

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління  
(повна назва)

Кафедра Автоматизації проектування обчислювальної техніки  
(повна назва)

## КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти перший (бакалаврський)  
(рівень вищої освіти)

Інтелектуальна система детекції обличчя  
(тема)

Виконав:

здобувач 4 року навчання


групи КІУКІ-21-9

Роскошний В.Р.  
(прізвище, ініціали)

Спеціальність 123 Комп'ютерна інженерія

Тип програми освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерна інженерія  
(повна назва освітньої програми)

Керівник  Мірошник А.М.  
(підпис) (прізвище, ініціали)

Допускається до захисту

Зав. кафедри \_\_\_\_\_  
(підпис)

Чумаченко С.В.  
(прізвище, ініціали)

2025 р.

Харківський національний університет радіоелектроніки

Факультет Комп'ютерної інженерії та управління

Кафедра Автоматизації проектування обчислювальної техніки

Рівень вищої освіти перший (бакалаврський)

Спеціальність 123 Комп'ютерна інженерія  
(шифр і назва)

Тип програми Освітньо-професійна  
(освітньо-професійна або освітньо-наукова)

Освітня програма Комп'ютерна інженерія  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

« \_\_\_\_\_ » \_\_\_\_\_ 2025 р.

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві Роскошному Владиславу Руслановичу  
(прізвище, ім'я, по батькові)

1. Тема роботи (проекту) Інтелектуальна система детекції обличчя

затверджена наказом по університету від "21 " 05 2025 р. № 403Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії 11.06.2025

3. Вихідні дані до роботи (проекту) \_\_\_\_\_

Технічне завдання на розробку інтелектуальної системи відстеження обличчя

Специфікації мови програмування Python

Специфікації мікроконтролера AVR ATmega328P

Інтегроване середовище розробки програмного забезпечення PyCharm

4. Перелік питань, що потрібно опрацювати у роботі \_\_\_\_\_

Face Tracking: теоретичні аспекти та аналіз існуючих рішень

Реалізація апаратної складової системи

Реалізація програмного забезпечення системи

Реалізація макетного зразка та результати дослідної експлуатації

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) Презентація (22 слайди)

---

---

---

---

---

6. Консультанти розділів роботи (проекту)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

7. Дата видачі завдання 06.05.2025

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи (проекту)	Термін виконання етапів проекту (роботи)	Примітка
1	Видача теми проекту, узгодження і затвердження теми	06.05.2025 - 10.05.2025	
2	Аналіз проблемної галузі, постановка задачі, вибір інструментальних засобів	10.05.2025 - 17.05.2025	
3	Розробка моделі системи	17.05.2025 - 24.05.2025	
4	Розробка апаратної платформи системи	24.05.2025 - 31.05.2025	
5	Розробка програмного забезпечення системи	31.05.2025 - 05.06.2025	
6	Проведення випробування системи	05.06.2025 - 07.06.2025	
7	Оформлення пояснювальної записки	07.06.2025 - 10.06.2025	
8	Перевірка виконаного проекту керівником допуск до захисту	10.06.2025 - 13.06.2025	
9	Захист проекту	13.06.2025 - 23.06.2025	

Здобувач



(підпис)

Роскошній В.Р.

Керівник роботи (проекту)



(підпис)

старший викладач Мірошник А.М.

(посада, прізвище, ініціали)

## РЕФЕРАТ

Пояснювальна записка до атестаційної роботи містить 61 сторінку, 21 рисунок, 2 таблиці, 4 додатки, 16 джерел за переліком посилань.

ІНТЕЛЕКТУАЛЬНА СИСТЕМА, ВІДСТЕЖЕННЯ ОБЛИЧЧЯ, КОМП'ЮТЕРНИЙ ЗІР, НЕЙРОННІ МЕРЕЖІ, ARDUINO, C++, PYTHON.

Об'єкт розробки – інтелектуальна система відстеження обличчя, призначена для автоматичного виявлення, ідентифікації та супроводження обличчя в реальному часі.

Предмет розробки – апаратно-програмний комплекс, що забезпечує функціонування системи відстеження обличчя з використанням алгоритмів комп'ютерного зору та машинного навчання.

Мета розробки – створення інтелектуальної системи відстеження обличчя, яка забезпечує високу точність і швидкість виявлення обличчя у реальному часі, стабільну роботу апаратної частини, інтеграцію з модулем комп'ютерного зору, а також можливість аналізу та збереження отриманих даних для подальшого використання.

У першому розділі кваліфікаційної роботи досліджено усі теоретичні аспекти комп'ютерного зору та систем відстеження обличчя, а також наявні технічні рішення у цій сфері.

У другому розділі представлено реалізацію апаратної складової системи з підбором конкретних технічних компонентів.

У третьому розділі наведено алгоритм та реалізацію програмного забезпечення системи, яке складається з двох частин – програмне забезпечення для плати Arduino, та програмне забезпечення для інтерфейсу системи.

В останньому розділі проведено тестування макетного зразка системи та наведено результати, які показали працездатність проектованої системи.

## ABSTRACT

Bachelor's thesis contains 61 pages, 21 figures, 2 tables, 4 appendices, 16 sources according to the list of references.

INTELLIGENT SYSTEM, FACE TRACKING, COMPUTER VISION, NEURAL NETWORKS, ARDUINO, C++, PYTHON.

Object of the development – an intelligent face-tracking system designed for automatic detection, identification, and tracking of faces in real time.

Subject of the development – a hardware-software complex that ensures the operation of the face-tracking system using computer vision algorithms and machine learning techniques.

Purpose of the development – to create an intelligent face-tracking system that provides high accuracy and speed in real-time face detection, stable operation of the hardware components, integration with a computer vision module, and the ability to analyze and store the obtained data for further use.

The first chapter of the bachelor's qualification work explores all theoretical aspects of computer vision and face-tracking systems, as well as existing technical solutions in this field.

The second chapter presents the implementation of the system's hardware components, including the selection of specific technical elements.

The third chapter describes the algorithm and implementation of the system's software, which consists of two parts: software for the Arduino board and software for the system interface.

The final chapter presents the testing of the prototype system and provides results that demonstrate the operability of the designed system.

## ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів ..	8
Вступ .....	9
1 Системи відстеження обличчя .....	11
1.1 Актуальність та значення систем відстеження обличчя.....	11
1.2 Теоретичні основи технології відстеження обличчя .....	14
1.3 Огляд основних методів виявлення та відстеження обличчя.....	18
1.3.1 Алгоритми Haar Cascade.....	18
1.3.2 Нейронні мережі та глибоке навчання (Convolutional Neural Networks, CNN).....	20
1.3.3 Використання методів машинного навчання .....	22
1.4 Особливості роботи з відео- та зображеннями у реальному часі	24
1.5 Технічне завдання на проектування .....	26
2 Розробка апаратної платформи інтелектуальної системи відстеження обличчя .....	28
2.1 Огляд апаратного забезпечення для систем відстеження обличчя	28
2.1.1 Камера .....	28
2.1.2 Датчики .....	31
2.1.3 Процесори .....	34
2.2 Реалізація апаратної платформи системи відстеження обличчя..	36
3 Розробка програмного забезпечення .....	43
3.1 Структура програмного забезпечення та вибір мови програмування .....	43

3.2 Реалізація програмного забезпечення для контролера плати Arduino .....	46
3.3 Реалізація програмного забезпечення для комп'ютера .....	48
4 Дослідна експлуатація макетного зразка .....	53
4.1 Опис макетного зразка та методики випробувань .....	53
4.2 Результати моделювання роботи тестового зразка .....	55
Висновки .....	58
Перелік джерел посилання .....	60
Додаток А .....	62
Додаток Б .....	71
Додаток В .....	72
Додаток Г .....	73

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

AI (Artificial Intelligence) – штучний інтелект.

ML (Machine Learning) – машинне навчання.

CV (Computer Vision) – комп'ютерний зір.

CNN (Convolutional Neural Network) – згортова нейронна мережа.

FPS (Frames Per Second) – кількість кадрів за секунду.

IoU (Intersection over Union) – метрика, що використовується для оцінки точності виявлення об'єктів.

RGB (Red, Green, Blue) – кольорова модель, що базується на трьох основних кольорах.

Haar Cascades – алгоритм для виявлення обличчя на основі каскадних класифікаторів.

OpenCV (Open Source Computer Vision Library) – відкрита бібліотека для комп'ютерного зору.

UART (Universal Asynchronous Receiver-Transmitter) – універсальний асинхронний приймач-передавач.

FPS (Face Positioning System) – система позиціонування обличчя.

Bounding Box – обмежувальний прямокутник, що використовується для позначення області об'єкта на зображенні.

TensorFlow – програмна бібліотека для машинного навчання.

dlib – бібліотека для аналізу зображень і машинного навчання.

Edge Detection – метод виявлення контурів об'єктів на зображенні.

YOLO (You Only Look Once) – алгоритм для швидкого виявлення об'єктів у реальному часі.

Face Embeddings – векторне представлення обличчя для подальшого аналізу.

## ВСТУП

В останні роки інтелектуальні системи набувають все більшого значення у різних галузях, включаючи безпеку, розваги та медицину. Однією з найактуальніших тем у цій сфері є технології відстеження обличчя, які дозволяють виявляти, аналізувати та взаємодіяти з людським обличчям у реальному часі. Відстеження обличчя використовується у системах безпеки для ідентифікації особистості, у пристроях із доповненою реальністю, а також у медичних додатках, де воно може бути корисним для моніторингу пацієнтів або діагностування емоційних станів.

Метою цієї кваліфікаційної роботи є розробка інтелектуальної системи відстеження обличчя, яка базується на алгоритмах комп'ютерного зору та машинного навчання. Основне завдання системи полягає в тому, щоб виявляти обличчя в кадрі, відстежувати його положення та переміщення в реальному часі, а також реагувати на зміни в положенні обличчя. Для реалізації проєкту використовуються сучасні методи машинного навчання, такі як алгоритми глибокого навчання (зокрема, нейронні мережі), а також бібліотеки комп'ютерного зору OpenCV, які надають широкий спектр інструментів для обробки зображень та відео.

Розробка інтелектуальних систем на основі відстеження обличчя має численні практичні застосування. Наприклад, такі системи можуть бути використані в системах контролю доступу для ідентифікації користувачів, у маркетингових дослідженнях для аналізу емоцій покупців, а також у розважальних додатках для створення інтерфейсів, що реагують на вирази обличчя.

У роботі будуть детально розглянуті теоретичні основи технологій відстеження обличчя, проаналізовані сучасні методи та інструменти для розробки таких систем, а також представлено прототип інтелектуальної

системи, що демонструє її можливості. Це дозволить оцінити ефективність розробленої системи та її перспективи у різних галузях.

Таким чином, об'єктом розробки є програмно-технічний комплекс інтелектуальної системи відстеження обличчя. А мета розробки – створення апаратної платформи та програмного забезпечення, що реалізує систему «face tracking».

## 1 СИСТЕМИ ВІДСТЕЖЕННЯ ОБЛИЧЧЯ

### 1.1 Актуальність та значення систем відстеження обличчя

Системи відстеження обличчя набувають широкого поширення у сучасному технологічному світі через зростаючу потребу в автоматизації процесів взаємодії між людиною та машиною, безпеки, а також персоналізації користувацьких досвідів. Основними причинами їх актуальності є:

- Підвищення безпеки та ідентифікації. Відстеження обличчя широко застосовується в системах контролю доступу, відеоспостереження, розпізнавання осіб у громадських місцях та інфраструктурі високого ризику (аеропорти, урядові будівлі). Інтелектуальні системи здатні швидко ідентифікувати підозрілих осіб та оперативно реагувати на потенційні загрози, підвищуючи рівень безпеки у суспільстві.

- Персоналізація та взаємодія з користувачем. В маркетингу та роздрібній торгівлі системи відстеження обличчя використовуються для аналізу реакцій клієнтів, персоналізації послуг та створення індивідуальних пропозицій. Наприклад, інтерактивні рекламні системи можуть змінювати вміст на основі емоцій та демографічних характеристик користувача.

- Медичні застосування. Системи відстеження обличчя використовуються у медицині для відстеження емоцій пацієнтів, вивчення стану здоров'я на основі виразів обличчя, моніторингу за пацієнтами у відділеннях інтенсивної терапії. Відстеження мікровиразів дозволяє визначати психологічний стан людини, що має велике значення у психіатрії та неврології.

- Зростання попиту на безконтактні технології. В умовах пандемії COVID-19 системи, які дозволяють зменшити необхідність фізичного

контакту, стали особливо популярними. Відстеження обличчя використовується для автоматичного входу в будівлі без використання карток чи паролів, що дозволяє уникати зайвих дотиків до поверхонь.

– Розвиток штучного інтелекту та машинного навчання. Нові досягнення в області нейронних мереж та алгоритмів машинного навчання дозволили значно підвищити точність та швидкість роботи систем відстеження обличчя. Це дозволяє системам функціонувати у реальному часі навіть на відносно простому обладнанні, що робить технологію доступнішою для масового використання.

Технології відстеження обличчя мають довгу історію, яка бере свій початок ще в середині 20 століття, коли дослідники почали розробляти перші методи автоматичного аналізу обличчя [1]. Основою ранніх систем була робота з геометричними ознаками обличчя, такими як відстань між очима, формою носа, розміщенням рота тощо.

Перші спроби (1960-1980-ті роки) [1]. У 1960-х роках дослідження в області комп'ютерного бачення розпочались з ручного аналізу фотографій. Одним з перших відомих алгоритмів відстеження обличчя був алгоритм, розроблений Вуді Бледсо і його колегами, що намагався автоматизувати процес ідентифікації обличчя за допомогою певних геометричних правил.

Фазовий розвиток (1990-ті роки) [1]. У 1990-х роках з'явилися методи машинного навчання, які зробили значний прорив у цій галузі. Одним з ключових кроків стало впровадження алгоритму Eigenfaces, що використовував аналіз головних компонент (РСА) для розпізнавання обличчя. Цей метод поклав основу багатьом сучасним технологіям, оскільки він дозволяв зменшувати обсяг даних, що потрібно було аналізувати, і зберігати ключові ознаки обличчя.

Бум нейронних мереж (2000-ті роки) [1]. На початку 2000-х нейронні мережі почали займати провідне місце в розробці технологій відстеження обличчя. Конволюційні нейронні мережі (CNN) стали основним інструментом, що дозволив значно покращити точність та швидкість обробки зображень.

Особливо важливим стало використання великих наборів даних для навчання мереж, таких як LFW (Labeled Faces in the Wild), що дало можливість масштабувати системи розпізнавання.

Сучасний етап (2010-ті - сьогодні) [1]. Останніми роками технології глибокого навчання зробили можливим впровадження реальних систем розпізнавання обличь, які працюють у реальному часі. Вони використовують глибинні нейронні мережі, що забезпечують надвисоку точність навіть у складних умовах, таких як різке освітлення або різні позиції обличчя.

Сучасні системи відстеження обличчя знайшли широке застосування у багатьох сферах завдяки значному покращенню точності та продуктивності алгоритмів.

– **Безпека.** Системи відстеження обличь широко застосовуються у сферах безпеки та правопорядку. Вони використовуються для моніторингу громадських місць, автоматичної ідентифікації осіб у базах даних, контролю доступу до важливих об'єктів, а також у боротьбі з тероризмом. Наприклад, на стадіонах, в аеропортах та урядових будівлях застосовуються системи, що дозволяють швидко виявити підозрілих осіб.

– **Маркетинг та роздрібна торгівля.** В маркетингових дослідженнях технології розпізнавання обличь застосовуються для відстеження реакції клієнтів на різні рекламні кампанії та продукти. Вони дозволяють аналізувати емоції, вік, стать та інші демографічні характеристики, щоб забезпечити персоналізований досвід для покупців. Магазини використовують ці системи для вивчення поведінки клієнтів та створення цільових рекламних кампаній.

– **Медицина.** У медицині технології відстеження обличь застосовуються для моніторингу пацієнтів, аналізу емоційних реакцій, діагностики неврологічних та психіатричних захворювань. Наприклад, вони допомагають виявляти мікровирази, що можуть свідчити про певні емоційні або фізичні стани, і використовуються для створення безконтактних систем для моніторингу стану здоров'я пацієнтів.

– Соціальні мережі та мобільні додатки. Відстеження обличчя активно використовується в соціальних мережах для автоматичного позначення людей на фотографіях, фільтрації зображень та відео, а також створення додатків з ефектами доповненої реальності. Популярні програми, такі як Snapchat або Instagram, використовують алгоритми відстеження обличчя для накладання фільтрів в реальному часі.

Таким чином, системи відстеження обличчя є важливим кроком у розвитку інтелектуальних технологій, що забезпечують більш високу безпеку, покращують користувацький досвід та відкривають нові можливості для інтеграції людини з технологіями у різних сферах діяльності.

## 1.2 Теоретичні основи технології відстеження обличчя

Інтелектуальні системи відстеження обличчя працюють на основі алгоритмів комп'ютерного зору та машинного навчання, що дозволяють автоматично виявляти, відстежувати та аналізувати обличчя на зображеннях або відео в реальному часі. Основні етапи роботи таких систем можна розділити на кілька ключових процесів: детекція обличчя, екстракція ознак, відстеження руху та ідентифікація обличчя.

### 1. Детекція обличчя (Face Detection)

Детекція обличчя є першим етапом в роботі системи. Мета цього процесу — виявити обличчя на зображенні або у відеопотоці. Для цього використовуються різні алгоритми, найбільш популярним з яких є каскад Хаара, а також сучасні методи, засновані на конволюційних нейронних мережах (CNN).

Принцип роботи каскаду Хаара полягає в тому, що система використовує попередньо навчені моделі, які здатні виявляти ознаки обличчя, такі як очі, ніс та рот[2]. Завдяки великій кількості навчальних

зображень алгоритм може знаходити обличчя навіть у складних умовах, наприклад при зміні освітлення або положення голови.



Рисунок 1.1 – Детектування обличчя на основі каскаду Хаара

На рисунку 1.1 показано, як система сканує зображення та виявляє обличчя, використовуючи попередньо навчені шаблони.

## 2. Екстракція ознак (Feature Extraction)

Після виявлення обличчя система переходить до екстракції ознак — процесу виділення унікальних характеристик обличчя для подальшого аналізу. Для цього використовуються техніки, які дозволяють розпізнати ключові точки на обличчі, такі як:

- Відстань між очима;
- Форма носа і рота;
- Візерунок брів.

Ці характеристики називаються «ландмарками» обличчя і з їх допомогою створюється математична модель обличчя, яка може бути використана для його ідентифікації або відстеження рухів.

### 3. Відстеження обличчя (Face Tracking)

Відстеження обличчя є важливою функцією системи, яка дозволяє "слідувати" за обличчям, коли воно рухається в кадрі. Це особливо важливо для динамічних відеопотоків, де обличчя постійно змінює свою позицію. Для цього використовуються алгоритми, які зчитують зміни у положенні «ландмарок» та коригують положення обличчя у кадрі.

Методи відстеження можуть бути різними:

- Алгоритми оптичного потоку дозволяють відслідковувати переміщення пікселів на зображенні;
- Регіональні нейронні мережі (R-CNN) використовують багатоваріантні нейронні мережі для точного відстеження обличчя [3].

### 4. Ідентифікація обличчя (Face Recognition)

На основі даних, отриманих під час екстракції ознак, система може виконати ідентифікацію обличчя. Для цього використовується порівняння математичної моделі обличчя із зразками в базі даних. Алгоритми SVM (Support Vector Machine), DeepFace або FaceNet дозволяють зіставити обличчя з базою даних ідентифікацій, визначаючи конкретну людину [4].

### 5. Інтелектуальні компоненти

Окрім базового детектування та відстеження обличчя, інтелектуальні системи можуть використовувати додаткові алгоритми для аналізу емоцій, віку, статі або навіть визначення стану людини (наприклад, втома або пильність).

Приклад використання інтелектуальної системи відстеження обличчя в реальному часі наведено на рисунку 1.2.

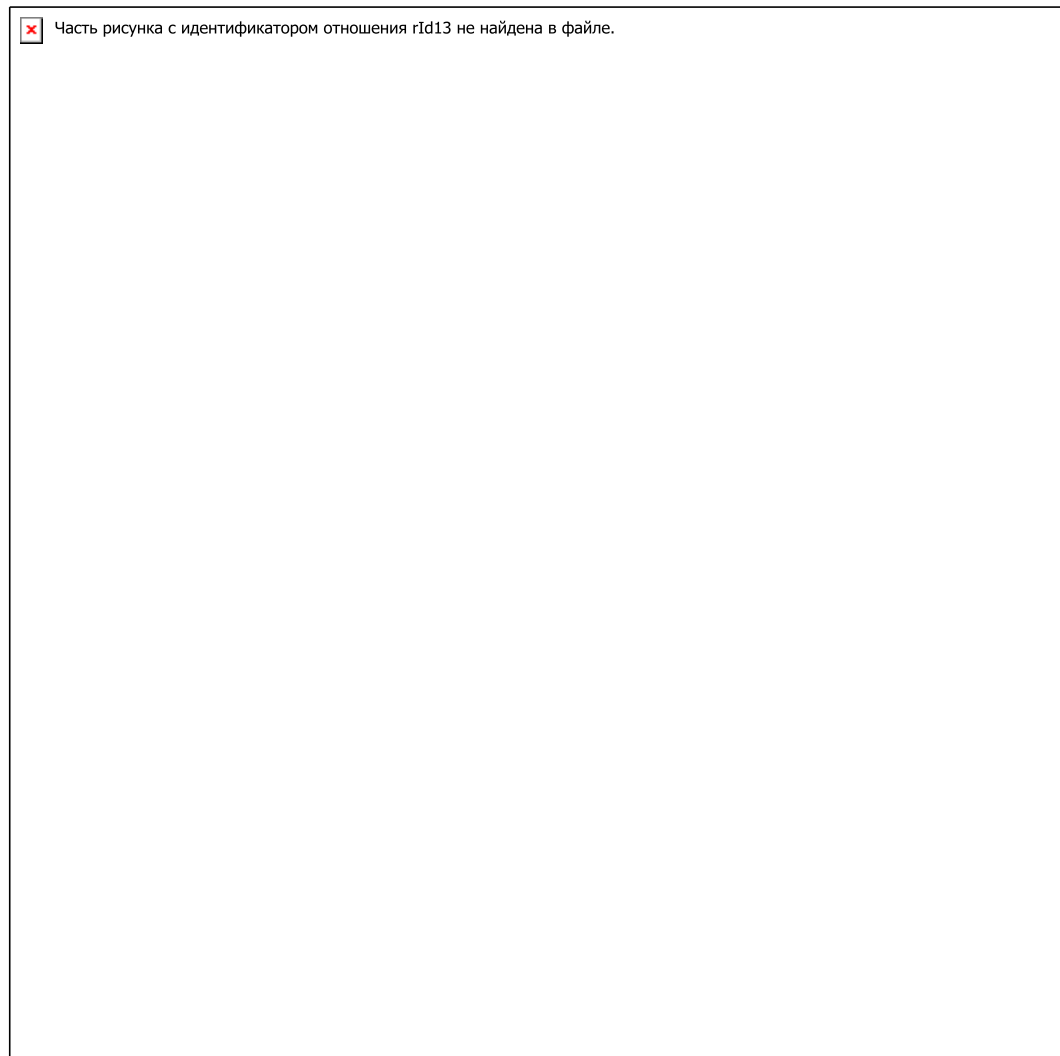


Рисунок 1.2 – Загальний принцип роботи системи відстеження обличчя

Ця схема ілюструє, як система спочатку виконує детектування, потім екстракцію ознак і відстеження руху обличчя.

Переваги інтелектуальних систем відстеження обличчя:

- Точність. Сучасні системи відстеження обличчя мають високу точність, здатні працювати у складних умовах освітлення та навіть з частково закритими обличчями.
- Швидкість обробки. Використання спеціалізованих алгоритмів, таких як CNN, дозволяє системам працювати в реальному часі.
- Масштабованість. Інтелектуальні системи можуть відстежувати кілька обличчя одночасно в одній сцені.

## 1.3 Огляд основних методів виявлення та відстеження обличчя

### 1.3.1 Алгоритми Haar Cascade

Алгоритми Haar Cascade є одним із найпоширеніших методів для детектування обличчя та об'єктів на зображеннях. Ця технологія використовується у багатьох сучасних системах відстеження обличчя завдяки її ефективності та швидкодії.

Haar Cascade — це метод, заснований на так званих каскадних класифікаторах, які використовують фільтри Хаара (рисунок 1.3). Фільтри Хаара — це спеціальні шаблони, які виявляють різні ознаки об'єктів на зображенні, наприклад, контури, межі між темними і світлими зонами, кути тощо. У контексті детектування обличчя вони можуть виявляти риси обличчя, як-от очі, ніс, рот або загальну форму обличчя [2].

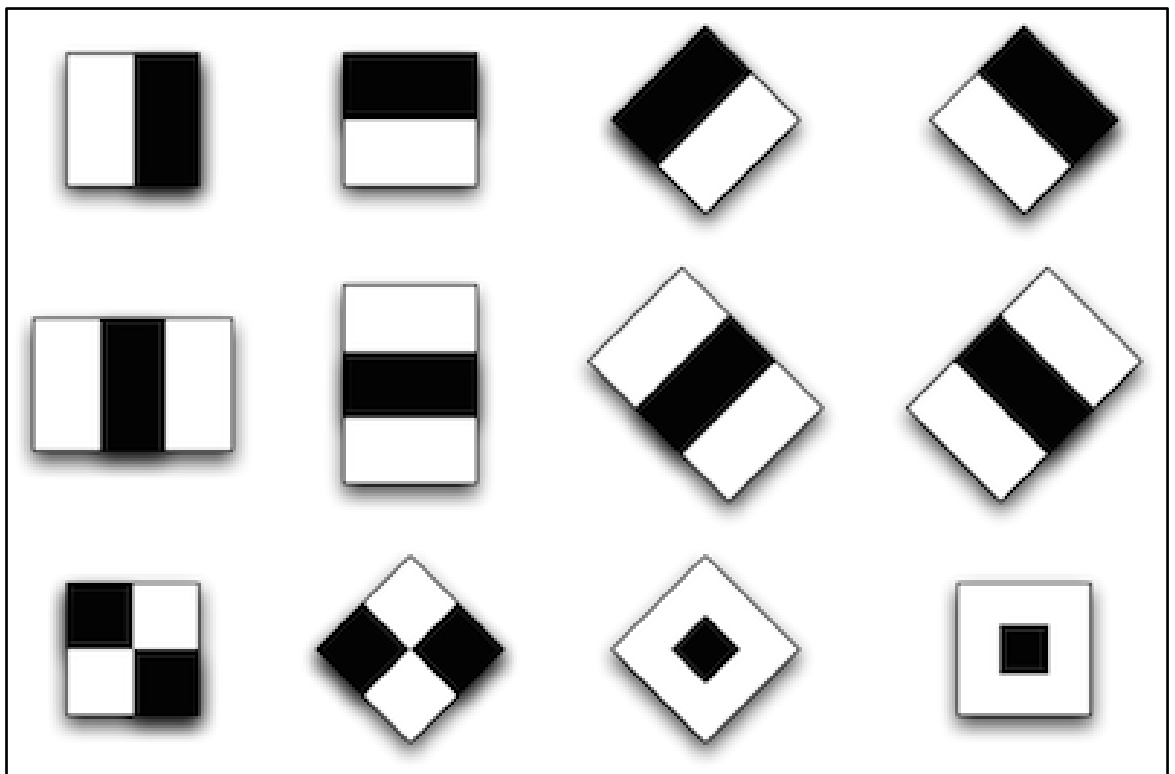


Рисунок 1.3 – Примітиви Хаара

Етапи роботи алгоритму:

1. Попередня обробка зображення:

Спочатку зображення перетворюється у градації сірого, щоб спростити обробку і знизити обчислювальні витрати. Після цього застосовуються кілька рівнів роздільної здатності зображення для аналізу об'єктів різного розміру.

2. Фільтри Хаара:

Алгоритм використовує набір фільтрів Хаара, які накладаються на зображення, щоб виявляти окремі ознаки обличчя. Це можуть бути шаблони для знаходження очей, носа чи губ. Кожен фільтр є простим прямокутним блоком, який визначає контраст між різними зонами зображення.

3. Каскадна структура:

Алгоритм використовує каскад класифікаторів, тобто ряд етапів перевірки, що дозволяють швидко відкидати зони зображення, де немає обличчя. Наприклад, якщо перший фільтр не виявив ознак обличчя, область не аналізується далі. Це дозволяє значно прискорити процес.

4. Навчання за допомогою алгоритму AdaBoost:

Процес навчання класифікаторів використовує метод AdaBoost, який вибирає найефективніші фільтри Хаара для кожного етапу каскаду. Під час навчання система обирає найкращі характеристики для розпізнавання обличчя і відсіювання зайвих ознак.

5. Виявлення обличчя:

Після проходження зображення через всі каскади алгоритм виділяє області, які можуть містити обличчя, і відзначає їх прямокутниками.

Переваги:

– Швидкість: Каскад Хаара працює дуже швидко, особливо для реального часу.

– Низькі обчислювальні витрати: Алгоритм не вимагає складних операцій, тому може використовуватися на вбудованих системах, таких як мікроконтролери.

Недоліки:

– Чутливість до умов освітлення: Haar Cascade може давати збої при сильних змінах освітлення або кута нахилу обличчя.

– Невисока точність порівняно з сучасними методами: Сучасні алгоритми, такі як CNN або методи глибокого навчання, мають вищу точність, але потребують більше ресурсів.

Алгоритм Haar Cascade широко застосовується в таких системах, як камери відеоспостереження, системи контролю доступу, та в мобільних додатках для розпізнавання обличчя.

### 1.3.2 Нейронні мережі та глибоке навчання (Convolutional Neural Networks, CNN)

Нейронні мережі є основою сучасних систем комп'ютерного зору, зокрема технологій відстеження та розпізнавання обличчя. Одним із найбільш потужних і ефективних підходів є Convolutional Neural Networks (CNN), або згорткові нейронні мережі [3]. Цей підхід став революційним завдяки своїй здатності виявляти складні патерни візуальних даних.

Основні принципи роботи CNN [5]:

1. Згортка (Convolution): Згорткові шари є основними блоками CNN. Вони використовують невеликі фільтри, які переміщуються (ковзають) по зображенню, щоб виділити локальні ознаки, як-от контури, крапки, текстури тощо. Це дозволяє мережі виявляти суттєві ознаки на різних рівнях абстракції — від простих країв до складних форм, як обличчя або його частини (очі, ніс, рот).

2. Шари активації: Після згортки до результату застосовуються нелінійні функції активації, як-от ReLU (Rectified Linear Unit), що дозволяє

моделі обробляти складні залежності між ознаками. Цей етап допомагає зробити модель здатною до навчання на нелінійних даних.

3. Pooling (Пулінг): Шари пулінгу (найчастіше max pooling) зменшують розмірність зображень, зберігаючи найбільш важливі ознаки. Це допомагає знизити кількість параметрів та обчислювальних ресурсів, необхідних для навчання моделі, водночас зберігаючи суттєву інформацію.

4. Повнозв'язні шари (Fully Connected Layers): Після кількох шарів згортки та пулінгу результат передається на повнозв'язні шари, які обробляють виділені ознаки і приймають рішення, наприклад, чи є на зображенні обличчя.

5. Класифікація: Останній етап — це класифікація, коли модель на основі отриманих ознак ухвалює рішення щодо об'єкта на зображенні (наприклад, чи є це обличчям, до якої особи воно належить тощо).

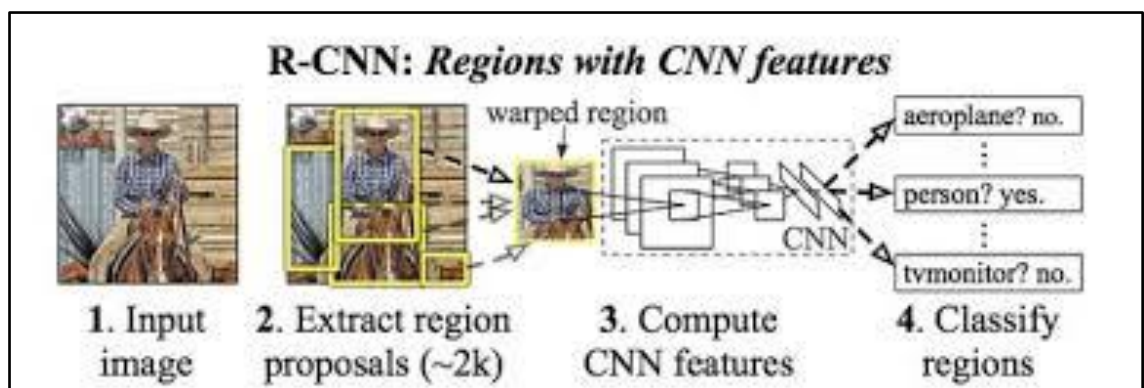


Рисунок 1.4 – Алгоритм роботи CNN

Переваги CNN [3]:

- Автоматичне виділення ознак: CNN не потребує ручного виділення ознак, як це було у класичних методах (наприклад, Haar Cascade). Мережа самостійно навчається виявляти патерни на різних рівнях абстракції.

- Гнучкість: CNN можуть бути адаптовані до різних задач — від детектування до класифікації та сегментації зображень.

- Висока точність: Завдяки глибокій архітектурі CNN, такі моделі забезпечують високу точність в задачах розпізнавання обличь.

Недоліки CNN [3]:

- Високі обчислювальні витрати: Глибокі CNN вимагають значних обчислювальних ресурсів та великих обсягів даних для навчання.

- Вимога до великих наборів даних: Для високої продуктивності CNN потрібні великі набори навчальних даних, що включають велику кількість зображень.

Сфери застосування CNN:

- Безпека: Відеоспостереження з автоматичним розпізнаванням облич на основі CNN використовується для контролю доступу та пошуку людей.

- Медицина: CNN застосовуються для аналізу медичних зображень, зокрема, для розпізнавання ознак захворювань на зображеннях шкіри, очей тощо.

- Маркетинг: У роздрібній торгівлі використання CNN допомагає відстежувати поведінку покупців, розпізнавати їхні емоції та надавати персоналізовані рекомендації.

### 1.3.3 Використання методів машинного навчання

Методи машинного навчання є основою сучасних інтелектуальних систем відстеження обличь, дозволяючи автоматично аналізувати та розпізнавати об'єкти на зображеннях або відеопотоках. Основне завдання таких методів полягає в тому, щоб система навчилася виявляти патерни, що дозволяють коректно розпізнавати обличчя на різних етапах аналізу. Це досягається шляхом тренування алгоритмів на великій кількості даних із прикладами обличь, фонових елементів і інших об'єктів.

Основні методи машинного навчання для відстеження обличь [5]:

– **Методи класифікації:** Однією з найбільш поширених задач машинного навчання в системах відстеження обличчя є класифікація. У цьому контексті система повинна вирішити, чи є наданий об'єкт обличчям чи ні. Це досягається шляхом тренування класифікаторів, як-от Support Vector Machines (SVM), або дерев рішень. Такі методи дозволяють системі вивчати особливості, які відрізняють обличчя від інших об'єктів.

– **Глибокі нейронні мережі (Deep Neural Networks, DNN):** Використання DNN у поєднанні з великими наборами даних дозволяє системам виявляти дуже складні патерни, включаючи деталі обличчя (як-от форма очей, носа, рота) і динамічні зміни (як-от міміка, ракурси обличчя). Найпоширеніші серед таких методів — Convolutional Neural Networks (CNN), що успішно використовуються для задач комп'ютерного зору завдяки їхній здатності обробляти зображення на різних рівнях абстракції.

– **Підхід Viola-Jones і каскади Хаара:** Один із ранніх методів машинного навчання для детектування обличчя базується на алгоритмі Viola-Jones і використанні каскадів Хаара. Цей підхід аналізує ознаки обличчя шляхом розбиття зображення на блоки та порівняння їх з відомими патернами. Система може детектувати обличчя навіть у реальному часі, завдяки своїй відносній простоті та швидкості.

– **Адаптивні алгоритми класифікації:** У сучасних системах часто використовують методи адаптивного навчання, які дозволяють системі підлаштовуватися до нових даних в реальному часі. Це важливо для систем відстеження обличчя, оскільки вони можуть зіткнутися з новими ракурсами, освітленням або змінами в зовнішності людей.

– **Підсилення і ансамблеві методи:** Методи, як-от Boosting (наприклад, AdaBoost), також часто використовуються для підвищення точності системи. Boosting дозволяє об'єднати кілька слабких класифікаторів у потужніший, що дає змогу точніше відстежувати обличчя в різноманітних умовах.

Етапи використання машинного навчання:

1. Збір і підготовка даних: Для побудови ефективної системи необхідно мати великий і різноманітний набір зображень. Дані мають охоплювати різні аспекти, як-от освітлення, положення обличь, якість зображень тощо. Часто використовуються такі набори даних, як LFW (Labeled Faces in the Wild), VGGFace, або навіть власні спеціалізовані набори.

2. Навчання моделі: Після збору даних використовується процес навчання моделі. Це етап, коли алгоритм машинного навчання використовує надані зображення для визначення ознак обличь і встановлення правил для їх подальшої ідентифікації.

3. Оцінка продуктивності: Оцінка моделей відбувається на тестових даних. Мета — перевірити, наскільки точно модель може ідентифікувати та відстежувати обличчя на нових зображеннях, яких вона не бачила під час навчання. Метрики продуктивності включають точність (accuracy), точність класифікації (precision), та інші.

4. Інференція та реальне використання: Після навчання модель інтегрується у реальні системи, що забезпечує відстеження обличь в реальному часі. Це може бути частиною великих систем безпеки, програм для відеоконференцій, або комерційних рішень у маркетингу та роздрібній торгівлі.

#### 1.4 Особливості роботи з відео- та зображеннями у реальному часі

Робота з відео- та зображеннями в реальному часі у контексті інтелектуальних систем відстеження обличь має свої особливості, які пов'язані з великою кількістю даних, що постійно надходять та потребують швидкої обробки. Для забезпечення ефективної роботи таких систем використовуються певні технічні та алгоритмічні рішення, які дозволяють виконувати аналіз кожного кадру в режимі реального часу.

Основні етапи роботи з відео та зображеннями в реальному часі [6]:

1. Захоплення відео- або фотоінформації. Камера або сенсор захоплює відеопотік чи серію зображень. Відеопотік ділиться на окремі кадри, які обробляються системою для детектування та відстеження обличч. Кожен кадр фактично є окремим зображенням, яке аналізується.

2. Попередня обробка зображень. Перед тим, як система почне шукати обличчя, часто проводиться попередня обробка зображень. Це може бути зменшення шуму, нормалізація освітлення, масштабування або навіть перетворення до чорно-білого формату для спрощення подальшої обробки.

3. Детектування обличчя. Основний етап – виявлення обличч на зображенні або відеокадрі. Найчастіше на цьому етапі використовуються алгоритми, як-от каскад Хаара або методи глибокого навчання (CNN). У реальному часі детектування повинно відбуватися з високою швидкістю та точністю.

4. Відстеження обличч. Після того, як обличчя було виявлено, система має відстежувати його рух у кадрі. Використовуються алгоритми, як-от KLT-трекінг (Kanade-Lucas-Tomasi) або оптичний потік. Важливо, щоб трекінг міг ефективно працювати навіть при зміні кута огляду, освітлення або часткових перекриттях обличчя.

5. Обробка в реальному часі. Однією з найважливіших характеристик є здатність системи обробляти відео в реальному часі з мінімальними затримками. Це забезпечується використанням оптимізованих алгоритмів, здатних аналізувати десятки або сотні кадрів за секунду.

6. Адаптація до нових даних. Реальні умови можуть змінюватися (освітлення, позиція обличчя, кількість людей в кадрі), тому система повинна мати здатність адаптуватися до таких змін, зберігаючи високу точність відстеження.

Основні проблеми при роботі з відео- та зображеннями в реальному часі:

- Висока обчислювальна потужність. Для обробки великої кількості кадрів потрібні швидкі процесори та спеціалізовані графічні чипи (GPU).
- Затримки в обробці. Чим швидше система обробляє дані, тим меншою буде затримка між захопленням кадру та його аналізом.
- Низька якість зображень або відео. Різке освітлення, рух об'єкта, шум можуть впливати на точність детектування та відстеження.

### 1.5 Технічне завдання на проектування

Метою кваліфікаційної роботи є розробка інтелектуальної системи відстеження обличчя в реальному часі з використанням алгоритмів комп'ютерного зору та машинного навчання. Основне завдання системи полягає в автоматичному виявленні, розпізнаванні та відстеженні облич на відеопотоці з камери або на окремих зображеннях.

Алгоритм роботи системи виглядає наступним чином:

1. Захоплення відеопотоку з камери. Після запуску система ініціалізує підключення до відеокамери або іншого джерела відео (наприклад, веб-камери), що є джерелом вхідного потоку даних.
2. Попередня обробка відео. На початковому етапі система проводить попередню обробку кадрів: зміна розміру зображення для зменшення обчислювального навантаження, перетворення кольорового зображення в чорно-біле для спрощення подальшої обробки.
3. Детектування обличчя. Далі використовуються алгоритми, такі як каскад Хаара або нейронні мережі, для виявлення обличчя на кожному окремому кадрі. Якщо обличчя було знайдено, система фіксує координати області кадру, де знаходиться обличчя.
4. Відстеження обличчя. Після виявлення система починає відстежувати положення обличчя на відео. Для цього можуть

використовуватися методи оптичного потоку або спеціалізовані алгоритми трекінгу, як-от KLT.

5. Реагування на дії користувача. Якщо користувач переміщується або обличчя змінює своє положення, система автоматично адаптується, коригуючи траєкторію відстеження.

6. Аналіз та виведення результатів. На останньому етапі система відображає результати в реальному часі на екрані, виділяючи області, де було знайдено обличчя.

7. Робота з помилками. У разі виявлення проблем (наприклад, відсутність обличчя або некоректне відстеження), система виводить повідомлення про помилки та надає можливість для їх виправлення.

Вимоги до системи:

- Простота використання. Інтерфейс повинен бути інтуїтивно зрозумілим, без необхідності глибоких технічних знань.
- Швидкість обробки. Система має забезпечувати високу швидкість обробки даних для роботи в реальному часі.
- Гнучкість. Можливість адаптації під різні умови (освітлення, різні обличчя, зміна положення камери).
- Надійність. Високий рівень точності та мінімум помилок у детектування та відстеженні.

Система має бути розроблена з врахуванням сучасних вимог до безпеки, її ефективність перевірятиметься під час експериментів в умовах реального часу.

## РОЗРОБКА АПАРАТНОЇ ПЛАТФОРМИ ІНТЕЛЕКТУАЛЬНОЇ СИСТЕМИ ВІДСТЕЖЕННЯ ОБЛИЧЧЯ

У першому розділі кваліфікаційної роботи було детально розглянуто предметну область, а саме теоретичні аспекти систем відстеження обличчя, а також існуючі у цій сфері технології.

У цьому розділі буде детально описано апаратну складову проєктованої системи та вибір необхідних для її реалізації інструментів.

### 2.1 Огляд апаратного забезпечення для систем відстеження обличчя

Будь-яка інтелектуальна система має апаратну складову, тобто комплекс пов'язаних між собою електронних компонентів, які реалізуються окремі функціональні частини системи.

Основними компонентами апаратної частини інтелектуальної системи відстеження обличчя є камери, датчики, підсистеми обробки зображень (спеціальні процесори) та додаткові механізми для захисту системи чи повороту камери тощо.

#### 2.1.1 Камера

Відеокамера — це пристрій, який перетворює оптичне зображення сцени на електричний відеосигнал, за допомогою чого система отримує початкову інформацію про об'єкт [7]. Якість відеосигналу залежить від технічних характеристик камери, яка є ключовим компонентом в системі відстеження. Камера складається зі світлочутливого елемента, генератора відеосигналу, підсилювача та системи автоматичного регулювання сигналу.

Більшість сучасних систем використовують ПЗЗ-матриці (прилади із зарядовим зв'язком) або КМОП-матриці (більш швидкі варіанти). Світлочутливий елемент накопичує заряди на основі кількості фотонів, що потрапляють на нього, формуючи зображення. Отримані заряди передаються для обробки та формування відеосигналу.

Камери поділяються на прості та професійні: перші мають вбудовані об'єктиви, а другі — змінні, з поліпшеними характеристиками. Відеосигнал може передаватися через радіохвилі, кабельні мережі або інтернет. У сучасних системах використовують цифрові носії для збереження відео.

Цифрові камери часто застосовуються в системах відеоспостереження та відеоконференцій, а в поєднанні з мікрофонами можуть записувати як відео, так і звук. Камери для професійної роботи також можуть записувати часовий код для синхронізації з іншими камерами. Сьогодні відеокамери інтегруються у смартфони, цифрові фотоапарати і навіть кінематографічні камери, що використовуються для зйомки фільмів.

Камери є основним джерелом зображень або відеопотоку, які використовуються для детектування та відстеження обличь. Є кілька видів камер, які застосовуються в таких системах [7]:

– Звичайні RGB-камери (рис.2.1). Це стандартні камери, що захоплюють кольорові зображення. Вони широко використовуються завдяки своїй доступності та можливості роботи з різними алгоритмами обробки зображень. Приклади таких камер: веб-камери, камери смартфонів або спеціалізовані USB-камери.

– Інфрачервоні камери (IR-камери) (рис.2.2). Вони використовуються для захоплення теплових зображень і можуть працювати у поганих умовах освітлення або навіть у повній темряві. Завдяки інфрачервоному діапазону ці камери забезпечують вищу точність при відстеженні обличь, оскільки не залежать від навколишнього освітлення.

– Глибокі камери (Depth Cameras) (рис.2.3). Ці камери вимірюють відстань до об'єктів у кадрі, що дозволяє створювати тривимірні зображення обличчя. Камери з підтримкою глибини, як-от Microsoft Kinect або Intel RealSense, використовуються для високоточних систем відстеження, зокрема у віртуальній реальності або системах ідентифікації.



Рисунок 2.1 – Веб-камера

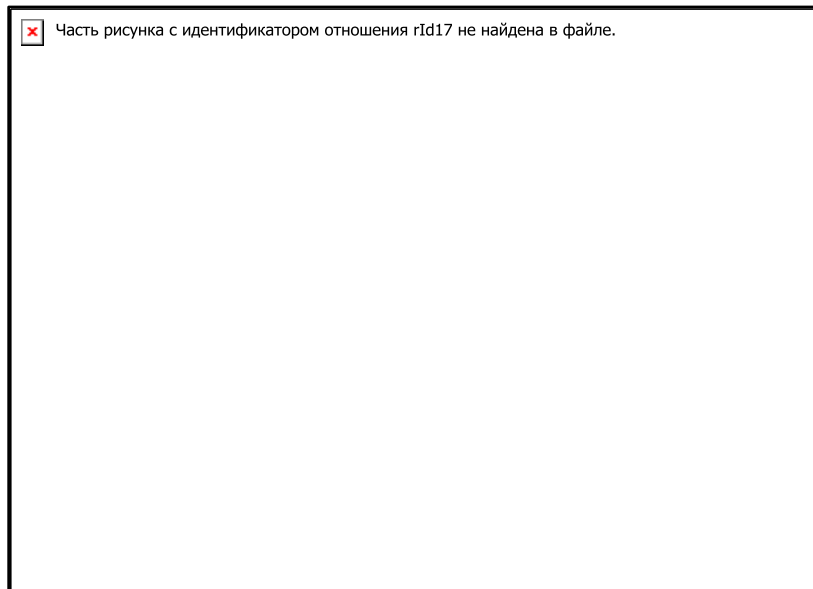


Рисунок 2.2 – Інфрачервона камера



Рисунок 2.3 – Камера з технологією глибини

### 2.1.2 Датчики

Датчики в системах відстеження обличчя є важливими компонентами, оскільки вони забезпечують збір первинної інформації про об'єкти в полі зору системи. Вони перетворюють фізичні параметри середовища (світло, інфрачервоні хвилі, глибину) у електричні сигнали, які система використовує

для аналізу й ідентифікації обличь [7]. Розглянемо детальніше, як працюють датчики, їх типи та які саме застосовуються в системах відстеження обличчя.

Датчики функціонують, перетворюючи певний вид енергії або інформації в електричний сигнал. Наприклад, оптичні датчики, які є основними в системах відстеження обличь, сприймають світло й перетворюють його на цифрову інформацію. Інші датчики, наприклад інфрачервоні або датчики глибини, можуть фіксувати температуру або відстань до об'єкта, що дозволяє системі "бачити" в умовах низької освітленості або навіть в темряві.

Типи датчиків, що використовуються у системах відстеження обличь:

- Оптичні датчики (відеокамери): Основний тип датчика, що використовується для відстеження обличчя. Вони перетворюють світло в цифрові дані, які потім обробляються для визначення присутності та позиції обличчя. Наприклад, відеокамери високої роздільної здатності можуть фіксувати найдрібніші деталі обличчя, що дозволяє системі розпізнавати риси та емоції.

- Інфрачервоні датчики: Ці датчики використовуються для фіксації теплового випромінювання від обличчя. Вони особливо корисні для розпізнавання в умовах слабкого освітлення або у темряві. Системи з інфрачервоними датчиками здатні розрізняти обличчя навіть при поганих погодних умовах або при недостатньому освітленні.

- Датчики глибини (3D-камери): Такі датчики вимірюють відстань до об'єкта. Вони можуть використовувати технології на зразок LiDAR або Time of Flight (ToF), щоб створити тривимірну карту обличчя. Цей підхід особливо важливий для точного розпізнавання обличчя у тривимірному просторі, що зменшує ризик помилок під час відстеження руху.

- Ультразвукові датчики: Такі датчики використовують звукові хвилі для визначення відстані до об'єкта. Хоча їх рідко використовують безпосередньо для розпізнавання обличчя, вони можуть бути корисними для

додаткових функцій, як-от фіксація руху або виявлення перешкод перед камерою.

– Гіроскопи та акселерометри: Вони використовуються для стабілізації зображення під час зйомки або для відстеження руху камери чи датчика в просторі. Це особливо важливо у мобільних пристроях або камерах, які знаходяться в русі.

В реальних умовах системи відстеження обличчя часто використовують комбінацію кількох типів датчиків. Наприклад, відеокамера може працювати в поєднанні з інфрачервоним датчиком для того, щоб "бачити" обличчя в різних умовах освітлення, або з датчиком глибини, щоб отримати 3D-модель обличчя. Такий підхід робить систему більш точною та надійною.

Приклади датчиків у існуючих системах:

– Apple Face ID (рис 2.4) використовує комбінацію інфрачервоних датчиків і 3D-сканера для точного відстеження обличчя, що дозволяє ідентифікувати користувача навіть у темряві.

– Microsoft Kinect використовує датчики глибини, які дозволяють камері відстежувати рухи та пози обличчя у тривимірному просторі.



Рисунок 2.4 – Інфрачервоні датчики у технології Apple Face ID

### 2.1.3 Процесори

Процесори є одними з ключових компонентів інтелектуальних систем відстеження обличь, оскільки вони відповідають за обробку даних, отриманих від камер та датчиків, і виконання складних алгоритмів для детектування обличчя в реальному часі. Вони виконують основні обчислювальні завдання та забезпечують швидкість і точність системи.

Для систем відстеження обличь використовують різні види процесорів, залежно від вимог до продуктивності, енергоефективності та складності алгоритмів:

- Центральні процесори (CPU): Відповідають за загальну обробку даних і виконання базових програмних інструкцій. CPU керує основними обчислювальними процесами, але для обробки відео та зображень можуть мати обмежену продуктивність. У багатьох системах відстеження обличь CPU працює у зв'язці з іншими спеціалізованими процесорами, такими як GPU або NPU.

- Графічні процесори (GPU): GPU мають здатність обробляти великі обсяги даних паралельно, що робить їх надзвичайно ефективними для роботи із зображеннями та відеопотоками, особливо в реальному часі. Вони використовуються для прискорення обчислень в алгоритмах машинного навчання, глибокого навчання та обробки великих матриць даних, які необхідні для розпізнавання обличчя. Найвідоміші GPU для таких завдань випускаються компаніями NVIDIA та AMD.

- Процесори нейронних обчислень (NPU): NPU спеціально розроблені для прискорення обробки задач, пов'язаних з машинним та глибоким навчанням. Вони оптимізовані для роботи з нейронними мережами та можуть ефективніше використовувати ресурси порівняно зі стандартними CPU та GPU. Процесори типу NPU використовуються в таких системах, як

Apple Face ID, де відбувається обробка обличчя у реальному часі з використанням штучного інтелекту.

– Цифрові сигнальні процесори (DSP): DSP оптимізовані для обробки сигналів у реальному часі, що включає обробку аудіо, відео та інших типів даних. Вони часто використовуються для роботи з потоками даних з датчиків у системах розпізнавання обличчя, оскільки можуть ефективно обробляти велику кількість інформації за короткий проміжок часу.

– ASIC (спеціалізовані інтегральні схеми): ASIC - це апаратні компоненти, розроблені спеціально для виконання певних завдань, таких як розпізнавання обличчя. Вони зазвичай вбудовані в пристрої, що вимагають високої продуктивності і ефективності. ASIC використовуються у великих системах відеоспостереження або біометричних пристроях, де необхідна швидкість і точність.

– Мікроконтролери: Вони є більш енергоефективними і часто використовуються у мобільних або портативних пристроях, які потребують меншої обчислювальної потужності. Такі процесори використовуються в базових інтелектуальних системах або там, де не потрібна обробка великих обсягів даних.

Процесори в системах відстеження обличчя забезпечують виконання кількох ключових функцій:

1. Обробка відеопотоків у реальному часі: Процесор повинен аналізувати відео, розпізнавати обличчя та відстежувати його переміщення, а також здійснювати обчислення в реальному часі.

2. Алгоритми розпізнавання: Процесори виконують складні алгоритми, такі як Haar Cascade або нейронні мережі (CNN), для виявлення та розпізнавання обличчя на зображеннях або відео.

3. Інтеграція з іншими системами: Процесори взаємодіють з іншими пристроями та компонентами, такими як камери, датчики та програмне

забезпечення для машинного навчання. Вони також можуть обробляти інформацію, отриману з кількох датчиків одночасно.

Приклади використання процесорів у системах розпізнавання обличь:

NVIDIA Jetson: Використовується в системах комп'ютерного зору і розпізнавання облич для прискорення обчислень за допомогою GPU.

Apple A-серія: Включає NPU, що використовується для розпізнавання обличчя в технології Face ID.

Qualcomm Snapdragon: Включає потужні CPU, GPU та DSP, які забезпечують обробку відеопотоків у смартфонах і системах відеоспостереження.

## 2.2 Реалізація апаратної платформи системи відстеження обличчя

Для реалізації апаратної частини системи відстеження обличчя необхідно підібрати електронні компоненти, які будуть покривати повний функціонал системи, а також відповідати усім вимогам, які висуваються до подібних систем.

При виборі технічних рішень для елементів структурної схеми важливо враховувати мінімальний набір функцій, який дозволяє забезпечити ефективну роботу системи, підтримуючи при цьому необхідний рівень надійності. До ключових критеріїв також належать низька вартість та оптимальне енергоспоживання, навіть при підключенні до мережі, а також зручність обслуговування. Крім того, обрані рішення мають відповідати вимогам щодо експлуатації та екологічним стандартам. Хоча ці фактори розташовуються за пріоритетом, для окремих компонентів їх важливість може змінюватися залежно від контексту використання.

Основна увага в роботі зосереджена на програмній частині системи, зокрема на алгоритмах та структурах даних, що використовуються для реалізації функцій. Тому апаратні компоненти обираються на основі вже

існуючих рішень, які за необхідності можуть бути адаптовані, а не розробляються з нуля.

Також важливо враховувати кліматичні умови середовища, де буде використовуватись система. Це означає, що технічні рішення повинні витримувати певні кліматичні умови, які поділяються на робочі показники (для нормальної роботи пристроїв) і граничні значення (для короточасних відхилень з можливістю повернення до номінальних параметрів після стабілізації умов).

Структурну схему апаратної платформи системи відстеження обличчя наведено на рисунку 2.5.

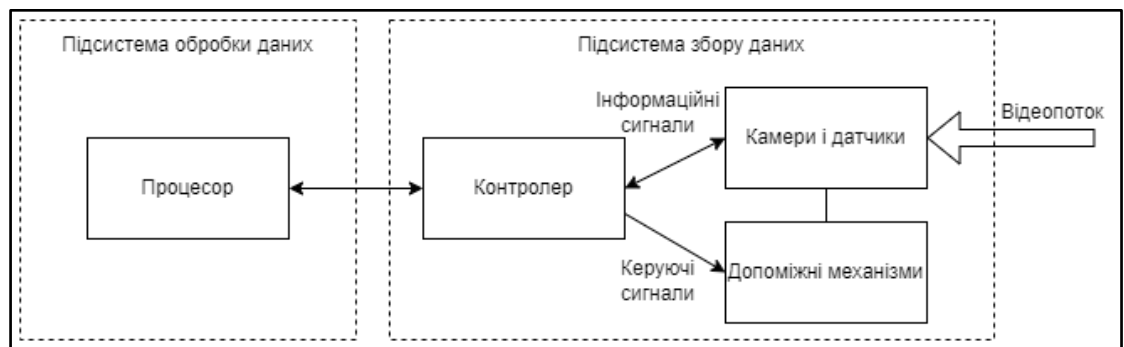


Рисунок 2.5 – Структурна схема проектованої системи

Основними компонентами у системі є камери і датчики, адже саме вони збирають інформацію про об'єкт. Для реалізації цієї частини структурної схеми можна використати або готове комбіноване технічне рішення, у вигляді камери з інфрачервоним датчиком та датчиком глибини (рис 2.6), що є більш дорогим варіантом, або по-окремо камеру та датчики, що є більш технічно складним варіантом, адже треба буде поєднувати всі компоненти.

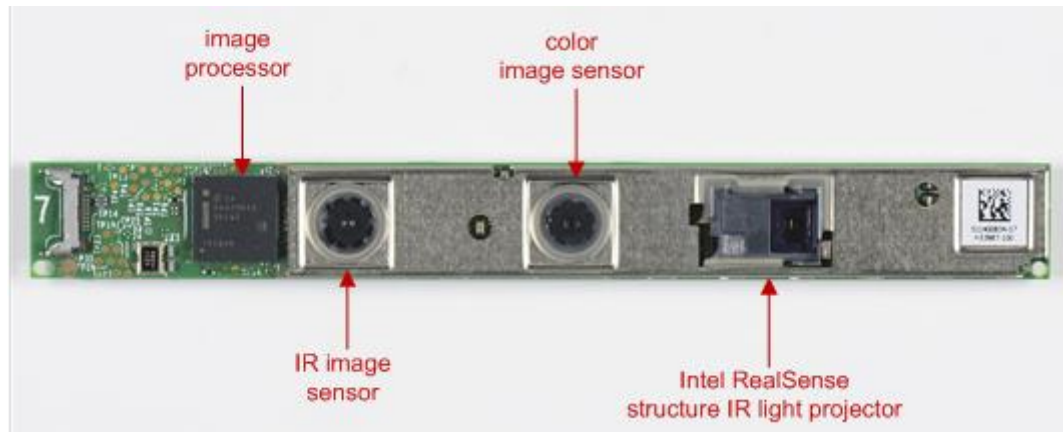


Рисунок 2.6 – Камера з інфрачервоним датчиком та датчиком глибини

Для спрощеного варіанту системи можна використати звичайну камеру, наприклад веб-камеру (рис 2.1), та більш складні алгоритми детектування та відстеження обличчя, що дозволить значно спростити апаратну складову системи.

Контролер системи виконує роль сполучника між камерами з датчиками та процесором. Основні його задачі – керувати камерою, збирати дані з датчиків та керувати допоміжними механізмами, наприклад поворотним механізмом камери.

На відміну від процесора, контролер не буде обробляти відеопотік, тому йому не потрібна велика обчислювальна потужність, проте невеликий форм-фактор та енергоефективність будуть грати суттєву роль в кінцевій реалізації пристрою системи. Таким чином, для реалізації контролера добре підійде мікроконтролер (рис 2.7).

Мікроконтролер — це компактний пристрій, що є мініатюрним комп'ютером, призначеним для керування електронними системами в режимі реального часу [8]. Він містить центральний процесор (CPU), оперативну пам'ять (RAM), постійну пам'ять (ROM/Flash), периферійні модулі для вводу-виводу (GPIO) та інші компоненти, такі як таймери, АЦП (аналогово-цифрові перетворювачі), послідовні інтерфейси (UART, SPI, I2C) тощо, зібрані на одному кристалі. Мікроконтролери використовуються в багатьох сферах, від

побутових пристроїв і промислових систем до робототехніки і систем автоматизації, для керування апаратними пристроями на основі отриманих даних та запрограмованих алгоритмів.

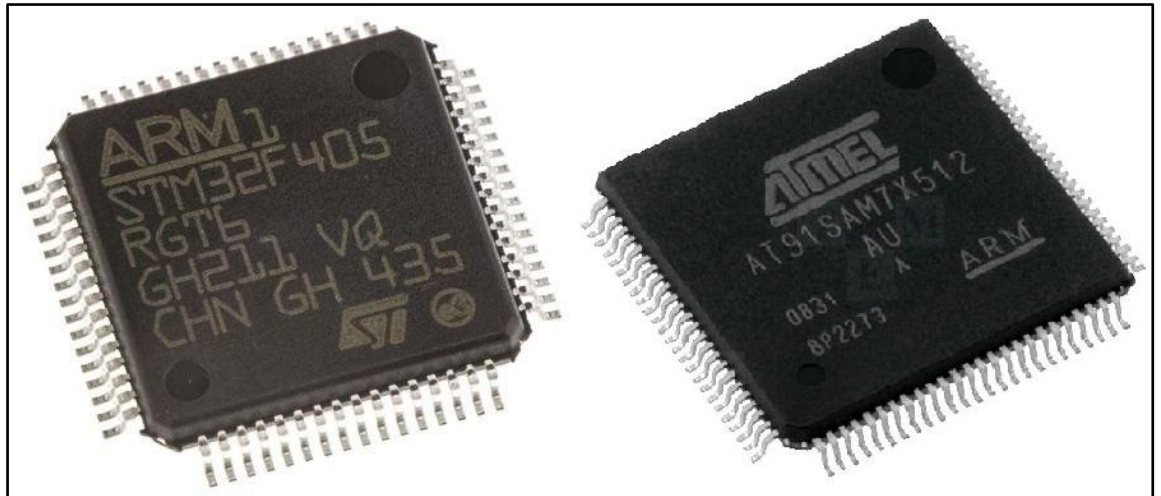


Рисунок 2.7 – Мікроконтролери

Найпоширенішим технічним рішенням на базі мікроконтролера при розробці електронних систем є платформа Arduino.

Arduino Nano — це мікроконтролерна плата, розроблена на основі мікроконтролера ATmega328 (Arduino Nano 3.0) або ATmega168 (Arduino Nano 2.x) [8]. Вона має компактний дизайн і часто використовується в проєктах, де потрібно мінімізувати розміри пристрою, але зберегти всі базові функції Arduino. Зовнішній вигляд плати та технічні характеристики представлено на рисунку 2.8 та таблиці 2.1 відповідно.

Таблиця 2.1 – Основні технічні характеристики Arduino Nano

Характеристика	Значення
Microcontroller	ATmega328 (8-bit AVR)
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (6 of which are PWM outputs)

Analog Input Pins	8
Flash Memory	32 KB (2 KB used by bootloader)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz
Dimensions	18 x 45 mm
Communication	UART, SPI, I2C
PWM Channels	6
ADC Channels	8 (10-bit resolution)
USB Interface	Mini-B USB for programming and power

Особливістю Arduino Nano є його невеликий розмір, завдяки чому вона часто використовується в компактних і мобільних системах. Вона може підключатися до комп'ютера через мікро-USB для програмування, а також підтримує зовнішнє живлення через роз'єм або пін.



Рисунок 2.8 – Arduino Nano

В якості допоміжного механізму буде використовуватися поворотний механізм на базі керованого мотора, для того, щоб камера могла повертатися при відстеженні обличчя.

Враховуючи високий рівень керованості при відносно невеликій вартості було прийнято рішення використовувати серводвигун, як основу поворотного механізму.

Серводвигун — це електромеханічний пристрій, що використовується для точного керування кутовим положенням, швидкістю або прискоренням обертання [9]. Він складається з двигуна, датчика положення (потенціометра або енкодера), системи зворотного зв'язку та електронного контролера, який отримує сигнал управління та перетворює його в рух.

Принцип роботи серводвигуна полягає в тому, що контролер відсилає сигнал, який порівнюється з реальним положенням валу двигуна. Якщо виникає різниця між бажаним і фактичним положенням, двигун коригує своє обертання, щоб звести цю різницю до мінімуму. Завдяки такій схемі зворотного зв'язку, серводвигун забезпечує високу точність керування.

Зовнішній вигляд та технічні характеристики серводвигуна наведено на рисунку 2.9 та у таблиці 2.2 відповідно.



Рисунок 2.9 - MicroServo MG90S

Таблиця 2.2 – Технічні характеристики MicroServo MG90S

Характеристика	Значення
Weight	14 г
Overall dimensions (L×W×H)	32×32×12 мм
Supply voltage	4,8...6 В
Torque	1,8...2,2 кг/см (при 4,8 В)
60° turn time	0,1с (при 4,8 В), 0,08с (при 6 В)
Working temperature	-30...+60°C
The length of the wires	24 см
Angle of rotation	180°
Gear material	метал
Shaft height	3.8 мм (+/- 0.1мм)

Найкращим готовим технічним рішенням для реалізації процесора є використання комп'ютера, адже для обробки відеопотоку у реальному часі потрібна не аби яка обчислювальна потужність, тому варіантів є два – використання комп'ютера, або розпаювати плату під процесор.

За для зменшення розмірів системи і збільшення її мобільності, доцільним буде використання комп'ютера з маленьким форм-фактором – нетбуки, нетопи та одноплатні комп'ютери. Приклад такого комп'ютера наведено на рисунку 2.10.



Рисунок 2.10 – Найменший комерційний нетбук GPD Pocket 3

### 3 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

У попередньому розділі було детально розглянуто апаратну частину системи відстеження обличь, а також надано аргументований вибір конкретних технічних рішень для реалізації апаратної частини системи. Проте окрім апаратної складової, ще треба реалізувати функціональну частину, а саме програмне забезпечення для обміну даними між камерою, датчиками, платою Arduino та процесором обробки даних. У цьому розділі представлено розробку такого програмного забезпечення.

Програмне забезпечення — це набір програм і супровідної документації, які необхідні для функціонування та управління системою обробки інформації. Це включає всі інструкції та інструменти, що допомагають налаштувати, використовувати та підтримувати роботу комп'ютерних програм [7].

#### 3.1 Структура програмного забезпечення та вибір мови програмування

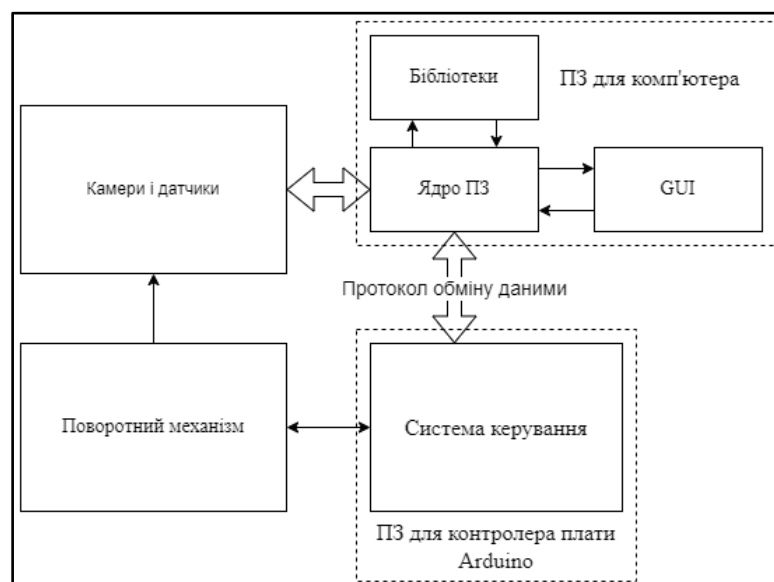


Рисунок 3.1 – Структура ПЗ проектованої системи

Як показано на рисунку 3.1, програмне забезпечення системи складається з двох основних частин: ПЗ для комп'ютера та ПЗ для контролера плати Arduino. Ці дві частини взаємодіють через програмний і фізичний інтерфейс Serial Port. Кожна з цих частин має свою структуру: ПЗ для комп'ютера містить графічний інтерфейс користувача (GUI), додаткові зовнішні модулі для коректної роботи та "ядро ПЗ", яке інтегрує всі елементи системи й забезпечує передачу даних. ПЗ для контролера Arduino включає підсистему управління поворотним механізмом, яка безпосередньо керує периферійними елементами і передає керуючі сигнали від комп'ютера до механізму.

Оскільки основна мета кваліфікаційної роботи — розробка програмно-технічного комплексу для керування периферійними пристроями на базі плати Arduino, програмна частина для контролера повинна бути розроблена спеціально для цієї платформи. Для цього використовується мова програмування C++ та середовище розробки Arduino IDE.

C++ — це компільована, статично типізована мова програмування загального призначення, яка підтримує такі парадигми як процедурне, об'єктно-орієнтоване і узагальнене програмування. Мова має багатий набір стандартних бібліотек, включаючи контейнери, алгоритми, введення-виведення, і багатопоточність[11].

Офіційне середовище розробки Arduino IDE забезпечує зручний інструмент для програмування мікроконтролерів на платі Arduino, дозволяючи ефективно компілювати і завантажувати код на плату для подальшого виконання [12]. Зовнішній вигляд Arduino IDE представлено на рисунку 3.2.

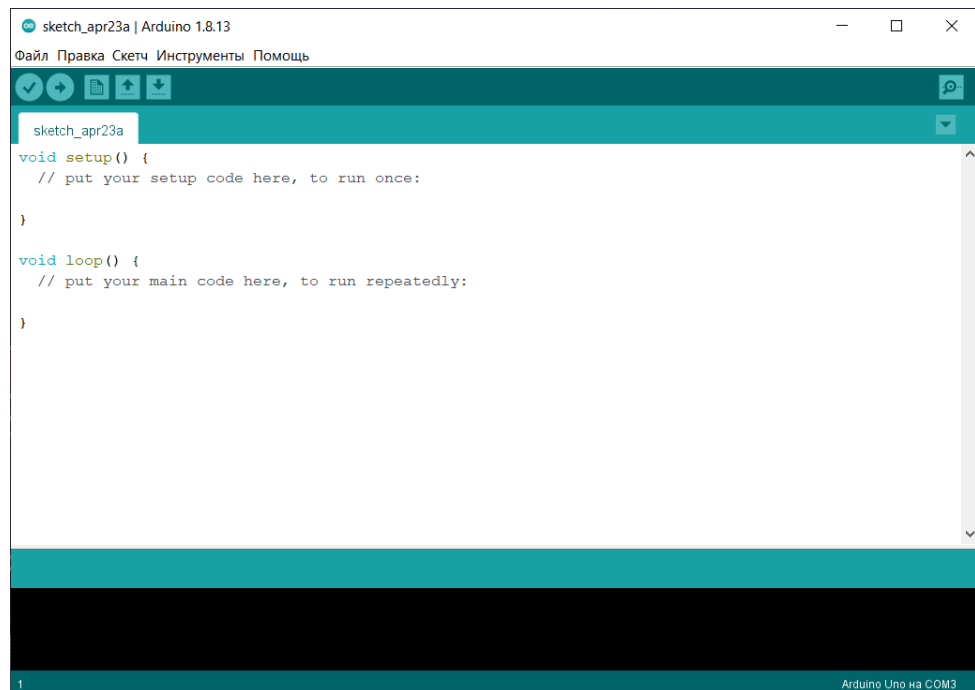


Рисунок 3.2 – ArduinoIDE

Для реалізації частини програмного забезпечення, що працює на комп'ютері і містить графічний інтерфейс користувача (GUI) та ядро системи, можливий вибір з широкого спектру мов програмування. Оскільки потрібно обробляти і відтворювати відеопотік, можна використати будь-яку сучасну мову високого рівня. Враховуючи, що Python є однією з найпопулярніших мов для обробки даних, його було обрано для цієї частини проекту. Python відповідає всім вимогам, необхідним для реалізації цього завдання.

Python — це інтерпретована мова програмування загального призначення з динамічною строгою типізацією та автоматичним управлінням пам'яттю. Вона орієнтована на підвищення продуктивності, зручне читання коду та підтримку багатьох платформ. У Python все є об'єктами, а його мінімалістичний синтаксис робить мову легкою для освоєння. Крім того, Python підтримує кілька парадигм програмування, серед яких: імперативне, процедурне, об'єктно-орієнтоване та функціональне програмування [15].

Завдяки популярності Python, існує велика кількість сторонніх модулів і бібліотек, які спрощують розробку складних додатків. Python має численне ком'юніті розробників та користувачів, що забезпечує наявність багатьох ресурсів для підтримки та розвитку. Найпопулярніші середовища розробки для Python включають: Python IDLE, PyCharm, Visual Studio Code, Sublime Text тощо [16]. Внутрішній інтерфейс Python IDLE представлено на рисунку 3.3.

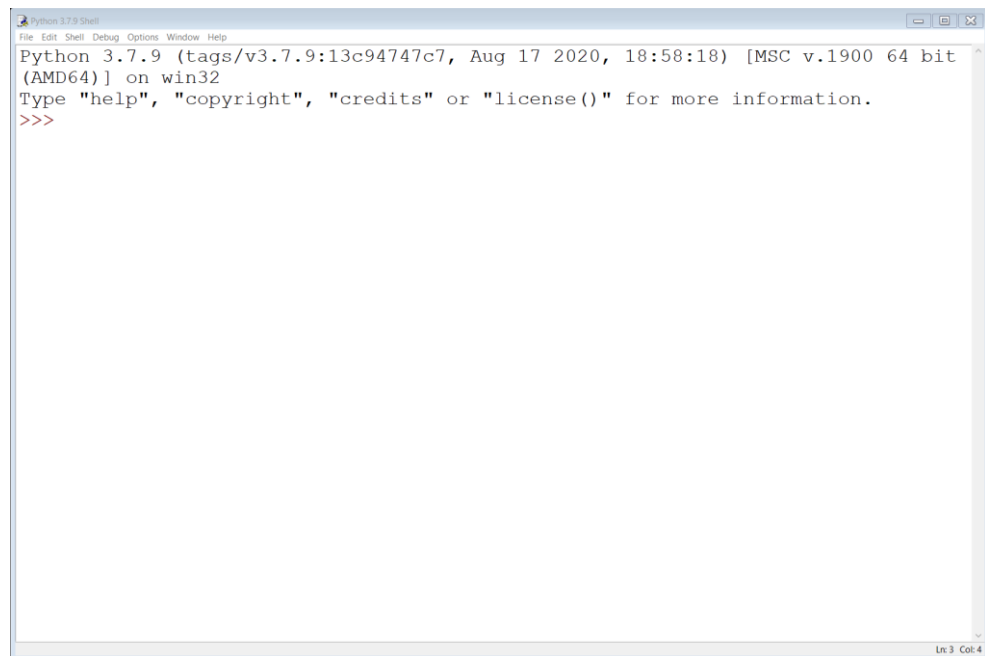


Рисунок 3.3 – Python IDLE

### 3.2 Реалізація програмного забезпечення для контролера плати Arduino

Програмне забезпечення для контролера плати Arduino має виконувати управління поворотним механізмом, який апаратно реалізовується за допомогою серводвигуна. Отже програмне забезпечення для контролера плати Arduino має керувати серводвигуном на основі сигналів з комп'ютера про те, де знаходиться обличчя користувача.

Нижче наведено декілька лістингів коду програмного забезпечення для контролеру плати Arduino. Повний код цієї частини ПЗ наведено у додатку Г.

### Лістинг 3.1 – Код функції setup()

```
void setup() {  
    Serial.begin(9600);  
    servoX.attach(9);  
    servoY.attach(10);  
    servoX.write(x);  
    servoY.write(y);  
    delay(1000);  
}
```

У лістингу 3.1 представлено код, який підв'язує серводвигуни до пінім на платі Arduino, а також встановлює початкові значення кутів повороту для серводвигунів у 90°.

### Лістинг 3.2 – Код функції loop()

```
void loop() {  
    if(Serial.available()){  
        input = Serial.read();  
        if(input == 'U'){  
            servoY.write(y+1);  
            y += 1;  
        }  
        else if(input == 'D'){  
            servoY.write(y-1);  
            y -= 1;  
        }  
    }  
    else{
```

```

    servoY.write(y);
}
if(input == 'L'){
servoX.write(x-1);
x -= 1;
} else if(input == 'R'){
servoX.write(x+1);
x += 1;
}
else{
servoX.write(x);
}
input = ""; }
}

```

У лістингу 3.2 представлено код, який зчитує керуючі сигнали з Serial порту та в залежності від того, який сигнал передано, робить поворот одного чи іншого серводвигуна, тим самим повертаючи камеру у двох площинах.

Таким чином, програмне забезпечення для контролера плати Arduino реалізує керування двома серводвигунами (один в площині X, інший - Y) на основі керуючих сигналів, що передаються від комп'ютера до поворотного механізму.

### 3.3 Реалізація програмного забезпечення для комп'ютера

У цьому підрозділі представлено детальний опис частини ПЗ системи, яка реалізує функціонал обробки відеопотоку та детектування обличчя на відео за для подальшого формування керуючого сигналу для контролера плати Arduino. Нижче наведено кілька лістингів коду цієї частини ПЗ, а повний код наведено у додатку Д.

До цієї частини програмного забезпечення відноситься «відкриття» послідовного порту та підключення плати до комп'ютера, а також отримання відеопотоку з камери та детектування обличчя. Для реалізації такого функціоналу системи необхідне використання додаткових модулів, а саме: cv2 – це бібліотека Python, яка надає інтерфейс до OpenCV (Open Source Computer Vision Library), що є набором інструментів для обробки зображень і комп'ютерного зору. З його допомогою можна виконувати різні операції з зображеннями та відео, такі як захоплення, обробка, аналіз, детектування об'єктів, розпізнавання обличчя, застосування фільтрів та багато іншого. Serial – стандартний модуль мови програмування Python, що реалізує обмін даними через послідовний порт. Time – модуль, який надає функції для роботи з часом. Він дозволяє отримувати поточний час, вимірювати час виконання програм та працювати із затримками. Модуль працює з часовими мітками (timestamps), що представляють кількість секунд з 1 січня 1970 року (епоха Unix).

Уся програма цієї частини ПЗ системи представляє собою нескінченний цикл, у якому реалізуються зчитування відеопотоку з камери; детектування обличчя; перевірка зони, де знаходиться обличчя і відправка керуючого сигналу на плату Arduino через послідовний порт.

### Лістинг 3.3 – Підключення до плати через послідовний порт

```
ard = serial.Serial("COM3", 9600)
```

У лістингу 3.9 представлено код для підключення до плати Arduino через послідовний порт «COM3», у разі підключення плати до іншого порту, можна змінити значення.

### Лістинг 3.4 – Підключення до камери та зчитування відеопотоку

```

vid = cv2.VideoCapture(1)
while True:
    _, frame = vid.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

```

У лістингу 3.4 наведено код, за допомогою якого можна підключитися до камери та зчитувати з неї відеопоток, паралельно конвертуючи картинку у чорно-білу гаму для спрощення подальшого детектування обличчя.

### Лістинг 3.5 – Детектування обличчя

```

face_cascade =
cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
faces = face_cascade.detectMultiScale(gray, minSize=(80,
80), minNeighbors=3)

```

У лістингу 3.5 наведено код, за допомогою якого можна детектувати обличчя з відеопотоку. Обличчя детектується по стандартному набору ознак записаних у файлі 'haarcascade\_frontalface\_default.xml'.

Лістинг 3.6 – Передача керуючих сигналів на плату Arduino в залежності від положення обличчя

```

for (x, y, w, h) in faces:
    cv2.rectangle(frame, (x, y), (x+w, y+h), (255, 0,
0), 2)

    Xpos = x+(w/2)
    Ypos = y+(h/2)
    if Xpos > 280:
        ard.write('L'.encode())
        time.sleep(0.01)
    elif Xpos < 360:

```

```
        ard.write('R'.encode())
        time.sleep(0.01)
    else:
        ard.write('S'.encode())
        time.sleep(0.01)
    if Ypos > 280:
        ard.write('D'.encode())
        time.sleep(0.01)
    elif Ypos < 200:
        ard.write('U'.encode())
        time.sleep(0.01)
    else:
        ard.write('S'.encode())
        time.sleep(0.01)
    break
```

У лістингу 3.6 представлено код, який обчислює положення обличчя відносно центру «фрейма» відеопотоку і в залежності від положення посилає різні керуючі сигнали для повороту механізму камери на плату Arduino через послідовний порт. Це продовжуватиметься до тих пір, доки обличчя не буде посередині «фрейму».

В цілому, принцип роботи програмного забезпечення для комп'ютера наступний: після запуску програми іде підключення до плати Arduino через Serial інтерфейс, паралельно з цим підключається камера і на екран виводиться головне вікно інтерфейсу з відеопотоком. Як тільки на відеопотоці з'являється обличчя, система реагує на нього і детектує полігон (квадрат) з обличчям. Далі система обчислює позицію полігону відносно центру «фрейма» і посилає сигнали на плату Arduino про те, куди треба повертати камеру. Відображення відеопотоку іде в реальному часі і користувач може чітко бачити, як система реагує на зміну положення обличчя.

У програмі також передбачено механізм обробки помилок. Наприклад, якщо плата Arduino не підключена, або втрачено зв'язок з камерою, або виникає інша помилка в системі, користувачу відображається відповідне попередження на екрані.

Програма працює у режимі безперервного циклу, тобто виконуватиметься постійно, доки користувач не зупинить її вручну або не вимкне живлення системи.

## 4 ДОСЛІДНА ЕКСПЛУАТАЦІЯ МАКЕТНОГО ЗРАЗКА

У попередніх розділах було докладно досліджено предметну область та теоретичну базу для розробки. Також вивчено методи і технології, що застосовуються для відстеження обличь, створено апаратну платформу системи та запропоновано конкретні технічні рішення для її реалізації. Крім того, було розроблено програмне забезпечення як для контролера плати Arduino, так і для комп'ютера. Щоб підтвердити працездатність і ефективність системи, необхідно провести експериментальне тестування макетного зразка, зібрати апаратну частину та запустити програмне забезпечення в умовах, максимально наближених до реальних умов експлуатації.

### 4.1 Опис макетного зразка та методики випробувань

Моделювання роботи макетного зразка проектованої системи необхідно проводити в умовах, наближених до умов експлуатації кінцевим користувачем, отже спочатку необхідно реалізувати апаратну складову, а далі запускати програмне забезпечення та проводити тестування.

Оскільки в рамках кваліфікаційної роботи основний акцент зроблено на розробці програмного забезпечення, доцільно використовувати готові технічні рішення для апаратної частини системи, як було зазначено у другому розділі. Всі електронні компоненти під'єднуються до плати Arduino за допомогою спеціальних проводів, а сама плата підключається до комп'ютера через USB-кабель через COM-порт. Щоб спростити процес збирання апаратної частини системи та уникнути плутанини з проводами, рекомендовано використання макетної плати «breadboard». Зовнішній вигляд

апаратної частини проектованої системи після підключення усіх компонентів представлено на рисунку 4.1.



Рисунок 4.1 – Реалізація апаратної складової системи

Після того, як апаратна частина системи зібрана, необхідно завантажити в мікроконтролер плати Arduino ту частину програмного коду, яка відповідає за роботу самої плати та її підключених периферійних елементів. Програмна частина, яка відповідає за обробку відео та розпізнавання обличчя, запускається на комп'ютері, до якого під'єднана плата Arduino. Після цього слід провести тестування всієї системи, яке включає кілька ключових етапів:

1. Запуск системи;
2. Перевірка підключення плати;
3. Перевірка підключення камери;
4. Перевірка відображення відеопотоку з камери;
5. Перевірка реакції системи на обличчя у кадрі;

6. Перевірка коректності детектування обличчя;
7. Перевірка роботи поворотного механізму камери;
8. Перевірка часу відгуку системи та таймаутів;
9. Перевірка відпрацювання нестандартних ситуацій, за яких виникають помилки у системі.

Результати тестування макетного зразка проектованої системи представлені у наступному підрозділі.

#### 4.2 Результати моделювання роботи тестового зразка

Першим ділом необхідно перевірити запуск системи, а саме роботу програмного забезпечення системи за умови виконання сценарію, який описано у ПЗ, а також за умови виникнення помилок. Наступним етапом є перевірка підключення плати до комп'ютера і підключення камери. Після цього перевіряємо відображення відеопотоку на екрані.

Результати запуску та перевірки відображення відеопотоку за умови підключення плати та камери до комп'ютера, а також повідомлення відсутності підключення, представлено на рисунках 4.2 та 4.3 відповідно.

Після запуску інтерфейсу та перевірки відображення відеопотоку, система буде у режимі очікування до поки не зафіксує обличчя у кадрі, тож наступне, що треба перевірити, це фіксація обличчя та реагування системи на зміну положення обличчя. Результат перевірки представлено на рисунку 4.4.



Рисунок 4.2 – Перевірка відображення відеопотоку

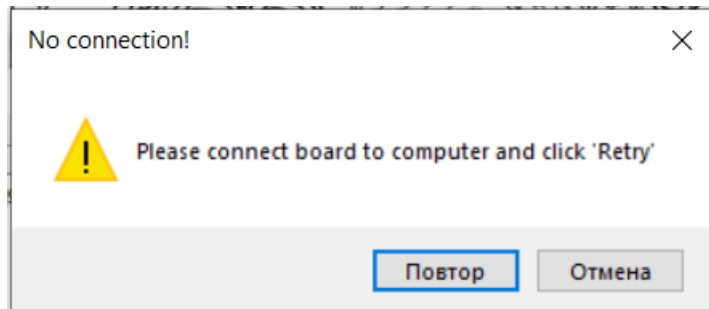
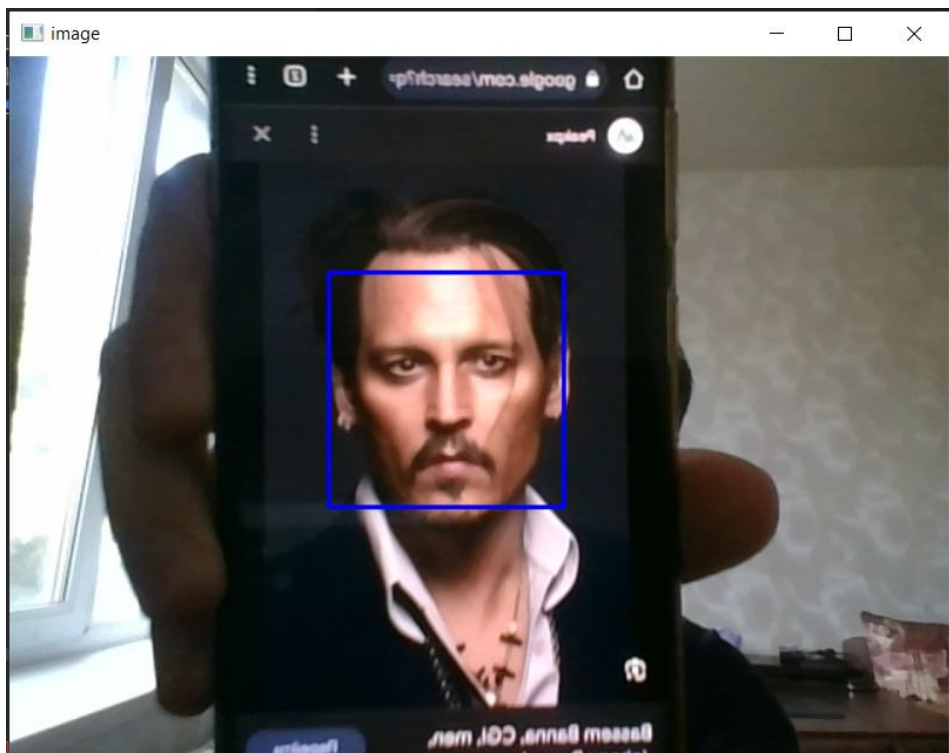


Рисунок 4.3 – Реакція системи на відсутність з'єднання з платою



#### Рисунок 4.4 – Фіксація обличчя

Нажаль, продемонструвати роботу поворотного механізму камери через рисунки не вийде, тому для цього було записано відео, на якому демонструється робота усієї системи і зокрема поворотного механізму. З цього відео видно, що поворотний механізм камери працює так, як це передбачено програмним забезпеченням.

Далі було проведено тестування часових затримок у системі, яке показало, що час відгуку системи знаходиться в межах норми і система реагує на зміну положення обличчя користувача у реальному часі з мінімальними затримками.

Отже за результатами тестування можна зробити наступний висновок – система працює відповідно до закладених у програмне забезпечення функцій. Вона пройшла всі етапи перевірки та продемонструвала свою працездатність і ефективність. Функціональні можливості системи виконуються без збоїв, а час реакції на дії користувача не перевищує задані таймаути. Система також правильно реагує на нестандартні ситуації, які викликають помилки в роботі. Враховуючи технічні особливості макетного стенду, можливо, що в реальних умовах продуктивність системи буде дещо іншою. Протягом тестування помилок або збоїв не зафіксовано.

## ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було створено інтелектуальну систему відстеження обличчя, яка поєднує апаратні та програмні рішення для автоматичного визначення і відслідковування положення обличчя в режимі реального часу. Основна мета розробки полягала в автоматизації процесу відстеження обличчя з використанням сучасних технологій комп'ютерного зору та алгоритмів машинного навчання.

Кваліфікаційна робота складається з кількох розділів, що відображають всі етапи розробки системи. У першому розділі проведено дослідження технологій, методів та інструментів, що використовуються в сучасних системах розпізнавання обличчя. На основі цього аналізу були визначені основні вимоги до системи, такі як точність і швидкість розпізнавання, а також стабільність роботи в умовах реального часу.

У другому розділі детально описано процес вибору апаратної платформи для системи. Вибір був зроблений на користь Arduino Nano як контролера для керування периферійними компонентами системи, а також камери і датчиків для захоплення зображення та отримання необхідних даних. Крім того, обговорено програмне забезпечення для взаємодії з апаратними компонентами.

Третій розділ присвячений розробці програмного забезпечення. Для аналізу зображень використано бібліотеку OpenCV (cv2), що забезпечує ефективну обробку відеопотоку та детектування обличь. Особлива увага приділена розробці алгоритмів для покращення точності та швидкості розпізнавання.

У заключному, четвертому розділі проведено тестування системи в різних умовах. Тестові випробування показали, що система здатна точно розпізнавати обличчя в різних ситуаціях і швидко реагувати на зміни

положення об'єкта в полі зору. Підсумкові результати підтвердили ефективність та працездатність розробленої системи.

Робота продемонструвала, що розроблена інтелектуальна система відстеження обличчя може бути успішно використана в різних сферах, таких як безпека, моніторинг або інтерактивні системи керування.


## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ


1. Система розпізнавання облич [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – Електронні данні. – Режим доступу:[https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0\\_%D1%80%D0%BE%D0%B7%D0%BF%D1%96%D0%B7%D0%BD%D0%B0%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F\\_%D0%BE%D0%B1%D0%BB%D0%B8%D1%87](https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D1%80%D0%BE%D0%B7%D0%BF%D1%96%D0%B7%D0%BD%D0%B0%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%BE%D0%B1%D0%BB%D0%B8%D1%87)
2. An Enhancement of Haar Cascade Algorithm Applied to Face Recognition for Gate Pass Security. / Antipona, C., Magsino, R. R., Dioses, R. M., & Mata, K. E. // *Problems of Information Technology*.2024.
3. Нейронні мережі [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – Електронні данні. – Режим доступу:[https://uk.wikipedia.org/wiki/Нейронні\\_мережі](https://uk.wikipedia.org/wiki/Нейронні_мережі)
4. Ідентифікація обличчя [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – Електронні данні. – Режим доступу:[https://uk.wikipedia.org/wiki/Ідентифікація\\_обличчя](https://uk.wikipedia.org/wiki/Ідентифікація_обличчя)
5. Deep Learning in Neural Networks: An Overview. / Schmidhuber, J. // *Neural Networks*, 61, 85–117.(2015)  
<https://doi.org/10.1016/j.neunet.2014.09.003>
6. Gonzalez, R. C., & Woods, R. E. (2017). *Digital Image Processing* (4th ed.). Upper Saddle River, NJ: Pearson. ISBN 978-0133356724.
7. Півняк Г. Г., Бусигін Б. С., Дівізінюк М. М. Тлумачний словник з інформатики. — Д.: Національний гірничий університет, 2010. — 392 с. — ISBN 978-966-350-306-0.
8. Мікроконтролер [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – Електронні данні. – Режим доступу:<https://uk.wikipedia.org/wiki/Мікроконтролер>

9. Arduino Nano [Електронний ресурс] // Arduino: офіційний сайт. – Електронні данні. – Режим доступу: <https://www.arduino.cc/>
10. Серводвигун [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – Електронні данні. – Режим доступу:<https://uk.wikipedia.org/wiki/Серводвигун>
11. **C++ Programming: Principles and Practice Using C++** / Bjarne Stroustrup. — Addison-Wesley Professional, 2014. — 1312 p. — ISBN 978-0321992789.
12. Elliot Williams. Make: AVR Programming: Learning to Write Software for Hardware. — Maker Media, Inc., 2014. — 472 p. — ISBN 978-1449355784.
13. Julien Bayle. C Programming for Arduino. — Packt Publishing, 2013. — 512 p. — ISBN 978-1849517584.
14. **Simon Monk. Programming Arduino: Getting Started with Sketches.** — McGraw-Hill Education, 2016. — 192 p. — ISBN 978-1259641633.
15. Python [Електронний ресурс] // Вікіпедія: вільна енциклопедія. – Електронні данні. – Режим доступу:<https://uk.wikipedia.org/wiki/Python>
16. Марк Лутц.[Programming Python](#) — 4 вид. — O'Reilly Media, 2010. — 1600 с. — [ISBN 978-0-596-15810-1](#).

## ДОДАТОК А

## Графічні матеріали (презентація) до кваліфікаційної роботи

	Харківський національний університет радіоелектроніки Кафедра АПОТ	
	Кваліфікаційна робота бакалавра	
<h2>Інтелектуальна система детекції обличчя</h2>		
Виконав: студент групи КІУКІ-21-9 <u>Роскошній В.Р.</u>		Керівник: ст. викладач кафедри АПОТ Мірошник А.М.
Харків 2025		

	<h2>Мета та задачі кваліфікаційної роботи</h2>	
	<p><b>Об'єкт розробки</b> – інтелектуальна система відстеження обличчя, призначена для автоматичного виявлення, ідентифікації та супроводження обличчя в реальному часі.</p> <p><b>Предмет розробки</b> – апаратно-програмний комплекс, що забезпечує функціонування системи відстеження обличчя з використанням алгоритмів комп'ютерного зору та машинного навчання.</p> <p><b>Мета розробки</b> – створення інтелектуальної системи відстеження обличчя, яка забезпечує високу точність і швидкість виявлення обличчя у реальному часі, стабільну роботу апаратної частини, інтеграцію з модулем комп'ютерного зору, а також можливість аналізу та збереження отриманих даних для подальшого використання.</p> <p><u>Для досягнення поставленої мети необхідно вирішити такі задачі:</u></p> <ol style="list-style-type: none"> <li>1. Розглянути предметну область та визначити проблематику;</li> <li>2. Реалізувати апаратну складову системи;</li> <li>3. Розробити програмне забезпечення системи;</li> <li>4. Реалізувати макетний зразок та провести тестування функціоналу розробленої системи.</li> </ol>	
2		



## СИСТЕМИ ВІДСТЕЖЕННЯ ОБЛИЧЧЯ

Системи відстеження обличчя набувають широкого поширення у сучасному технологічному світі через зростаючу потребу в автоматизації процесів взаємодії між людиною та машиною, безпеки, а також персоналізації користувацьких досвідів. Основними причинами їх актуальності є:

- Підвищення безпеки та ідентифікації.
- Персоналізація та взаємодія з користувачем.
- Медичні застосування.
- Зростання попиту на безконтактні технології.
- Розвиток штучного інтелекту та машинного навчання.

3



## Системи комп'ютерного зору

Інтелектуальні системи відстеження обличчя працюють на основі алгоритмів комп'ютерного зору та машинного навчання, що дозволяють автоматично виявляти, відстежувати та аналізувати обличчя на зображеннях або відео в реальному часі. Основні етапи роботи таких систем можна розділити на кілька ключових процесів: детекція обличчя, екстракція ознак, відстеження руху та ідентифікація обличчя.



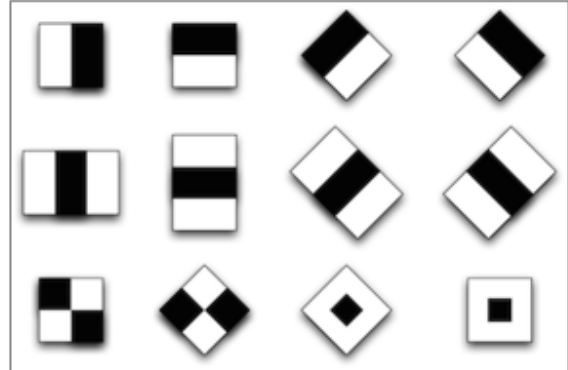
4



## Огляд основних методів виявлення та відстеження обличчя

Алгоритми Haar Cascade є одним із найпоширеніших методів для детектування обличчя та об'єктів на зображеннях. Ця технологія використовується у багатьох сучасних системах відстеження обличчя завдяки її ефективності та швидкодії.

Haar Cascade — це метод, заснований на так званих каскадних класифікаторах, які використовують фільтри Хаара. Фільтри Хаара — це спеціальні шаблони, які виявляють різні ознаки об'єктів на зображенні, наприклад, контури, межі між темними і світлими зонами, кути тощо. У контексті детектування обличчя вони можуть виявляти риси обличчя, як-от очі, ніс, рот або загальну форму обличчя.



5

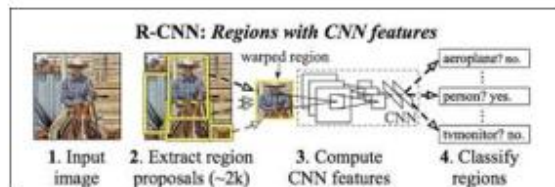


## Огляд основних методів виявлення та відстеження обличчя

Нейронні мережі є основою сучасних систем комп'ютерного зору, зокрема технологій відстеження та розпізнавання обличчя. Одним із найбільш потужних і ефективних підходів є Convolutional Neural Networks (CNN), або згорткові нейронні мережі. Цей підхід став революційним завдяки своїй здатності виявляти складні патерни візуальних даних.

Основні етапи роботи CNN:

- Згортка (Convolution);
- Шари активації;
- Pooling (Пулінг);
- Повноз'язні шари (Fully Connected Layers);
- Класифікація.



6



## Особливості роботи з відео- та зображеннями у реальному часі

Робота з відео- та зображеннями в реальному часі у контексті інтелектуальних систем відстеження обличчя має свої особливості, які пов'язані з великою кількістю даних, що постійно надходять та потребують швидкої обробки. Для забезпечення ефективної роботи таких систем використовуються певні технічні та алгоритмічні рішення, які дозволяють виконувати аналіз кожного кадру в режимі реального часу.

Основні етапи роботи з відео та зображеннями в реальному часі:

- Захоплення відео- або фотоінформації;
- Попередня обробка зображень;
- Детектування обличчя;
- Відстеження обличчя;
- Обробка в реальному часі;
- Адаптація до нових даних.

Основні проблеми при роботі з відео- та зображеннями в реальному часі:

- Висока обчислювальна потужність. Для обробки великої кількості кадрів потрібні швидкі процесори та спеціалізовані графічні чипи (GPU).
- Затримки в обробці. Чим швидше система обробляє дані, тим меншою буде затримка між захопленням кадру та його аналізом.
- Низька якість зображень або відео. Різне освітлення, рух об'єкта, шум можуть впливати на точність детектування та відстеження.

7



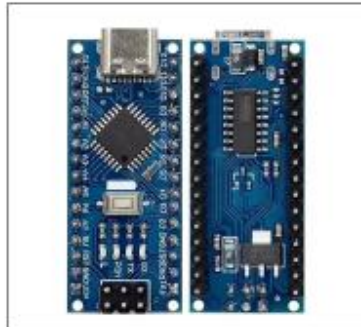
## Структурна схема апаратної платформи системи



8



## Технічні компоненти для реалізації апаратної складової системи



Arduino Nano

Характеристика	Значення
Microcontroller	ATmega328 (8-bit AVR)
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	14 (6 of which are PWM outputs)
Analog Input Pins	8
Flash Memory	32 KB (2 KB used by bootloader)
SRAM	2 KB
EEPROM	1 KB
Clock Speed	16 MHz
Dimensions	18 x 45 mm
Communication	UART, SPI, I2C
PWM Channels	6

8



## Технічні компоненти для реалізації апаратної складової системи



MicroServo MG90S

Характеристика	Значення
Weight	14 г
Overall dimensions (L×W×H)	32×32×12 мм
Supply voltage	4,8...6 В
Torque	1,8...2,2 кг/см (при 4,8 В)
60° turn time	0,1с (при 4,8 В), 0,08с (при 6 В)
Working temperature	-30...+60°C
The length of the wires	24 см
Angle of rotation	180°
Gear material	метал
Shaft height	3,8 мм (+/- 0,1мм)

10



## Елементна база апаратної складової системи



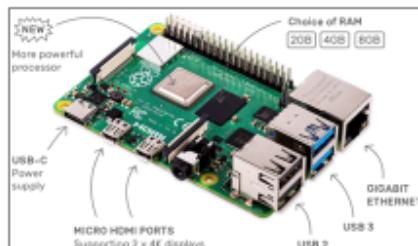
Веб-камера

Характеристика	Опис
Сенсор	1.3 МП
Максимальна роздільність відео	640 × 480 пікселів (VGA)
Частота кадрів	30 кадрів/сек
Мікрофон	Вбудований
Інтерфейс	USB
Довжина кабелю	1.82 м (71.7 дюймів)
Тип підключення	Plug & Play
Ширина	38 мм (1.5 дюймів)
Глибина	49 мм (1.93 дюймів)
Висота	50.5 мм (1.99 дюймів)

11



## Елементна база апаратної складової системи



Raspberry Pi 4

Характеристика	Значення
System on a Crystal (SoC)	Broadcom BCM2711 (CPU + GPU)
Processor	ARMv8 Cortex-A72 (1.5 ГГц)
Graphics processor	VideoCore VI GPU
RAM	2/4/8 ГБ SDRAM LPDDR4
ROM	карта пам'яті MicroSD
Interfaces	DSI, Serial, I <sup>2</sup> C, TWI, SPI
Ports	2 x USB 3.0; 2 x USB 2.0; 2 x micro-HDMI; 1 x LAN
Wireless technologies	Wi-Fi 802.11b/g/n; Bluetooth 5.0
Feeding	5 В, 3.0А через порт USB type C або GPIO
OS	Ubuntu, Debian, Fedora, Arch Linux, Gentoo, RISC OS, Android, Firefox OS, NetBSD, FreeBSD, Slackware, Tiny Core Linux, Windows 10 IoT
Dimensions	85.6 мм x 56.5 мм x 17 мм
Weight	45 г

12



## Структурна схема програмного забезпечення системи



13



## Розробка програмного забезпечення системи

### Лістинг коду функції setup()

```
void setup() {
  Serial.begin(9600);
  servoX.attach(9);
  servoY.attach(10);
  servoX.write(x);
  servoY.write(y);
  delay(1000);
}
```

### Лістинг коду функції loop()

```
void loop() {
  if(Serial.available()){
    input = Serial.read();
    if(input == 'U'){
      servoY.write(y+1);
      y += 1;
    }
    else if(input == 'D'){
      servoY.write(y-1);
      y -= 1;
    }
    else{
      servoY.write(y);
    }
  }
  if(input == 'L'){
    servoX.write(x-1);
    x -= 1;
  }
  else if(input == 'R'){
    servoX.write(x+1);
    x += 1;
  }
  else{
    servoX.write(x);
  }
  input = "";
}
```

14



## Розробка програмного забезпечення системи

### Лістинг коду для зчитування відеопотоку та детекції обличчя

```

vid = cv2.VideoCapture(1)
while True:
    _, frame = vid.read()
    gray = cv2.cvtColor(frame,
cv2.COLOR_BGR2GRAY)
    face_cascade =
cv2.CascadeClassifier('haarcascade_
frontalface_default.xml')
    faces =
    face_cascade.detectMultiScale(gray,
minSize=(80, 80), minNeighbors=3)

```

### Лістинг коду для передачі керуючих сигналів на плату Arduino в залежності від положення обличчя

```

for (x, y, w, h) in faces:
    cv2.rectangle(frame, (x, y),
(x+w, y+h), (255, 0, 0), 2)
    Xpos = x+(w/2)
    Ypos = y+(h/2)
    if Xpos > 280:
        ard.write('L'.encode())
        time.sleep(0.01)
    elif Xpos < 360:
        ard.write('R'.encode())
        time.sleep(0.01)
    else:
        ard.write('S'.encode())
        time.sleep(0.01)
    if Ypos > 280:
        ard.write('D'.encode())
        time.sleep(0.01)
    elif Ypos < 200:
        ard.write('U'.encode())
        time.sleep(0.01)
    else:
        ard.write('S'.encode())
        time.sleep(0.01)
        break

```

15



## Тестування макетного зразка проектованої системи

Реалізована інтелектуальна система детектування обличчя



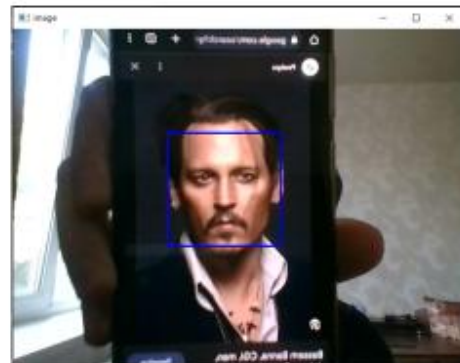
16



## Тестування макетного зразка проекрованої системи



Реакція системи на відсутність з'єднання з платою



Фіксація обличчя

17



## Висновки

У процесі виконання кваліфікаційної роботи було створено інтелектуальну систему відстеження обличчя, яка поєднує апаратні та програмні рішення для автоматичного визначення і відслідковування положення обличчя в режимі реального часу. Основна мета розробки полягала в автоматизації процесу відстеження обличчя з використанням сучасних технологій комп'ютерного зору та алгоритмів машинного навчання.

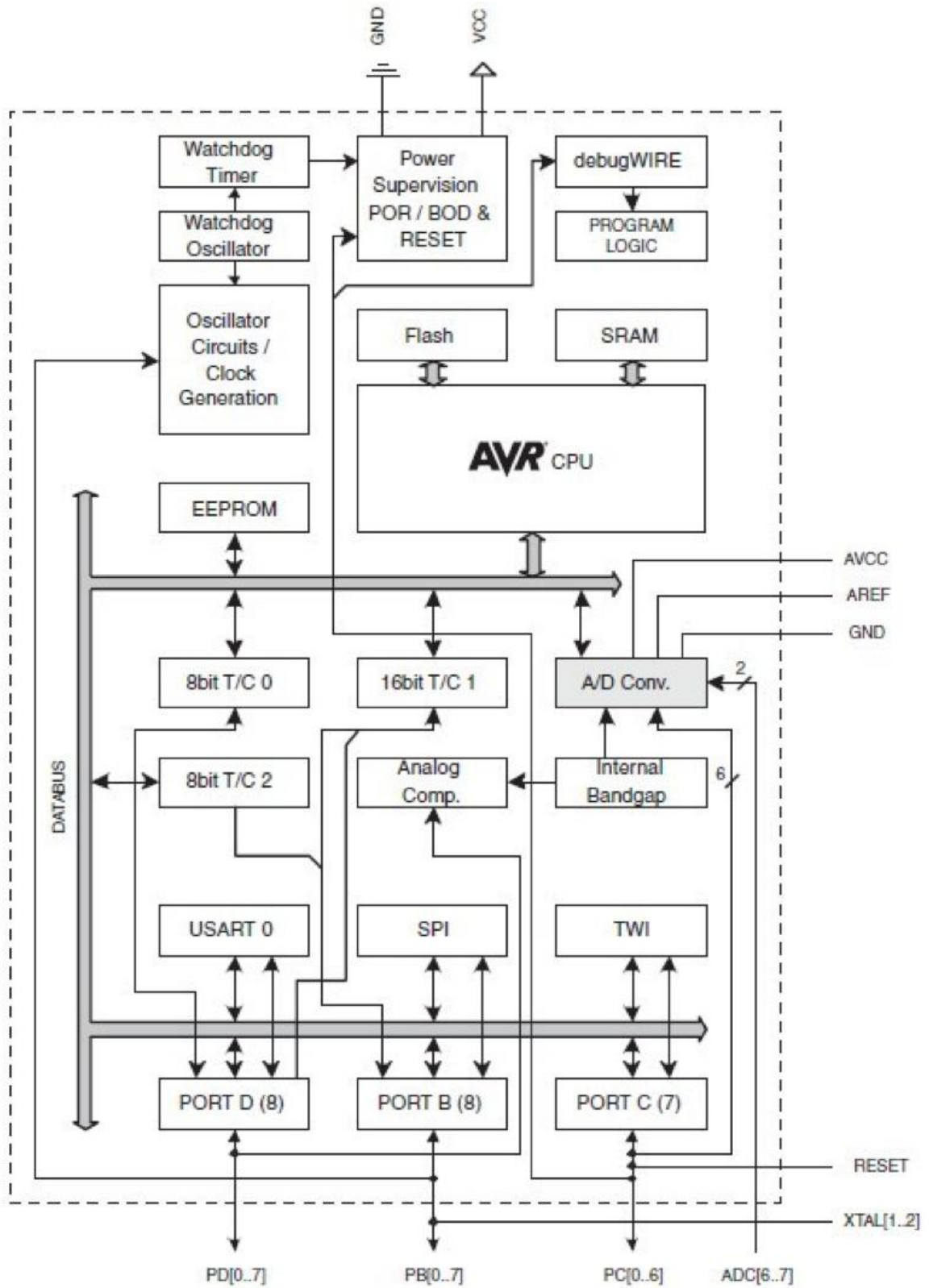
Було проведено аналіз предметної області та огляд існуючих методів та алгоритмів комп'ютерного зору та зокрема детекції обличчя у відеопотоці; обґрунтовано вибір компонентів для апаратної частини системи; описано розробку програмного забезпечення, що охоплює всі аспекти функціонування системи, включаючи алгоритми навчання системи та відстеження обличчя; проведено тестування макетного зразка.

Результати тестування проекрованої системи підтвердили працездатність та ефективність розробленої в кваліфікаційній роботі інтелектуальної системи детектування обличчя.

18

## ДОДАТОК Б

Блок-схема внутрішньої архітектури мікроконтролера АТмега328.



## ДОДАТОК В

## Код програмного забезпечення для контролеру плати Arduino

```

#include <Servo.h>

Servo servoX;
Servo servoY;
int x = 90;
int y = 90;
void setup() {
  // put your setup code here, to run once:
  Serial.begin(9600);

  servoX.attach(9);
  servoY.attach(10);
  servoX.write(x);
  servoY.write(y);
  delay(1000);
}
char input = ""; //serial input is stored in this variable
void loop() {
  // put your main code here, to run repeatedly:
  if(Serial.available()){ //checks if any data is in the serial
buffer
  input = Serial.read(); //reads the data into a variable
  if(input == 'U'){
    servoY.write(y+5); //adjusts the servo angle according to
the input
    y += 5; //updates the value of the angle
  }
  else if(input == 'D'){
    servoY.write(y-5);
    y -= 5;
  }
  else{
    servoY.write(y);
  }
  if(input == 'L'){
    servoX.write(x-5);
    x -= 5;
  } else if(input == 'R'){
    servoX.write(x+5);
    x += 5;
  }
  else{
    servoX.write(x);
  }
  input = ""; //clears the variable
}
}

```

## Додаток Г

## Код програмного забезпечення для комп'ютера

```

import cv2
import numpy as np
import serial
import time

ard = serial.Serial("COM3", 9600)
face_cascade =
cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
vid = cv2.VideoCapture(0)
while True:
    _, frame = vid.read()#reads the current frame to the
variable frame
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)#converts
frame -> grayscaled image

    #the following line detects faces.
    #First parameter is the image on which you want to
detect on
    #minSize=() specifies the minimum size of the face in
terms of pixels
    faces = face_cascade.detectMultiScale(gray, minSize=(80,
80), minNeighbors=5)

    #A for loop to detect the faces.
    for (x, y, w, h) in faces:
        cv2.rectangle(frame, (x, y), (x+w, y+h), (255, 0, 0),
2)#draws a rectangle around the face
        Xpos = x+(w/2)#calculates the X co-ordinate of the
center of the face.
        Ypos = y+(h/2)#calcualtes the Y co-ordinate of the
center of the face
        if Xpos > 280:
            #The following code
blocks check if the face is
            ard.write('L'.encode()) #on the left, right, top
or bottom with respect to the
            time.sleep(0.01) #center of the frame.
        elif Xpos < 360:
            #If any of the
conditions are true, it send a command to
            ard.write('R'.encode()) #the arduino through the
serial bus.
            time.sleep(0.01)
        else:
            ard.write('S'.encode())
            time.sleep(0.01)
        if Ypos > 280:
            ard.write('D'.encode())
            time.sleep(0.01)
        elif Ypos < 200:





```

```
        ard.write('U'.encode())
        time.sleep(0.01)
    else:
        ard.write('S'.encode())
        time.sleep(0.01)
    break

    cv2.imshow('frame', frame)#displays the frame in a seperate
window.
    k = cv2.waitKey(1)&0xFF
    if(k == ord('q')): #if 'q' is pressed on the keyboard, it
exits the while loop.
        break

vid.release() #stops receiving video from the web cam
ard.close() #closes the serial communication
cv2.destroyAllWindows() #closes all windows
```

ВІДОМІСТЬ КВАЛІФІКАЦІЙНОЇ РОБОТИ  
«Інтелектуальна система детекції обличчя»

	Прізвище та ініціали відповідальної особи	Підпис	Дата
Роботу виконав здобувач групи КІУКІ-21-9 Структура кваліфікаційної роботи: – пояснювальна записка 61 с.; – графічний матеріал 18 сл.	Роскошний В.Р.		29.05.2025
Керівник роботи	Мірошник А.М.		29.05.2025
Перевірка на антиплагіат здійснено, відповідальна особа	Литвинова Є.І.		01.06.2025
Нормоконтроль проведено :	Мірошник А.М.		29.05.2025