та оптимізація методу розпізнавання обличчя за допомогою глибинного навчання.

Спрямованість продукту полягає в детекції та розпізнаванні обличь в режимі реального часу.

Таблиця 5.1

Зміст ідеї проекту

Зміст ідеї	Напрямки застосування	Цінність для споживачів
Додаток, в основі якого йдуть алгоритми глибинного навчання, що допомагають знайти людей на зображенні та згенерувати структурну модель обличчя.	У навчальних закладах, на великих підприємствах в якості системи контролю доступу	Підвищення рівня безпеки у цих будівлях
	В розумних будинках в якості системи безпеки	Контроль системи безпеки власного будинку

На ринку є аналоги подібних систем, але більшість із них розробляються виключно для вирішення конкретного завдання, занадто дорогі або мають обмежену функціональність.

Після визначення груп потенційних клієнтів, було проведено SWOT-аналіз проекту, представлений у табл. 5.2, який допоміг визначити сильні і слабкі сторони, можливості та можливі загрози.

Таблиця 5.2.

SWOT-аналіз проекту

<i>Сильні сторони (S):</i> ціна, універсальність, невелика кількість виконавців, інноваційні технології.	<i>Слабкі сторони (W):</i> високі вимоги до операційної системи; відсутність репутації компанії.
<i>Можливості (O):</i> попит, секьюрність, розширення цільової аудиторії.	<i>Загрози (T):</i> конкуренція, програмні та технологічні обмеження, плагіат та політичні фактори.

Проведено порівняльний аналіз показників і визначено слабкі, нейтральні та сильні сторони цього проекту, порівнюючи з вже існуючими аналогами. Усі результати наведено у табл. 5.3.

Таблиця 5.3.

Аналіз сильних, слабких й нейтральних характеристик ідеї проекту

№	Техніко- економічні характеристики ідеї	Продукція конкурентів			W	N	S
		Мій проект	Felena Soft	MultiPerson Recognition			
1.	Вартість одного продукту	2500\$	5000\$	4000\$	-	-	+
2.	Точність розпізнавання	80-99%	75-99%	70-75%	-	-	+
3.	Вимоги до операційної системи	Високі вимоги до операційної системи	Високі апаратні вимоги	Середні вимоги до операційної системи	+	-	-
4.	Можливість розпізнавання в режимі реального часу	Присутня	Присутня	Відсутня	-	+	+
5.	Відомість бренду	Невідомий	Відомий	Невідомий	+	-	-

Розробка ринкової стратегії є доволі важливим кроком у розвитку проекту. Вона визначає стратегії охоплення ринку та каналів збуту. Опис потенційних сегментів споживачів наведено в табл. 5.4.

Таблиця 5.4.

Взаємовідносини зі споживачами та канали збуту

№	Сегмент споживачів	Особливості поведінки	Вимоги споживачів	Канали збуту
1.	Навчальні заклади	- надання інструкції щодо використання програмного продукту	- до продукції: 1) дешевизна - до компанії-постачальника (виробника): 1) наявність чітко сформульованої документації	Тендери на рівні НЗ

5.3 Технологічний аудит ідеї проекту

У цьому підрозділі описується технологічний аудит, який може бути використаний для реалізації ідеї проекту (технологія створення продукту). Визначення доцільності реалізації ідеї проекту передбачає аналіз складових:

1. Яка технологія буде використана для виготовлення продукту на основі ідеї проекту?
2. Чи існують ці технології чи їх потрібно розвивати/удосконалювати?
3. Чи доступні ці технології авторам проекту?

У табл. 5.5. враховуються всі перераховані вище компоненти.

Таблиця 5.5.

Технологічний аудит ідеї проекту

№	Технології її реалізації	Наявність технологій	Доступність технологій
1.	Реалізація проекту на мові програмування Python	Наявна	Доступна
	Реалізація проекту в середовищі PyCharm	Наявна	Доступна
	Багатоплатформова бібліотека Dlib	Наявна	Доступна
	Бібліотека комп'ютерного зору OpenCV	Наявна	Доступна

Обрана технологія реалізації ідеї проекту: реалізація проекту на мові програмування Python в середовищі PyCharm з використанням бібліотек Dlib та OpenCV.

5.4. Обґрунтування ресурсів та витрат проекту

Як вже було сказано в першому підрозділі, для виробництва та реалізації стартап-проекту необхідно близько 300 тис. грн. Щоб покрити ці витрати, цей продукт спочатку буде просуватись комерційним навчальним закладам, зацікавленим у додатковій системі відстеження відвідуваності, а вже потім, на тендерному рівні, послуга буде запропонована державним навчальним закладам. Розрахунок загальних початкових інвестицій наведений у табл. 5.6.

Таблиця 5.6.

Загальні початкові інвестиційні витрати

№	Назва етапу	Терміни виконання	Обсяги фінансування, тис. грн.
1.	Проведення досліджень	3 місяці	50
2.	Розробка або придбання технології	1 місяць	50
3.	Придбання устаткування	-	100
4.	Організація діяльності та запуск проекту	6 місяців	50
5.	Витрати на управління	2 місяці	50
6.	Інші витрати (ризики)	-	-
<i>Разом:</i>		<i>1 рік</i>	<i>300</i>

Розрахунок виробничих витрат наведений у табл. 5.7.

Таблиця 5.7.

Загальні виробничі витрати

№ з/п	Стаття витрат	Сукупні витрати за період, тис. грн.		
		2022	2023	2024
1.	Загальногосподарські витрати	-	-	-
1.1.	Витрати на оренду та утримання приміщень, обладнання	200	300	350
1.2.	Комунальні витрати	65	100	125
1.3.	Витрати на збут, просування та рекламу	50	75	100
2.	Витрати на матеріальні ресурси (комплектуючі, сировина)	-	-	-
3.	Витрати на оплату праці	1500	2000	2500
4.	Інші витрати (якщо є)	-	-	-
<i>Разом:</i>		<i>1815</i>	<i>2475</i>	<i>3075</i>

5.4. Обґрунтування ресурсів та витрат проекту

Найважливішим і водночас найскладнішим кроком у аналізі ефективності інвестицій є оцінка грошових потоків проекту, тобто інвестиційних витрат, необхідних для реалізації проекту, і підсумкового річного чистого грошового доходу.

Формування грошового потоку від реалізації проекту наведено у табл. 5.8.

Таблиця 5.8.

Загальні виробничі витрати

№	Показник	Значення по роках			Разом
		2022	2023	2024	
1.	Надходження від проекту (виручка від реалізації продукції, послуг)	750 000	2 000 000	10 000 000	12 750 000
2.	Загальні витрати	2 115 000	2 475 000	3 075 000	7 665 000
3.	Грошовий потік	-1 365 000	-475 000	6 925 000	5 085 000
4.	Акумуляований грошовий потік	-1 365 000	-1 840 000	5 085 000	-

Розрахунок вартості проекту з використанням різних методів продемонстрований у табл. 5.9.

Таблиця 5.9.

Розрахунок вартості проекту з точки зору інвестора

Найменування методу	Умови застосування	Отримана оцінка	Частка венчурного інвестора	Рішення щодо інвестування в проект
Метод мультиплікаторів	Стартап на етапі демонстраційного MVP	364 тис.грн	15%	Позитивне
ROV-метод	Стартап на етапі демонстраційного MVP	369 тис.грн	15%	Позитивне
Метод «хокейної ключки»	Стартап на етапі демонстраційного MVP	364 тис.грн	15%	Позитивне
Перший чиказький метод	Стартап на етапі демонстраційного MVP	373 тис.грн	15%	Позитивне
Метод скорингу	Стартап на етапі демонстраційного MVP	360 тис.грн	15%	Позитивне

Висновки до розділу 5

З наведеного вище аналізу можна зробити висновок, що технологія детекції та ідентифікації людини стає дуже популярною з кожним роком, і відповідно, збільшуються області, напрямки застосування цих систем розширюються, а ринок користувачів зростає, що призводить до значного попиту на таку продукцію.

При відносно низьких інвестиційних витратах доходи від надання послуг на 2024 рік заплановані в сумі 10 000 000 грн, що підвищує привабливість проекту, оскільки через 2,5 роки доходи мають перевищити витрати в 3 рази.

ЗАГАЛЬНІ ВИСНОВКИ

На основі методів Віюлі-Джонса та k -найближчих сусідів розроблено систему розпізнавання облич у відеопотоках. Метод Віюлі-Джонса використовувався для виявлення облич у відеопотокових зображеннях. Виявлені обличчя були класифіковані за методом найближчого сусіда.

Випробування розробленої системи показали точність розпізнавання обличчя на рівні близько 91% при обробці зображення відеопотоку реального часу з веб-камери, що підтвердило ефективність обраного методу.

Реалізована система може бути використана для вирішення різноманітних завдань відеоаналізу, і що більш важливо, вона безпосередньо застосовна в системах контролю доступу та ідентифікації особистості.

Основними напрямками подальшого розвитку розробленого алгоритму можна назвати вдосконалення роботи класифікатора обличчя і вдосконалення графічного інтерфейсу користувача. Також слід зазначити, що структура розробленого застосунку дозволяє легко змінювати окремі модулі, що відкриває більші можливості для розвитку та вдосконалення системи на різних апаратних платформах.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. About OpenCV // URL: <http://opencv.org/about.html>.
2. Barni M. Privacy-Preserving Fingercod Authentication [Електронний ресурс] / Mauro Barni – Режим доступу до ресурсу: http://piurilabs.di.unimi.it/Papers/awmm_2010.pdf
3. C. Fuhrer, J. E. Solem, and O. Verdier, ScientificComputing with Python 3, Packt Publishing Ltd,2016.
4. D. Roth, The SNoW Learning Architecture // Technical Report UIUCDCS-R-99- 2102 . UIUC Computer Science Department, 1999.
5. Dlib Python API Tutorials [Electronic resource] – Access mode: <http://dlib.net/python/index.html>
6. E. Osuna, R. Freund, F. Girosi. Training support vector machines:an application to face detection // In Proceedings of Computer Vision and pattern Recognition 1997, C. 130-136
7. EmguCV// URL: http://www.emgu.com/wiki/index.php/Main_Page
8. F. Schroff, D. Kalenichenko, and J. Philbin,“Facenet: A unified embedding for face recognition and clustering,” Proceedings of the IEEE conference on computer vision and pattern recognition,pp. 815-823, 2015.
9. Face ID: [Electronic resource] – Access mode: <https://support.apple.com/uk-ua/HT208109>
10. G.J. Edwards, T.F. Cootes, C.J. Taylor. Face Recognition Using Active Appearance Models // Computer Vision — ECCV’98, Volume 1407 of the series Lecture Notes in Computer Science. C. 581–595
11. H. A. Rowley, S. Baluja, .T. Kanade. Neural Network-Based Face Detection // PAMI, January 1998.
12. H. A. Rowley, S. Baluja, .T. Kanade. Neural Network-Based Face Detection // PAMI, January 1998.
13. H. W. Ng, and S.Winkler, “Adata-driven approach to cleaning large face datasets,” IEEE International Conference on Image Processing (ICIP), pp.

- 343-347, 2014.
14. Jochen Maydt and Rainer Lienhart. Face Detection with Support Vector Machines and a Very Large Set of Linear Features, IEEE ICME 2002, Lousanne, Switzerland, Aug. 2002
 15. K. He, X. Zhang, S. Ren, and J. Sun, Deep residual learning for image recognition,” In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778, 2016.
 16. K.-R. Muller. S. Mika, G. Ratsch, K. Tsuda. and B. Scholkopf. An introduction to kernel-based learning algorithms, IEEE Transactions on Neural Networks, 12(2). pp. 181–201, 2001.
 17. M. L. Hetland, Beginning Python: from novice to professional, 3rd Ed., Apress, 2017.
 18. M. Nilsson, M. Dahl, I. Claesson. The successive mean quantization transform / Proceedings of IEEE Int. Conf.ICASSP 2005, Vol. 4 / The Institute of Electrical and Electronics Engineers Signal Processing Society C. 429 – 432.
 19. M. Turk, A. Pentland. Eigenfaces for recognition. // Cognitive Neuroscience, 1991, 88 №3(1), C.71–86.
 20. Machine learning methods // Graphicon. URL: <http://www.graphicon.ru/oldgr/ru/publications/text/gc2006avezh.pdf>.
 21. N. Dalal, and B. Trigs, “Histograms of oriented gradients for human detection,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 886-893, 2005.
 22. O. M. Parkhi, A. Vedaldi, and A. Zisserman,“Deep face recognition,” In British Machine Vision Conference, vol. 1, no. 3, p. 6, September, 2015.
 23. O. Papageorgiou, P. Papageorgiou. A general framework for object detection / International Conference on Computer Vision, 1998 / The Institute of Electrical and Electronics Engineers, Inc. C. 555–562.
 24. OpenCV: OpenCV Tutorials [Electronic resource] – Access mode: <https://docs.opencv.org/2.4/doc/tutorials/tutorials.html>

25. P. N. Belhumeur, J. P. Hespanha, D. J. Kriegman. Eigenfaces vs. Fisherfaces Recognition Using Class Specific Linear Projection // IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 19, July 1997, №7 C. 711–720.
26. P. Viola, and M. Jones, “Rapid object detection using a boosted cascade of simple features,” Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. I-511-I-518, 2001.
27. P. Viola, M. Jones. Rapid object detection using a boosted cascade of simple features / 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Vol. 1. 8–14 December 2001 / The Institute of Electrical and Electronics Engineers, Inc. C. 511–518
28. Paul Viola and Michael Jones. Robust real-time object detection. In International Journal of Computer Vision, 2001.
29. R. V. Hattem, Mastering Python: master the art of writing beautiful and powerful Python by using all of the features that Python 3.5 offers, Packt Publishing, 2016
30. T. Ahonen, A. Hadid, M. Pietikäinen. Face Description with Local Binary Patterns: Application to Face Recognition // IEEE Trans. Pattern Analysis and Machine Intelligence, 1996, №28(12), C. 2037–2041.
31. T. Ojala, M. Pietikäinen, D. Harwood . A Comparative Study of Texture Measures with Classification Based on Feature Distributions // Pattern Recognition, Vol. 29,1996, C. 51–59.
32. T. Rawlinson, A. Bhalerao, L. Wang. Principles and Methods for Face Recognition and Face Modelling // Handbook of research on computational forensics, digital crime and investigation: methods and solutions. IGI Global, C. 53-78.
33. U. Prabhu, K. Seshardi. Facial Recognition Using Active Shape Models, Local Patches and Support Vector Machines // ECE Department Carnegie Mellon University.

34. V. Gupta, D. Sharma. A Study of Various Face Detection Methods // International Journal of Advanced Research in Computer and Communication Engineering Vol. 3, May 2014, №5. С. 6694–6697.
35. V. Kazemi, and J. Sullivan, “One millisecond face alignment with an ensemble of regression trees,” Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1867-1874, 2014.
36. W.Y. Zhao, R. Chellappa. Image-based Face Recognition — Issues and Methods // Image recognition and Classification, 2002, С. 375–402.
37. Yale Face Database B // UCSD Computer Vision. URL: <http://vision.ucsd.edu/~leekc/ExtYaleDatabase/ExtYaleB.html>.
38. Zhang, G., Huang, X., Li, S.Z., Wang, Y., Wu, X.: Boosting local binary pattern (LBP)-based face recognition. In: Li, S.Z., Lai, J.-H., Tan, T., Feng, G.-C., Wang, Y. (eds.) SINOBIOMETRICS 2004. LNCS, vol. 3338, pp. 180– 187. Springer, Heidelberg (2004).
39. Броневи́ч А. Н. Лекции по методам машинного обучения // URL: http://window.edu.ru/resource/800/73800/files/lect_Lepskiy_Bronevich_pss.pdf.
40. Вежневeц В. Обнаружение и локализация лица на изображении / В. Вежневeц, А. Дегтярева. // Компьютерная графика и мультимедиа. — 2003. — №1(3).
41. Іванішин І. В. Використання методів Віоли-Джонса і к найближчих сусідів для детекції та розпізнавання облич // Міжнародний науковий журнал "Інтернаука". — 2022. — №11. <https://doi.org/10.25313/2520-2057-2022-11-8281>.
42. Местецкий Л.М. Математические методы распознавания образов, М.: МГУ, ВМиК, 2002–2004. – 85 с.
43. Р. Гонсалес, Р. Вудс. Цифровая обработка изображений, М.: Техносфера, 2005. – 1072 с.