Державний торговельно-економічний університет

Кафедра комп'ютерних наук та інформаційних систем

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

«Програмна реалізація інтелектуальної пропускної системи на основі механізмів розпізнавання образів засобами Python»

Студента 2 курсу, гр.. 4м

Масовця
Олександра
Леонідовича

_____
підпис студента

спеціальності
122 «Комп'ютерні науки»

Науковий керівник
кандидат технічних наук, доцент

_____
підпис керівника

Томашевська
Тетяна
Володимирівна

Гарант освітньої програми
доктор фізико-математичних наук,
професор

_____
підпис керівника

Пурський Олег
Іванович

Київ 2023

Державний торговельно-економічний університет

Факультет інформаційних технологій
Кафедра комп'ютерних наук та інформаційних систем
Спеціальність 122 «Комп'ютерні науки»

Затверджую
Зав. кафедри _____Пурський О. І.
« 9 » грудня 2022р.

Завдання
на випускну кваліфікаційну роботу (проект) студенту

Масовцю Олександру Леонідовичу
(прізвище, ім'я, по батькові)

1. Тема випускної кваліфікаційної роботи (проекту)
«Програмна реалізація інтелектуальної пропускної системи на основі механізмів розпізнавання образів засобами Python»
Затверджена наказом ректора від *«06» грудня 2022 р. № 3284*
2. Строк здачі студентом закінченої роботи *25 листопада 2023 року*
3. Цільова установка та вихідні дані до роботи
Мета роботи: обґрунтування та розробка інтелектуальної пропускної системи на основі використання сучасних механізмів розпізнавання образів
Об'єкт дослідження: процеси проектування та розробки інтелектуальних систем обробки зображень.
Предмет дослідження: методи та засоби розробки інтелектуальних пропускних систем на основі розпізнавання образів
4. Перелік графічного матеріалу_____
_____
_____

5. Консультанти по роботі із зазначенням розділів, за якими здійснюється консультування:

Розділ	Консультант (прізвище, ініціали)	Підпис, дата	
		Завдання видав	Завдання прийняв
1	Томашевська Т. В.		
2	Томашевська Т. В.		
3	Томашевська Т. В.		

6. Зміст випускної кваліфікаційної роботи (проекту) (перелік питань за кожним розділом)

ВСТУП
РОЗДІЛ 1. АНАЛІЗ ПІДХОДІВ ДО РОЗРОБКИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ В СФЕРІ БЕЗПЕКИ
1.1. Сучасні тенденції використання інформаційних технологій для вирішення безпекових питань .
1.2. Огляд підходів до реалізації систем розпізнавання зображень в системах безпеки
1.3. Вимоги до інтелектуальної пропускної системи
Висновки до розділу
РОЗДІЛ 2. ПРОЕКТУВАННЯ ІНТЕЛЕКТУАЛЬНОЇ ПРОПУСКНОЇ СИСТЕМИ
2.1. Моделювання інтелектуальної пропускної системи
2.2. Розробка математичного забезпечення інтелектуальної пропускної системи
2.3. Розробка інформаційного забезпечення інтелектуальної пропускної системи
Висновки до розділу
РОЗДІЛ 3. РОЗРОБКА ІНТЕЛЕКТУАЛЬНОЇ ПРОПУСКНОЇ СИСТЕМИ
3.1. Програмна реалізація інтелектуальної пропускної системи
3.3. Розробка інтерфейсу пропускної системи
3.3. Тестування роботи пропускної системи
Висновки до розділу
ВИСНОВКИ
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

7. Календарний план виконання роботи

№ пор.	Назва етапів випускної кваліфікаційної роботи	Строк виконання етапів роботи	
		За планом	фактично
1	2	3	4
1	*Вибір теми випускної кваліфікаційної роботи*	*01.11.2022*	*01.11.2022*
2	*Розробка та затвердження завдання на випускну кваліфікаційну роботу*	*09.12.2022*	*09.12.2022*
3	*Вступ*	*01.05.2023*	*01.05.2023*
4	*Розділ 1. Аналіз підходів до розробки інтелектуальних систем в сфері безпеки*	*14.06.2023*	*14.06.2023*
5	*Розділ 2. Проектування інтелектуальної пропускної системи*	*20.06.2023*	*20.06.2023*
6	*Розділ 3. Розробка інтелектуальної пропускної системи*	*08.09.2023*	*08.09.2023*
7	*Висновки*	*20.10.2023*	*20.10.2023*
8	*Здача випускної кваліфікаційної роботи на кафедру науковому керівнику*	*02.11.2023*	*02.11.2023*
9	*Попередній захист випускної кваліфікаційної роботи*	*22.11.2023*	*22.11.2023*
10	*Виправлення зауважень, зовнішнє рецензування випускної кваліфікаційної роботи*	*29.11.2023*	*29.11.2023*
12	*Представлення готової зшитої випускної кваліфікаційної роботи на кафедру*	*04.12.2023*	*04.12.2023*
13	*Публічний захист випускної кваліфікаційної роботи*	*За розкладом роботи ЕК*	

8. Дата видачі завдання «24» грудня 2022 р.

9. Керівник випускної кваліфікаційної роботи (проекту)
Томашевська Т. В.
(прізвище, ініціали, підпис)

10. Гарант освітньої програми Пурський О. І.
(прізвище, ініціали, підпис)

11. Завдання прийняв до виконання студент-дипломник
Масовець О. Л.
(прізвище, ініціали, підпис)

12. Відгук керівника випускної кваліфікаційної роботи (проекту)

____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____
____

Керівник випускної кваліфікаційної роботи (проекту)

____
(підпис, дата)

13. Висновок про випускну кваліфікаційну роботу (проект)

Випускна кваліфікаційна робота (проект) студента Масовця О. Л.
(прізвище, ініціали)
може бути допущена до захисту в екзаменаційній комісії.

Гарант освітньої програми_________________________Пурський О. І.
(підпис, прізвище, ініціали)

Завідувач кафедри ________________________________Пурський О. І.
(підпис, прізвище, ініціали)

«_____»_________________2023 р.

ДТЕУ ДЕРЖАВНИЙ ТОРГОВЕЛЬНО-ЕКОНОМІЧНИЙ УНІВЕРСИТЕТ

ВСТУП

Актуальність теми полягає в тому, що розпізнавання образів та інші суміжні завдання стають все більш важливими у сучасному світі. Інтелектуальна пропускна система може використовуватися для автоматичного розпізнавання облич, штрих-кодів, QR-кодів або інших типів ідентифікаційних маркерів для контролю доступу до будівель, приміщень або областей, де важлива безпека і облік пересування людей.

Мета дослідження полягає у створенні системи, яка може автоматично ідентифікувати та контролювати доступ осіб до певних приміщень або територій на основі аналізу образів.

Загальні завдання дослідження включають:

- захоплення зображення: система повинна мати можливість підключатись до камери або іншого пристрою для захоплення відео та отримувати зображення особи;
- пре-процесування зображення: перед подальшим розпізнаванням образів, зображення може бути піддане певним операціям пре-процесування, таким як ідентифікація особи, покращення якості зображення обличчя особи на моніторі тощо. Це допоможе покращити якість обробки зображення та розпізнавання облич;
- виявлення облич: застосовуючи алгоритми розпізнавання облич, система може виділяти та виявляти обличчя на зображеннях. Це може включати використання методів, таких як каскади Хаара, нейронні мережі, методи детекції ключових точок тощо;
- витягування ознак: після виявлення облич, система може виконувати витягування ознак з образів, що представляють унікальні характеристики особи та її персональні дані. Це можуть бути вектори ознак, що описують форму обличчя, положення очей, рота тощо. Для цього можна використовувати навчені моделі, такі як нейронні мережі або локальні бінарні шаблони (LBP);

- порівняння з базою даних: отримані ознаки порівнюються з базою даних, яка містить відомі обличчя та відповідні дані.

Основні конкретизовані завдання даної тематики включають:

- збір та побудова набору даних: завданням є збір достатньої кількості зображень для тренування моделей розпізнавання облич та побудови бази даних з відомими обличчями. Це може включати фотографування осіб, які мають доступ до приміщення, та використання цих зображень для тренування та тестування моделей;
- розробка алгоритмів розпізнавання облич: завдання полягає у виборі та реалізації алгоритмів розпізнавання облич, які можуть ефективно виявляти та витягувати ознаки з образів. Це може включати використання популярних бібліотек, таких як OpenCV, TensorFlow або PyTorch, для побудови та навчання нейронних мереж, або використання готових моделей, які можна налаштувати та адаптувати до конкретних потреб системи;
- зберігання та управління базою даних: завданням є розробка механізму для зберігання та управління базою даних, яка містить інформацію про відомі обличчя. Це може включати використання реляційних баз даних, таких як MySQL або PostgreSQL, або спеціалізованих систем для зберігання облич, таких як Redis або Elasticsearch;
- інтеграція з системою контролю доступу: завдання полягає у розробці інтерфейсу та інтеграції програмної реалізації з існуючою системою контролю доступу.

Об'єктом дослідженя є використання таких компонентів і бібліотек як камера або інший джерело вхідного відеосигналу, бібліотека OpenCV, модель для розпізнавання об'єктів і алгоритми обробки зображень та розпізнавання об'єктів.

Предметом дослідження є розробка системи, яка може автоматично розпізнавати обличчя людей або інші об'єкти на вхідних зображеннях або

відеопотоці.

Методами дослідження є вибір бібліотек та фреймворків (OpenCV, TensorFlow, PyTorch, scikit-learn тощо), збір та підготовка даних та розробка моделей_розпізнавання_образів.

Інформаційна база дослідження є бібліотеки машинного навчання, набори даних, навчання моделі, тестування і оцінка моделі.

РОЗДІЛ 1

АНАЛІЗ ПІДХОДІВ ДО РОЗРОБКИ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ В СФЕРІ БЕЗПЕКИ

1.1. Сучасні тенденції використання інформаційних технологій для вирішення безпекових питань.

Аналіз підходів до розробки інтелектуальних систем в сфері безпеки включає в себе розгляд сучасних тенденцій використання інформаційних технологій для вирішення безпекових питань. В даному випадку доцільно розглянути наступні основні підходи та сучасні тенденції в цій області:

1. Машинне навчання і штучний інтелект (ШІ). Завдяки розвитку машинного навчання та ШІ, стали можливими нові підходи до аналізу та передбачення безпекових загроз. Системи машинного навчання можуть використовуватися для виявлення аномалій, класифікації підозрілих дій, прогнозування ризиків та автоматизації безпекових процесів. Вони можуть аналізувати великі обсяги даних для виявлення патернів та залежностей, що допомагає виявляти загрози та реагувати на них швидко.

2. Аналітика великих даних (АВД). Використання аналітики великих даних стало невід'ємною частиною безпекових систем. Великі обсяги даних, які збираються з різних джерел (наприклад, відеоспостереження, соціальні медіа, транзакції тощо), можуть бути проаналізовані для виявлення аномалій, прогнозування поведінки та ідентифікації загроз. АВД допомагає забезпечити швидке реагування на небезпеку та прийняття обґрунтованих рішень.

3. Кібербезпека. У зв'язку з зростанням кіберзагроз, безпека мереж та інформаційних систем стала важливим аспектом. Розробка

інтелектуальних систем для виявлення, відстеження та протидії кібератакам є актуальним завданням. Штучний інтелект може використовуватися для автоматизації виявлення загроз, аналізу потенційних вразливостей, реагування на кібератаки та відновлення роботи систем.

4. Автоматизовані системи безпеки. Сучасні технології дозволяють створювати автоматизовані системи безпеки, що інтегрують різноманітні сенсори, камери, детектори та інші пристрої для виявлення загроз та автоматичної реакції на них. Ці системи можуть використовувати різні технології, такі як комп'ютерне зору, обробку сигналів, машинне навчання тощо, для виявлення небезпек та попередження про них.

5. Інтернет речей (IP) і безпека. З розвитком IP, з'явилися нові виклики з точки зору безпеки. Багато з'єднаних пристроїв можуть бути вразливими до атак, тому необхідно розробляти інтелектуальні системи для моніторингу та захисту IP-інфраструктури. Використання штучного інтелекту та аналітики даних може допомогти виявити незвичайну активність та запобігти кібератакам на пристрої IP.

Ці підходи та тенденції показують, що розробка інтелектуальних систем в сфері безпеки заснована на використанні сучасних інформаційних технологій, таких як машинне навчання, аналітика великих даних, кібербезпека, автоматизація та IP. Ці технології дозволяють покращити ефективність виявлення загроз, швидкість реагування та прийняття рішень у сфері безпеки.

Зважаючи на зростання загроз в сфері безпеки, розробка інтелектуальних систем стала необхідністю для ефективного виявлення, аналізу та протидії різноманітним безпековим загрозам. Тут також доречно навести наступні підходи і сучасні тенденції в розробці інтелектуальних систем в сфері безпеки:

1. Глибинне навчання (ГН). Глибинне навчання є підгрупою машинного навчання, яке використовує нейронні мережі зі штучними нейронами, щоб розпізнавати складні патерни та залежності в даних. У сфері

безпеки глибинне навчання застосовується для виявлення шкідливих програм, ідентифікації зловмисних активностей, аналізу великих обсягів мережевого трафіку та інших безпекових завдань. Воно дозволяє автоматизувати процеси виявлення та реагування на загрози, а також покращує точність виявлення за рахунок навчання на великій кількості даних.

2. Аналіз поведінки (АП). Цей підхід передбачає стеження за нормальною поведінкою системи, користувачів або мережевого трафіку, щоб виявити аномальну або підозрілу активність. Застосовується аналіз поведінки для виявлення вторгнень, викриття витоку даних, виявлення несанкціонованого доступу та інших безпекових загроз. Алгоритми машинного навчання використовуються для побудови моделей поведінки та автоматичного розпізнавання аномалій.

3. Розподілена обробка даних та обчислення. З ростом обсягів даних та потужності обчислювальних систем, виникає потреба у розподіленій обробці та аналізі безпекових даних. Це означає використання розподілених систем зберігання даних та обчислювальних ресурсів, щоб швидко та ефективно аналізувати великі обсяги інформації. Розподілена обробка даних дозволяє покращити швидкість виявлення загроз та реагування на них у реальному часі.

4. Використання розширеної реальності та віртуальної реальності (VR та AR). AR та VR технології широко використовуються у сфері безпеки для симуляції небезпечних ситуацій, навчання персоналу, тренування реакцій та аналізу безпекових систем. Вони дозволяють створювати інтерактивні середовища для моделювання реальних сценаріїв та вирішення безпекових питань без фактичного втручання в реальний світ.

5. Використання блокчейн технологій. Блокчейн технології впроваджуються в сфері безпеки для забезпечення надійного зберігання та обміну безпекової інформації. Блокчейн може забезпечити захист даних, цілісність історії та відстеження змін, що робить його корисним для аудиту

безпеки, керування доступом та обміну безпекових подій.

Ці підходи та тенденції використання інформаційних технологій у сфері безпеки показують постійний розвиток та стрімкий прогрес у забезпеченні безпеки систем та даних. Інтеграція розумних алгоритмів, аналітики даних та інноваційних технологій допомагає покращити виявлення, аналіз та реагування на безпекові загрози, забезпечуючи більш ефективне та надійне функціонування систем безпеки.

Також варто звернути увагу на менш відомі, але не менш важливі технології щодо вирішення ефективності та точності розпізнавання зображень. До них можна віднести:

1. *Кіберфізична безпека (CPS Security)*: кіберфізична безпека об'єднує інформаційні технології та фізичну інфраструктуру для захисту систем, які контролюють індустріальні процеси, енергетичні системи, медичні пристрої тощо. Ця тенденція включає в себе захист від кібератак на об'єкти фізичної інфраструктури.

2. *Квантова криптографія*: квантова криптографія використовує принципи квантової механіки для створення абсолютно надійних систем шифрування. Вона також створює незламність квантових ключів, для створення абсолютно надійних систем шифрування. Це може вирішити проблеми, пов'язані з
розширенням обчислювальних можливостей квантових комп'ютерів, які можуть зламувати сучасні шифри.

3. *Децентралізовані системи безпеки (Decentralized Security)*: технології блокчейну та розподілені реєстри можуть бути використані для створення децентралізованих систем безпеки, які не мають центральних точок вразливості і забезпечують високий рівень захисту даних та операцій. Вони також працюють без центральних точок збереження даних, що робить їх менш вразливими до атак та зламу.

4. *Використання штучного інтелекту у виявленні загроз*: методи машинного навчання та штучного інтелекту використовуються для аналізу

великих обсягів даних з метою виявлення аномалій та загроз у реальному часі. Це дозволяє вдосконалити системи моніторингу та реагування на потенційні ризики.

5. *Безпека Інтернету речей (IoT Security)*: зі зростанням кількості підключених пристроїв до Інтернету речей, безпека їхнього функціонування стала важливою темою. Інформаційні технології використовуються для розробки протоколів (HTTPS) і заходів захисту для IoT-пристроїв.

6. *Захист від соціальної інженерії*: сучасні техніки соціальної інженерії можуть використовувати соціальні медіа, фейкові новини та інші технології для маніпуляції індивідами та організаціями. Захист від цих атак вимагає інформаційних та психологічних методів. Захист від соціальної інженерії охоплює стратегії та технології, спрямовані на захист від маніпуляцій та обману осіб через фейкові новини та інші канали впливу. Це включає в себе інформаційні кампанії, які сприяють свідомому та критичному мисленню.

Зважаючи на швидкий розвиток технологій, менш відомі тенденції використання інформаційних технологій для вирішення безпекових питань завжди цікаві.

До даних тенденцій можна віднести захист DNA-даних, адже за допомогою біотехнологій та інформаційних технологій стає можливим захист особистих DNA-даних від несанкціонованого доступу та зламу. Це стає важливим в контексті розширення генетичних тестів та медичних досліджень.

Також варто звернути увагу на застосування квантових обчислювальних систем, оскільки однією зі складних але важливих тенденцій є розвиток квантових обчислювальних систем, які можуть ламати сучасні шифри. Це створює потребу у розробці квантово-стійких методів шифрування та захисту даних.

Серед цієї тенденції також має місце біометричний аналіз структури кісток. Всі нові технології використовують біометричний аналіз структури

кісток для ідентифікації осіб. Це може бути використано для безпечного входу в приміщення або для підтвердження ідентичності на відстані.

Цікавою тенденцією є аналіз психофізіологічних показників, адже за допомогою сенсорів та аналітики даних можна аналізувати психофізіологічні показники, такі як рівень стресу, пульс, теплові реакції тощо. Це може бути використано для виявлення підозрілих осіб або визначення їхнього стану.

Тут також варто віднести захист від атак в ігрових середовищах та захист від кіберзлочинності в медичних системах. Велика кількість цифрових атак відбуваються в ігрових середовищах, де використовуються валюти та активи. Технології блокчейну та криптографії використовуються для захисту ігрових даних та управління безпекою геймерів. Медичні пристрої стають предметом кібератак через їхню підключеність до Інтернету. Нові технології інформаційної безпеки включають в себе використання блокчейну та end-to-end шифрування для захисту медичних даних та пристроїв.

І найбільш цікавою тенденцією є геопросторова аналітика, адже використання геопросторової аналітики та супутникових даних допомагає вирішувати безпекові питання, такі як моніторинг кордонів, виявлення незаконних видобутків природних ресурсів та аналіз великих обсягів геоданих для прогнозування подій.

Ці тенденції свідчать про те, що інформаційні технології постійно розвиваються для вирішення різноманітних завдань безпеки та відповідають на сучасні виклики.

1.2. Огляд підходів до реалізації систем розпізнавання зображень в системах безпеки

Системи розпізнавання зображень в системах безпеки використовуються для автоматичного виявлення, ідентифікації та аналізу об'єктів на зображеннях або відео.

Існує кілька підходів до реалізації таких систем, але найпоширеніші з

них включають:

1. *Класифікація з використанням машинного навчання*: цей підхід базується на навчанні моделей машинного навчання на великій кількості позитивних та негативних зображень. Модель навчається розрізняти різні класи об'єктів і використовується для класифікації нових зображень на підставі набутих знань.

2. *Виявлення об'єктів*: цей підхід використовує алгоритми для виявлення об'єктів на зображенні або відео. Часто використовуються методи, такі як аналіз меж, виявлення руху, або детектори об'єктів, засновані на нейронних мережах.

3. *Сегментація зображення*: цей підхід спрямований на виділення окремих областей або об'єктів на зображенні. Він використовується для створення масок або контурів об'єктів, що дозволяє здійснювати більш детальний аналіз і розпізнавання об'єктів.

Під час вирішення процесу розпізнавання зображень у системах безпеки можуть виникати деякі основні питання:

1. *Вибір підходу*: не існує універсального підходу, що підходить для всіх випадків. Вибір підходу залежить від конкретного завдання, доступних даних та ресурсів.

2. *Навчання моделі*: якщо використовується підхід з машинним навчанням, потрібно мати набір даних для навчання моделі. Цей набір даних повинен бути репрезентативним для реальних умов і містити достатню кількість позитивних та негативних зображень.

3. *Обробка в реальному часі*: деякі системи безпеки вимагають обробки зображень в реальному часі. Це може бути викликом, оскільки необхідно забезпечити достатню швидкість обробки зображень без втрати точності або продуктивності.

4. *Розпізнавання ускладнених сценаріїв*: деякі сценарії вимагають розпізнавання об'єктів ускладненої форми або в умовах обмеженої видимості, наприклад, в темряві або вогні. Вирішення таких сценаріїв може бути

складним і вимагати спеціалізованих методів або додаткового обладнання.

5. *Захист від помилкових спрацьовувань*: системи розпізнавання зображень можуть сприймати невідповідні об'єкти або помилково не розпізнавати реальні загрози.

Важливо розробити методи для зменшення кількості помилкових спрацьовувань і забезпечення високої точності системи.

Ці питання вимагають уваги та досліджень для успішної реалізації систем розпізнавання зображень у системах безпеки. Крім того, варто зазначити, що система розпізнавання зображень може стикатися з великим обсягом даних або великою кількістю одночасних запитів. Вона повинна бути масштабованою, тобто здатною ефективно працювати зі зростаючими обсягами даних і обробляти їх у відповідний спосіб. Також, системи безпеки часто працюють з конфіденційними даними, тому важливо забезпечити високий рівень приватності і безпеки. Тут варто розглядати використання методів шифрування, аутентифікації та контролю доступу для захисту даних системи розпізнавання зображень.

Системи розпізнавання зображень в системах безпеки мають багато історії, що ведуть від ранніх досліджень до сучасних технологій. В 1950-і - 1960-і роки були розроблені перші спроби автоматичного розпізнавання зображень. Основними методами були використання витягування характеристик, таких як краї та текстури, і подальшого порівняння з базовими зразками. В 2000-ті роки з'явилися перші системи розпізнавання зображень, в основу яких були покладені сверточні нейронні мережі (CNN). Ці системи стали ефективнішими в розпізнаванні об'єктів на зображеннях та виявленні патернів. На сьогоднішній день системи розпізнавання зображень в системах безпеки поєднують різні підходи та використовують сучасні глибокі нейронні мережі, що дозволяє досягти високої точності розпізнавання об'єктів та ситуацій. Крім того, використання додаткових технологій, таких як обробка зображень в реальному часі та аналіз відеопотоків, робить системи розпізнавання зображень в системах безпеки

ще більш потужними та ефективними.

Існують деякі проблеми, пов'язані з оглядом підходів до реалізації систем розпізнавання зображень в системах безпеки. Серед них може бути недостатність даних. Адже для ефективного навчання моделей розпізнавання зображень потрібно мати великі та репрезентативні набори даних. Однак, отримання достатньої кількості позначених даних може бути викликом, особливо коли йдеться про рідкісні події або об'єкти, що потребують розпізнавання в системах безпеки. Дані проблеми потрібно розглядати й ураховувати при розробці та впровадженні систем розпізнавання зображень в системах безпеки для забезпечення їх ефективності та надійності

Системи розпізнавання зображень грають важливу роль у сфері безпеки, включаючи відеоспостереження, контроль доступу та виявлення злочинів. Ось огляд деяких менш відомих або менш загальновідомих підходів до реалізації таких систем:

1. *Глибоке навчання для виявлення аномалій*: глибоке навчання, зокрема нейронні мережі, може використовуватися для виявлення аномалій на відеозаписах. За допомогою навчання на прикладах можна створити модель, яка розпізнає незвичайну поведінку або об'єкти на відео. Цей підхід дозволяє виявляти ситуації, які не можуть бути передбачені заздалегідь.

2. *Використання мультимодальних даних*: підходи, які поєднують різні види даних, такі як відео, аудіо та дані з сенсорів, можуть покращити точність систем розпізнавання. Наприклад, спільна обробка відео та аудіозаписів може допомогти виявити події, такі як агресивна розмова або сутичка.

3. *Використання геопросторової інформації*: геопросторова інформація може бути важливою для систем безпеки, особливо для відстеження руху та аналізу подій у реальному часі. Географічні дані можуть допомогти встановити місце події або злочину, а також визначити оптимальні шляхи втручання.

4. *Використання квадрокоптерів та дронів*: дрони і квадрокоптери

стають все більш популярними для використання в системах безпеки. Вони можуть використовуватися для візуального нагляду з висоти, пошуку загрози та надання додаткових даних для систем розпізнавання.

5. *Використання обробки природних мов*: Аналіз тексту і обробка природних мов можуть бути корисними для систем безпеки, зокрема для моніторингу соціальних медіа, детектування загроз та аналізу текстової інформації для розуміння ситуацій.

6. *Використання технологій блокчейну*: Технологія блокчейну може бути використана для забезпечення безпеки даних та заборони несанкціонованого доступу до важливих систем.

Дані підходи можуть бути менш відомими, але вони представляють інноваційні можливості для покращення безпеки і стали важливими складовими великих систем безпеки.

Загалом, рідкісна інформація про системи розпізнавання зображень у сфері безпеки включає аспекти теплової інфрачервоної спектроскопії. Вона дозволяє розпізнавати об'єкти на основі їх теплового випромінювання. Це може бути корисним для виявлення людей або тварин у темряві або в умовах обману. До цього пункту також можна віднести використання акустичних сенсорів, адже акустичні сенсори можуть виявляти звуки, які не завжди видно на відео, наприклад, голоси або навіть вибухи. Їх використання може покращити системи виявлення загроз. Серед цих систем також поширений аналіз графічних даних. Дані системи можуть використовувати аналіз графічних даних, таких як схеми, рисунки або логотипи, для виявлення підробок або незаконних змін у системах безпеки.

Також варто звернути увагу на використання біометричних даних та квантових технологій, адже окрім відомих біометричних методів, таких як відбитки пальців і розпізнавання обличчя, існують більш рідкісні біометричні характеристики, такі як розпізнавання вен на долонях або звуковий аналіз голосу. Дослідження в області квантових обчислень може призвести до створення надійних систем розпізнавання зображень з

неперевершеною обчислювальною потужністю.

Ще одним із маловідомих факторів є розпізнавання міміки обличчя, оскільки покращені системи розпізнавання міміки обличчя можуть допомогти виявити емоційний стан осіб на відеозаписах, що може бути корисним у відстеженні психологічного стану та виявленні загроз.

Всі вище перераховані підходи і технології є рідкісними, але демонструють що область систем безпеки постійно розвивається і використовує різноманітні інноваційні методи для покращення ефективності та точності розпізнавання зображень.

Розвиток систем розпізнання інтелектуальних систем в останні роки спрямований на покращення їхньої точності, швидкості та універсальності. Використання глибокого навчання і нейромереж дозволяє створювати більш складні та гнучкі моделі, які здатні до розпізнавання складних зразків та абстракцій. Одним з напрямків розвитку є поєднання різних модальностей, таких як обробка тексту, аудіо та візуальна інформація в єдиній системі. Наприклад, системи мовленнєвого інтерфейсу, які використовують розпізнавання мовлення, обробку природної мови і відновлення тексту з аудіо-записів. Іншим напрямком є забезпечення системам можливості навчатися в реальному часі та адаптуватися до нових умов. Наприклад, персоналізовані системи рекомендацій, які використовують нейромережі для аналізу користувацьких взаємодій та навчання на основі динамічних вподобань що змінюються. Напрямки, на яких може відбуватися аналіз, можуть бути різні. Серед головних це *машинне навчання, нейромережі, мультимодальність, машинне навчання, бази даних, інженерія з бази даних* та інші напрямки.

1.3 Вимоги до інтелектуальної пропускної системи

Інтелектуальна пропускна система може мати різні вимоги, залежно від конкретного контексту і потреб користувачів, але до основних вимог відносяться наступні:

1. *Безпека*: вимога до забезпечення безпеки є ключовою для будь-

якої пропускної системи. Інтелектуальна система повинна мати вбудовані механізми аутентифікації та авторизації, щоб гарантувати, що лише авторизовані особи отримують доступ.

2. *Ідентифікація та аутентифікація*: інтелектуальна пропускна система повинна мати здатність ідентифікувати особу і перевіряти її автентичність перед наданням доступу. Це може здійснюватися за допомогою різних технологій, таких як біометричні дані (відбитки пальців, сканування обличчя і т.д.), ідентифікаційні картки або паролі.

3. *Гнучкість*: пропускна система повинна бути гнучкою і пристосовуватися до різних потреб і контекстів. Вона може включати можливість налаштування рівнів доступу для різних користувачів, налаштування режимів роботи (наприклад, розкладу роботи) та здатність пристосовуватися до змінних умов.

4. *Інтеграція з іншими системами*: інтелектуальна пропускна система може вимагати інтеграції з іншими системами безпеки або управління, такими як системи відеоспостереження, системи контролю доступу, системи виявлення вторгнень тощо. Інтеграція з іншими системами дозволяє покращити ефективність та забезпечити більш широкий контекст контролю доступу.

5. *Аналітика та звітність*: інтелектуальна пропускна система може мати вбудовані функції аналітики та звітності, які дозволяють відстежувати та аналізувати дані про вхід і вихід користувачів, відвідуваність, тривалість перебування тощо. Це може бути корисно для управління безпекою, внутрішньою логістикою та статистичного аналізу.

6. *Масштабованість*: інтелектуальна пропускна система повинна бути здатна працювати з великим обсягом користувачів і забезпечувати швидку обробку запитів. Вона також повинна бути масштабованою і здатною розширюватися у випадку збільшення обсягу роботи.

Зазначені вимоги є загальними і можуть бути додатково встановлені в залежності від конкретних потреб та вимог конкретної організації чи місця використання системи пропуску.

Окремі вимоги до інтелектуальної пропускної системи можуть варіюватися в залежності від конкретного контексту та потреб користувачів. Однак, існують наступні кілька можливих окремих вимог, які можуть бути встановлені:

1. *Багатофакторна аутентифікація*: вимога до використання не одного, а кількох факторів аутентифікації для забезпечення більш високого рівня безпеки. Наприклад, може вимагатися поєднання біометричних даних (відбитків пальців, сканування обличчя) з ідентифікаційною карткою або паролем.

2. *Інтеграція з мобільними пристроями*: вимога до системи, яка може інтегруватися з мобільними пристроями, такими як смартфони, для зручного та безконтактного доступу. Це може включати використання технологій NFC (безконтактний обмін даними), QR-кодів або мобільних додатків для автентифікації та доступу.

Тим не менш, існують інші менш відомі, але не менш важливі вимоги до інтелектуальної пропускної системи. Вони можуть включати такі аспекти:

1. *Захист від фізичного підроблення*: система повинна бути здатна виявляти та захищати пропускні картки або ідентифікатори від фізичного підроблення, такого як підробка, копіювання або модифікація.

2. *Спеціальні облікові записи*: для певних застосувань, наприклад, у важливих державних об'єктах або дослідницьких лабораторіях, може вимагатися можливість створювати спеціальні облікові записи для користувачів із підвищеними привілеями або доступом до конфіденційної інформації.

3. *Інтеграція з біометричними системами*: в деяких випадках може бути необхідно інтегрувати інтелектуальну пропускну систему з

біометричними засобами ідентифікації, такими як відбитки пальців, розпізнавання обличчя або сканери радужки.

4. *Моніторинг і тривожна сигналізація*: система повинна забезпечувати можливість моніторингу дій користувачів та, в разі необхідності, генерувати тривожні сигнали або аварійні повідомлення.

5. *Інтеграція з відеоспостереженням*: інтеграція з відеоспостереженням може забезпечувати візуальне підтвердження ідентифікації користувачів та відстеження їхньої активності в реальному часі.

6. *Віддалена адміністрація і керування*: може бути важливою можливість віддаленої адміністрації та керування системою, щоб забезпечити її ефективне управління та налаштування.

7. *Захист від переповнення даними*: система повинна бути захищеною від атак, пов'язаних із спробами переповнення бази даних або мережевих ресурсів.

8. *Захист від атак на перехоплення даних*: вимога полягає у захисті від можливості перехоплення інформації, передаваної між пропускними точками та центральною системою.

9. *Масштабованість та гнучкість*: в залежності від потреб користувачів, система повинна бути здатною масштабуватися і підлаштовуватися до різних сценаріїв використання.

Ці вимоги є лише прикладами і можуть бути додатково встановлені або змінені в залежності від конкретних вимог та потреб вашої організації або проекту.

Питання, які можуть відноситися до вимог щодо інтелектуальної пропускної системи, включають які функціональні можливості можуть бути надані; які види пропусків (наприклад, фізичні карти, QR-коди, біометричні дані тощо) можуть існувати; яку систему ідентифікації варто встановлювати і т.д. Це основні питання які можуть відноситися до даної тематики, які можуть виникати при розробці вимог до інтелектуальної пропускної системи. Реальні питання можуть змінюватися в залежності від конкретних потреб та

контексту використання системи.

Висновки до розділу

Даний розділ надає важливі висновки та узагальнення на основі проведеного дослідження. Розробка інтелектуальних систем в сфері безпеки є актуальною та перспективною галуззю.

Використання передових технологій штучного інтелекту, машинного навчання та комп'ютерного зору може покращити ефективність систем безпеки. Розробка інтелектуальних систем в сфері безпеки є актуальною та перспективною галуззю. Використання передових технологій штучного інтелекту, машинного навчання та комп'ютерного зору може покращити ефективність систем безпеки. Також, важливим кроком у розробці інтелектуальних систем безпеки є тестування та оцінка їхньої ефективності. Використання реальних тестових сценаріїв, метрик оцінки та валідація результатів допомагають підтвердити працездатність та надійність системи. Перед впровадженням інтелектуальних систем безпеки необхідно враховувати правові та етичні аспекти. Законодавчі обмеження щодо збору, зберігання та використання даних, а також етичні стандарти щодо використання технологій штучного інтелекту повинні дотримуватися.

Ці висновки слугують основою для подальшої розробки та впровадження інтелектуальних систем в сфері безпеки, допомагають забезпечити їхню ефективність, безпеку та відповідність вимогам.

Сучасні технології інформаційної безпеки використовують широкий спектр інструментів для захисту даних та систем від різноманітних загроз. Огляд підходів до реалізації систем розпізнавання зображень в системах безпеки показує значний прогрес у цій галузі.

Додатково, використання технологій штучного інтелекту для розпізнавання зображень значно покращило системи безпеки. Алгоритми машинного навчання навчаються розпізнавати образи, що дозволяє

автоматично виявляти підозрілі об’єкти або вчинки.

Загалом, сучасні технології в області безпеки орієнтовані на використання новаторських методів розпізнавання та ідентифікації, з великим акцентом на безпеку даних та автоматизацію процесів для забезпечення найвищого рівня захисту.

РОЗДІЛ 2
ПРОЕКТУВАННЯ ІНТЕЛЕКТУАЛЬНОЇ ПРОПУСКНОЇ СИСТЕМИ

2.1. Моделювання інтелектуальної пропускної системи

Моделювання інтелектуальної пропускної системи відбувається за допомогою мови програмування Python, бібліотек OpenCV, модуля clx.xms та requests і також можливо використати додаткову бібліотеку NumPy. Готова система наводить червоний (або інший колір) курсор у вигляді квадрату на обличчя людини, сканує її та інформує нас через окремий чат вказуючи де перебуває людина, яка потрапила в поле зору інтелектуальної системи пропуску.

Вище перераховані бібліотеки та модулі мови програмування Python відповідають за **наступні принципи роботи функціональності інтелектуальної пропускної системи**:

1. *OpenCV (cv2)*: бібліотека, мета якої є робота з алгоритмами комп'ютерного зору і обробленням зображень під час дії роботи інтелектуальної пропускної системи.

2. *clx.xms*: спеціалізований модуль, який функціонує як координатор для сканування особистості та отримання даних про неї.

3. *requests*: модуль, мета якого буде надавати спосіби здійснювати HTTP-запити, взаємодіяти з веб-серверами, отримувати від них відповіді і виконувати різні операції, пов'язані з HTTP-протоколом, такі як отримання «апдейтів» про особу, намір якої є рух далі, яка потрапила в поле зору інтелектуальної пропускної системи. Модуль буде відправлення POST- або PUT-запити.

4. *NumPy*: бібліотека, яка може знадобитися для вичислення координатів напряму особистостів в межах видимості інтелектуальної пропускної системи.

На додаток, моделювання інтелектуальної пропускної системи (далі

ІПС) також включає розробку програмного забезпечення та алгоритмів, які дозволяють автоматизувати та оптимізувати процес контролю доступу до певних зон або приміщень. Ця система може використовуватися у будівлях, компаніях, громадських спорудах тощо, де потрібно керувати потоком людей.

Моделювання інтелектуальної пропускної системи включає розробку логіки та алгоритмів, тестування та оптимізацію системи, а також інтеграцію з необхідними апаратними засобами та іншими системами. Важливим етапом є також визначення вимог до системи, специфікація функціональності та інтерфейсів, аналіз ризиків та розробка плану безпеки.

Основні компоненти інтелектуальної пропускної системи включають:

1. *Сенсори та датчики*: ІПС може використовувати різні типи сенсорів, такі як біометричні сканери (відбитки пальців, розпізнавання обличчя), картки доступу, RFID-мітки, смартфони тощо, для збору даних про особу, яка намагається отримати доступ.

2. *База даних*: ІПС може мати централізовану базу даних, в якій зберігаються інформація про користувачів, включаючи їхні біометричні дані, дозволи на доступ, розклад роботи тощо.

3. *Аналітика та алгоритми*: Інтелектуальна пропускна система може використовувати різні алгоритми та аналітичні інструменти для прийняття рішень щодо надання або відмови в доступі

4. *Керування доступом*: ІПС може мати механізми керування доступом, які дозволяють автоматично відкривати або блокувати двері, шлагбауми або інші пристрої, що контролюють доступ, в залежності від результату аутентифікації та авторизації.

5. *Інтеграція з іншими системами*: ІПС може бути інтегрована з іншими системами безпеки або управління будівлею, такими як система відеоспостереження, система контролю доступу до паркінгу, система управління освітленням тощо, щоб забезпечити більш широкий контекст

контролю доступу.

Сьогодні система пропуску у світовій практиці варіюється від вирішення проблем приватної безпеки і підвищення даної безпеки. При цьому основними технічними складовими виступають засоби телематики, орієнтовані на отримання і передачу інформації з метою вирішення завдань, пов'язаних з організацією дистанційного діагностування технічного стану пропускної системи. **Класифікація системи контролю доступом (СКД):**

- *Автономні*: інформація не передається на центральний пункт охорони і не контролюється операторами.

- *Центральні (Мережеві)*: відбувається обмін з центральним пунктом охорони для керування виконавчими пристроями.

- *Універсальні*: можуть працювати як в автономному режимі, так і в мережевому. Якщо відбувається збій центрального пристрою управління, то перемикається на автономний режим.

Моделювання інтелектуальної пропускної системи дозволяє створити ефективну та безпечну систему контролю доступу, яка може бути використана в різних сферах, включаючи комерційні будівлі, установи охорони здоров'я, навчальні заклади та багато інших.

Першим етапом моделювання є аналіз вимог інтелектуальної пропускної системи. Необхідно з'ясувати, які функції має виконувати система, які обмеження і вимоги до безпеки інформації, а також які технології і ресурси будуть використовуватися.

Після аналізу вимог виконується проектування системи. Це включає в себе створення архітектури системи, вибір необхідних компонентів, розробку інтерфейсів для користувачів та інтеграцію зовнішніх пристроїв, таких як біометричні сканери або RFID-читачі.

Після проектування системи розпочинається розробка програмного забезпечення. Це може включати створення бази даних для зберігання інформації про користувачів, розробку логіки контролю доступу, створення веб-інтерфейсу для адміністрування системи та інтеграцію

зовнішніх пристроїв.

Після розробки програмного забезпечення необхідно провести його тестування, щоб переконатися, що воно працює правильно і відповідає всім вимогам. Тестування може включати перевірку функціональності, безпеки, навантаження і здатності відновлення.

Коли система успішно пройшла тестування, вона готова до **впровадження**. Впровадження може включати установку обладнання, налаштування програмного забезпечення, навчання персоналу та перенесення існуючих даних до нової системи.

Після впровадження системи необхідно забезпечити її **підтримку і обслуговування**. Це включає в себе регулярні оновлення програмного забезпечення, відповідь на запити користувачів та моніторинг роботи системи для виявлення і усунення проблем.

Як кінцевий результат, робота інтелектуальної системи з реалізацією моделювання представлена в наступних прикладах у вигляді зображень (див. рисунки 2.1-2.4)

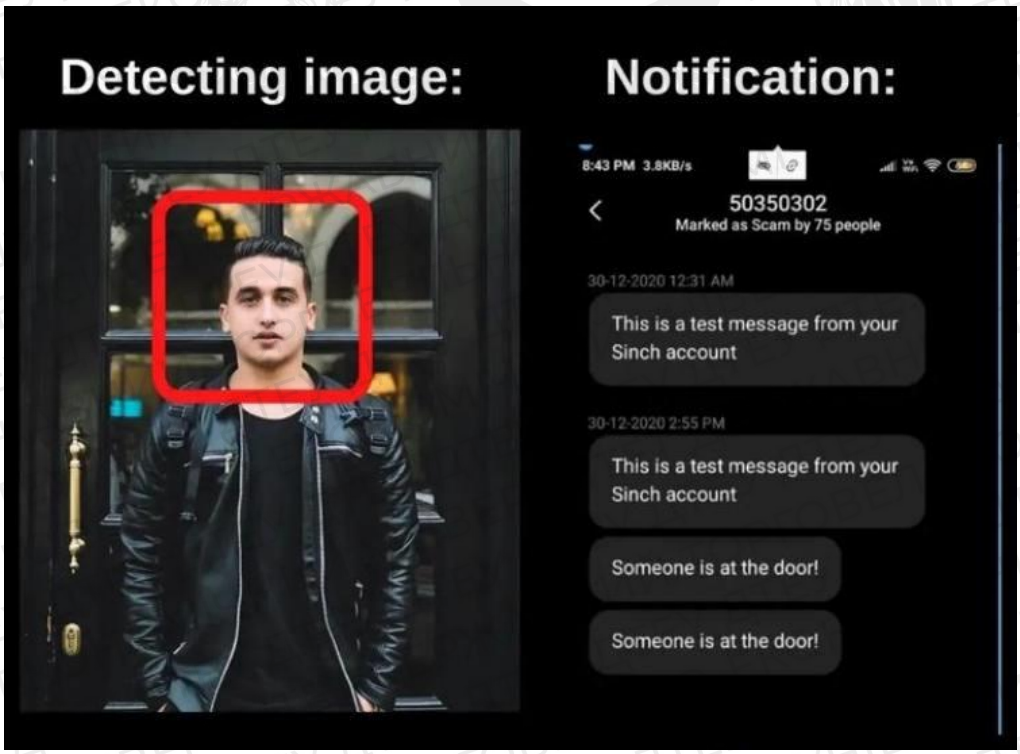

Рис. 2.1. *Приклад інтерфейсу моделювання інтелектуальної пропускної системи з функцією автовідповідача з даними про особу та/або предмет.*

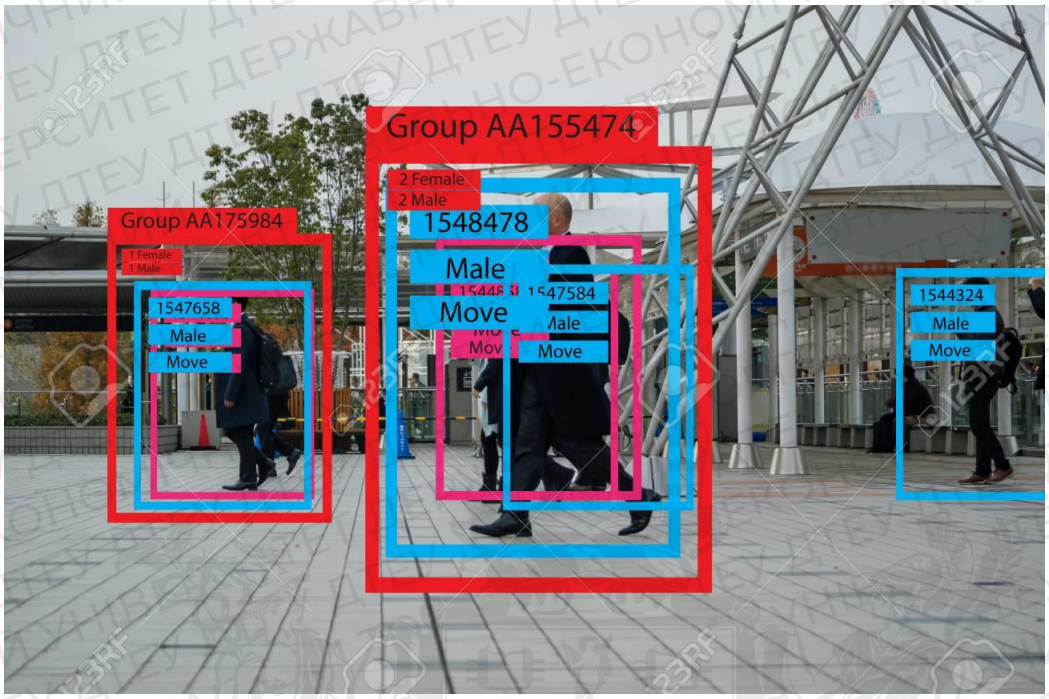

Рис. 2.2. *Приклад інтерфейсу моделювання інтелектуальної пропускної системи з автоматичним отримуванням даних про рух людей в полі зору камери системи.*

Рис. 2.3. *Приклад інтерфейсу моделювання інтелектуальної пропускної системи з діапазоном руху людей та/або предметів та їхні наступні маневри*

які очікуються даною системою в полі зору в реальному часі.

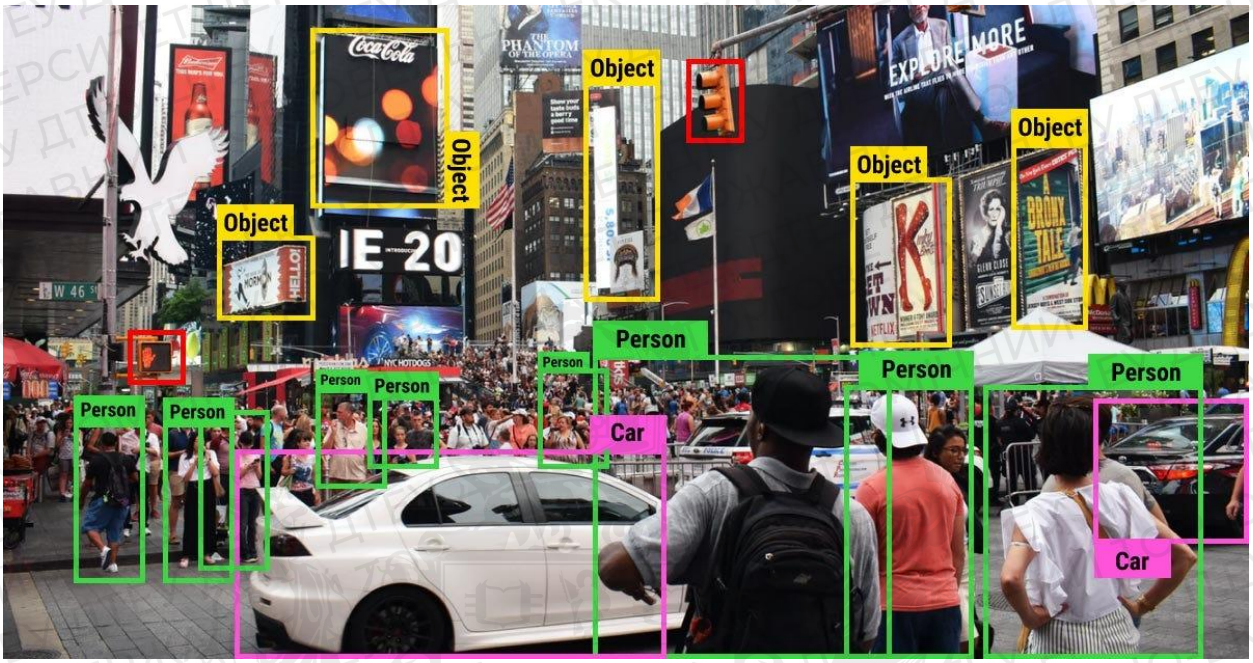

Рис. 2.4. *Приклад інтерфейсу моделювання інформаційної пропускної системи з кольоровими курсорами які отримують дані про предмет чи особу та повертають їхні значення на екран в реальному часі.*

Зображення, що продемонстровані вище, є лише прикладами того, як інтелектуальна пропускна система виконує свій функціонал та алгоритм дій щодо сканування предметів та людей щоб отримати їхні дані, тип та реальний час.

Як правило, моделювання інтелектуальної пропускної системи, яка використовує червоний курсор для сканування особи, може бути цікавим завданням. Взагалом, можна використовувати курсори різного кольору (як це продемонстровано в зображеннях вище), але найчастіше можна побачити червоний курсор. Ця система може використовувати різні технології для розпізнавання особи та керування доступом. Ось кілька кроків, які можна врахувати при моделюванні такої системи:

Вибір технологій розпізнавання особи:

1. Використання камер для збору зображень осіб та/або предметів.
2. Використання алгоритмів розпізнавання обличчя для ідентифікації осіб.
3. Використання біометричних ознак, таких як відбитки пальців або розпізнавання радужки ока.

Створення інтерфейсу:

1. Розробка користувацького інтерфейсу, який дозволяє введення інформації про користувача та керування доступом.
2. Реалізація курсора, який може використовуватися для вибору обличчя або інших об'єктів на екрані.

Аналіз та ідентифікація:

1. Відправка зображення, отриманого через курсор, для аналізу.
2. Застосування алгоритмів розпізнавання для ідентифікації особи/предмету.
3. Порівняння розпізнаної особи з базою даних користувачів.

Керування доступом:

1. Визначення, чи має користувач доступ до системи.
2. Відкриття дверей або надання іншого доступу користувачеві у разі підтвердженої ідентифікації.

Безпека та захист даних:

1. Забезпечення безпеки зібраних даних про користувачів та результатів розпізнавання.
2. Використання шифрування та інших методів захисту даних.

Тестування та налагодження:

1. Проведення тестів для перевірки точності та ефективності системи.
2. Налагодження параметрів алгоритмів розпізнавання для оптимальної роботи системи.

Підтримка та адміністрування:

1. Забезпечення можливості адміністрування системи, включаючи додавання та видалення користувачів.
2. Постійна підтримка та оновлення системи для забезпечення безпеки та надійності.

Це загальний опис процесу моделювання інтелектуальної пропускної системи з використанням червоного курсора для сканування осіб. Ви можете ознайомитися з основною організацією моделювання інтелектуальної пропускної системи на зображенні нижче (див. рис. 2.5):

Рис. 2.5. *Основна загальноприйнята організація моделювання інтелектуальної пропускної системи.*

До всього вищезгаданого в даному пункті, слід зауважити що до моделювання інтелектуальної пропускної системи входить завдання як **методи розпізнавання**. Інтелектуальні пропускні системи використовують різні методи розпізнавання для ефективного управління доступом та

ідентифікації осіб. Ось деякі з методів, які можуть бути використані:

1. *Розпізнавання обличчя (Facial Recognition)*. Цей метод базується на аналізі унікальних рис обличчя для ідентифікації осіб. Використання алгоритмів глибокого навчання, таких як Convolutional Neural Networks (CNN), дозволяє створювати точні моделі для розпізнавання обличчя в реальному часі.

2. *Розпізнавання відбитків пальців (Fingerprint Recognition)*. Цей метод використовує унікальність відбитків пальців для ідентифікації особи. Алгоритми, які використовуються для обробки відбитків пальців, зазвичай базуються на мінуванні та порівнянні характеристик відбитків.

3. *Голосове розпізнавання (Voice Recognition)*. Використання характеристик голосу для ідентифікації осіб. Алгоритми глибокого навчання можуть використовуватися для ефективного розпізнавання голосу.

4. *Аналіз поведінки*. Системи можуть аналізувати унікальні аспекти поведінки осіб, такі як стиль ходьби або манери носіння мобільного пристрою.

Ці методи можуть використовуватися окремо або комбінуватися для створення більш ефективних та надійних систем інтелектуального контролю доступу. Кожен з методів має свої переваги та обмеження, і вибір конкретного підходу залежить від вимог конкретного застосування та рівня безпеки, який потрібно забезпечити.

2.2. Розробка математичного забезпечення інтелектуальної пропускної системи

Розробка математичного забезпечення інтелектуальної пропускної системи включає в себе створення комп'ютерних алгоритмів та програм, які допомагають автоматизувати та оптимізувати процес контролю доступу до певної території або об'єкта. Тут слід звернути увагу, що це залежить від будь-якого місця, де необхідно визначити, хто має право на доступ, а хто - ні.

Математичне забезпечення такої системи включає в себе наступні аспекти:

1. *Аутентифікація і авторизація*: розробка алгоритмів, що дозволяють перевірити ідентифікацію особи, яка намагається отримати доступ, і визначити її права доступу.

2. *Біометрична ідентифікація*: використання біометричних даних (відбитки пальців, розпізнавання обличчя тощо) для точної ідентифікації особи.

3. *Аналіз поведінки*: розробка алгоритмів для визначення звичайної поведінки користувачів та виявлення аномальних дій.

4. *Криптографія*: забезпечення безпеки даних та комунікацій шляхом використання криптографічних методів.

5. *Математичні моделі*: створення математичних моделей, які допомагають прогнозувати ризики і оптимізувати ресурси для ефективного управління доступом.

6. *Аналітика даних*: розробка інструментів для обробки та аналізу даних про доступи з метою виявлення трендів, незвичайних подій або вразливостей.

7. *Автоматизація процесів*: розробка програмних рішень, які допомагають автоматично керувати доступом на основі заздалегідь встановлених правил.

8. *Штучний інтелект*: використання технологій штучного інтелекту для покращення точності і швидкості ідентифікації, а також для навчання системи розпізнавати нові ситуації.

9. *Оптимізація ресурсів*: розробка алгоритмів, що дозволяють раціонально використовувати ресурси системи для оптимального розподілу прав доступу.

10. *Тестування та валідація*: виконання тестів для перевірки правильності та надійності розроблених алгоритмів та програм.

Тут варто зазначити, що дана розробка вимагає спеціалізованих знань у

галузях криптографії, математичного моделювання, обробки даних, програмування та безпеки інформації.

Для забезпечення безпеки та контролю доступу, система має розробити методи автентифікації, які підтверджують ідентичність користувачів, такі як паролі, біометричні дані, смарт-карти тощо. Однак це лише один з перших кроків. Залежно від того яка система, вона повинна вирішити які ресурси і дії можуть виконувати різні користувачі.

Використання унікальних біометричних даних дозволяє більш точно та надійно ідентифікувати осіб. Алгоритми розпізнавання облич, відбитків пальців, радужок ока та інших біометричних параметрів дозволяють системі швидко порівнювати ідентифікаційні дані зі збереженими у базі даних.

Розробка алгоритмів, що аналізують звичайну поведінку користувачів, допомагає виявляти незвичайні, відхилені вчинки. Наприклад, система може виявити підозрілу активність, яка може бути зв'язана зі зловмисними діями.

Також, центральний аспект у забезпеченні безпеки інформації. Використання криптографічних методів допомагає шифрувати дані, підтверджувати цілісність і забезпечувати конфіденційність під час передачі даних між компонентами системи.

Розробка математичних моделей дозволяє прогнозувати ризики та оптимізувати ресурси. Наприклад, можуть бути розроблені моделі, що оцінюють ймовірність незаконного доступу або аналізують патерни активності для передбачення можливих загроз.

Беручи до уваги аналітику даних, то зібрані дані про доступи можуть бути оброблені для виявлення незвичайних трендів або подій, що потребують уваги. Це допомагає операторам системи реагувати на можливі загрози швидше та ефективніше.

Розробка програмних рішень для автоматичного керування доступом дозволяє зменшити людський фактор та оптимізувати ресурси. Система може автоматично надавати доступ, обмежувати його на певний час або виконувати інші дії на основі заздалегідь встановлених правил.

Окрім цього, застосування методів штучного інтелекту дозволяє системі навчатися на основі нових даних та адаптуватися до змінних умов. Наприклад, можуть бути розроблені моделі машинного навчання для вдосконалення розпізнавання обличчя або для виявлення аномальної активності.

При оптимізації ресурсів розробка алгоритмів для їх ефективного розподілу допомагає системі працювати швидко та ефективно, забезпечуючи максимально можливий рівень безпеки.

Додаючи до всього вище згаданого, ретельне тестування розроблених алгоритмів та програм є важливим етапом, що допомагає виявити можливі помилки та допрацювати систему до оптимальної робочої версії.

Розробка математичного забезпечення інтелектуальної пропускної системи є складним та багатогранним процесом, який об'єднує знання з різних галузей, таких як комп'ютерна безпека, криптографія, біометрія, штучний інтелект та інші вище згадані галузі. Це вимагає глибокого розуміння технічних аспектів, математичних принципів та інноваційних методів.

Для успішної розробки такої системи необхідно вирішити завдання з автентифікації та авторизації, використовуючи біометричні дані та криптографічні методи. Також важливо розробити алгоритми аналізу поведінки, які допомагають виявити аномалії та підозрілі дії. Використання штучного інтелекту та методів машинного навчання може покращити точність і ефективність системи.

Математичні моделі дозволяють прогнозувати ризики та оптимізувати ресурси, що є важливим для забезпечення безпеки та ефективності. Автоматизація процесів та використання технологій інтелектуального аналізу даних допомагають управляти системою більш ефективно.

Однак, розробка такої системи також пов'язана з великими викликами, такими як забезпечення конфіденційності, захист від кібератак та врахування етичних аспектів, пов'язаних зі збереженням інформації про користувачів.

Загалом, розробка математичного забезпечення інтелектуальної пропускної системи вимагає глибокого розуміння технічних, математичних та бізнес-аспектів, а також уважного підходу до забезпечення безпеки, конфіденційності та ефективності системи.

Варто звернути увагу, що процес розроблення математичного забезпечення має забезпечити шлях від усвідомлення потреб програми до його експлуатації. Він складається з таких етапів (див. рис. 2.6):

Рис. 2.6. *Етапи процесу розроблення математичного забезпечення та його підготовка до експлуатації.*

- Визначення вимог – збір та аналіз вимог проектування системи та подання їх у нотації;
- Проектування – перетворення вимог до розроблення у послідовність проектних рішень щодо способів реалізації вимог: формування загальної архітектури програмної системи та принципів її прив'язки до конкретного середовища функціонування; визначення детального складу модулів кожної з архітектурних компонент;
- Реалізація – перетворення проектних рішень у програмну систему, що реалізує означені рішення;

- Тестування – перевірка кожного з модулів та способів їх інтеграції; тестування програмного продукту в цілому (так звана верифікація); тестування відповідності функцій працюючої програмної системи вимогам, що були до неї поставлені замовником (так звана валідація);
- Експлуатація та супроводження готової системи.

Підготовча робота починається з вибору моделі для математичного забезпечення, що відповідає масштабові, значимості і складності проекту. Процес розроблення має відповідати обраній моделі. Розробник повинен вибрати, адаптувати до умов проекту і використовувати погоджені стандарти, методи й засоби розробки, а також скласти план виконання робіт.

Аналіз вимог до системи розглядає функціональні можливості, вимоги користувача, вимоги до надійності і безпеки, вимоги до зовнішніх інтерфейсів тощо. Вимоги до системи оцінюються відповідно до критеріїв реалізації і можливості перевірки при тестуванні. Проектування архітектури системи полягає у визначенні компонентів її устаткування, ПЗ й операцій, що виконуються персоналом.

Аналіз вимог до ПЗ визначає: функціональні можливості, включаючи характеристики продуктивності і середовища функціонування компонента; зовнішні інтерфейси; специфікації надійності і безпеки; ергономічні вимоги; вимоги до даних; вимоги до інсталяції та введення системи; вимоги до документації користувачів; вимоги до експлуатації і супроводу.

Проектування архітектури математичного забезпечення разом з програмним забезпеченням включає такі задачі (для кожного компонента):

1. Трансформацію вимог в архітектуру проектування, що визначає структуру і склад його компонентів.
2. Розроблення і документування програмних інтерфейсів проектування і бази даних.
3. Розроблення попередньої версії документації щодо даних сканованих осіб/предметів.

4. Розроблення і документування попередніх вимог до тестів і плану інтеграції проектування.

Детальне проектування математичного забезпечення включає такі задачі:

1. Опис компонентів проектування й інтерфейсів між ними на нижчому рівні, що достатній для їх подальшого самостійного кодування і тестування.

2. Розроблення і документування детального проекту бази даних.

3. Відновлення (за необхідності) документації.

4. Розроблення і документування вимог до тестів і плану тестування компонентів на фінальній стадії проектування.

5. Відновлення плану інтеграції проектування.

Кодування і тестування ПЗ охоплюють такі задачі:

1. Розроблення (кодування) і документування кожного компонента під час проектування і бази даних, а також сукупності тестових процедур і даних для їхнього тестування.

2. Тестування кожного компонента готової моделі і її бази даних на відповідність вимогам. Результати тестування компонентів мають бути документовані.

3. Відновлення (за необхідності) документації користувачів.

4. Відновлення плану інтеграції готової моделі.

Інтеграція моделі та її проектування передбачає збирання розроблених компонентів відповідно до плану інтеграції і тестування компонентів. Для кожного з компонентів розробляються набори тестів і тестові процедури, що призначені для перевірки кваліфікаційних вимог при наступному кваліфікаційному тестуванні. Кваліфікаційна вимога – це набір критеріїв або умов, який необхідно виконати, щоб кваліфікувати програмний продукт на відповідність своїм специфікаціям і можливість його використовувати в умовах експлуатації.

Кваліфікаційне тестування проводиться розробником для впевненості

того, що ПЗ дійсно відповідає своїм специфікаціям. Кваліфікаційне тестування здійснюється для кожного компонента моделі щодо всіх вимог при використанні різних тестів. При цьому також перевіряються повнота технічної документації та її адекватність самим компонентам моделі під час її проектування.

Таким чином, математичне забезпечення інтелектуальної пропускної системи грає важливу роль у її функціонуванні та надійності. Завдяки математичному забезпеченню інтелектуальна пропускна система може ефективно розпізнавати та ідентифікувати осіб, забезпечувати безпеку даних та керувати доступом до об'єктів або приміщень. Математичні моделі та алгоритми грають важливу роль у забезпеченні точності та надійності цієї системи.

Опираючись на дані з даного розділу, математична модель у візуальному образі може бути побудована за допомогою **з'єднання схем-блоків.** Коли блоки з'єднані послідовно один з одним, їх можна спростити до одного блоку, перемноживши декілька передатних функцій, які відповідають за математичне забезпечення системи. Якщо ж схема складніша, з кількома ділянками зворотного зв'язку, то для спрощення необхідна покрокова перестановка, як показано в прикладі нижче. Ви можете бачити, що шляхом множення або ділення передаточних функцій ми можемо переміститися в системі туди, куди вказує стрілка (див. рис. 2.7):

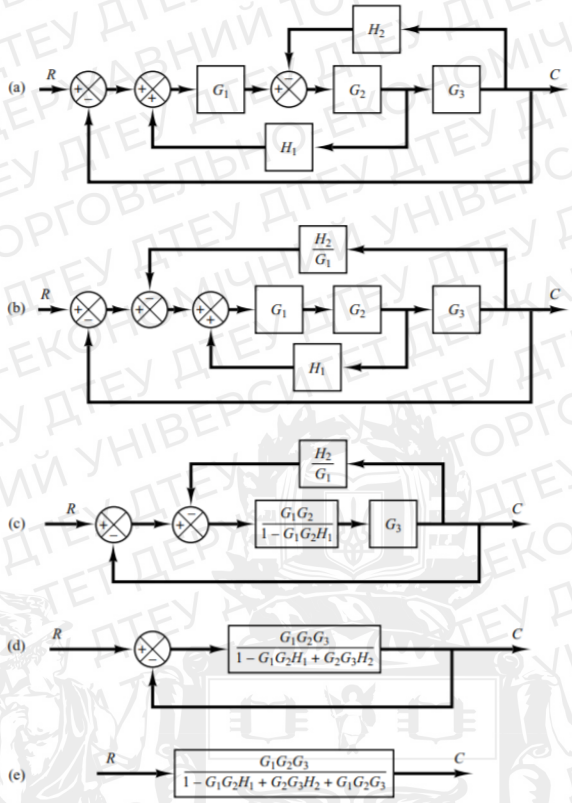

Рис. 2.7. *Візуальна модель математичного забезпечення для інтелектуальної пропускної системи.*

В інтелектуальній пропускній системі мають бути **точки підсумовування.** Як видно на малюнку нижче (рис. 2.8), коло з хрестиком - це символ, який позначає операцію підсумовування. Знак "плюс" або "мінус" біля кожної стрілки вказує, чи потрібно додавати або віднімати сигнал для отримання результату:

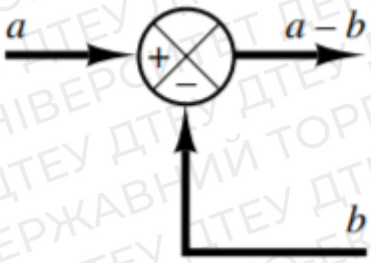

Рис. 2.8. *Точки підсумовування інтелектуальної пропускної системи.*

Важливу роль в реалізації візуальної математичної моделі пропускної системи має **точка розгалуження**, яка відрізняється від точки підсумовування тим, що це точка, з якої сигнал від блоку йде одночасно до інших блоків або точок підсумовування (див. рис. 2.9):

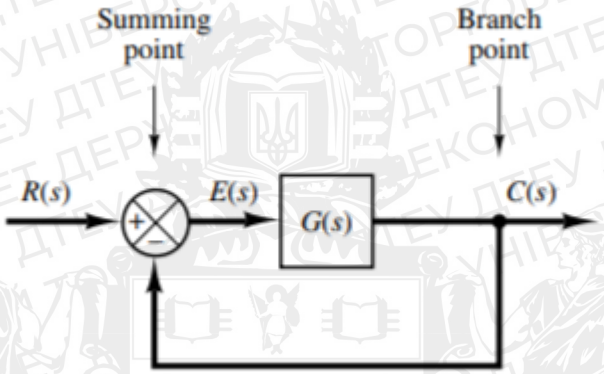

Рис. 2.9. *Точка розгалуження в математичній моделі інтелектуальної пропускної системи.*

2.3. Розробка інформаційного забезпечення інтелектуальної пропускної системи

Як було згадано раніше в попередніх розділах, інтелектуальна пропускна система - це сучасне рішення, яке дозволяє автоматизувати та полегшити процес контролю доступу на певну територію або в приміщення. Вона використовує різноманітні технології для ідентифікації осіб та/або предметів, що мають права входу, і забезпечує оптимальний рівень безпеки.

Але варто зазначити, що розробка інформаційного забезпечення інтелектуальної пропускної системи вимагає співпраці спеціалістів з різних галузей, таких як програмування, мережеві технології, кібербезпека, аналітика та багато інших. Така система може бути використана в різних

сферах, таких як офісні будівлі, медичні заклади, промислові об'єкти, транспортні вузли тощо, для забезпечення ефективного та безпечного контролю доступу.

Оскільки розробка інформаційного забезпечення інтелектуальної пропускної системи є складним та багатоетапним процесом, який включає в себе наступні кроки:

1. *Аналіз вимог*: перший етап - це збір і аналіз вимог. Команда розробників спільно зі замовником визначає, які функції має виконувати система, які методи ідентифікації будуть використовуватися, які зони потребуватимуть контролю доступу та інші ключові деталі.

2. *Проектування архітектури*: на цьому етапі розробляється загальна архітектура системи. Вирішується питання про апаратне та програмне забезпечення, системну архітектуру, розташування пристроїв, способи зберігання даних тощо.

3. *Розробка програмного забезпечення (далі ПЗ)*: розробники створюють програмне забезпечення для інтелектуальної пропускної системи. Це може бути веб-додаток, мобільний додаток або комбінація обох. Реалізуються функції ідентифікації, керування доступом, зберігання даних та інші необхідні функції.

4. *Розробка бази даних*: створюється база даних, де будуть зберігатися дані про користувачів, розклад роботи, права доступу та інша важлива інформація.

5. *Розробка інтерфейсу користувача*: Розробляється інтерфейс, через який оператори системи та користувачі зможуть керувати системою, переглядати статистику, налаштовувати параметри тощо.

6. *Інтеграція апаратного забезпечення*: Встановлюються та налаштовуються всі необхідні пристрої, такі як камери, сканери відбитків пальців, картрідери тощо. Ці пристрої повинні бути підключені до системи та інтегровані з програмним забезпеченням.

7. *Тестування*: Після розробки виконується тестування системи на

різних сценаріях. Виявлені помилки та проблеми виправляються.

8. *Впровадження та навчання*: Після успішного тестування система готова до впровадження. Виконується встановлення та налаштування системи на реальному об'єкті. Оператори та користувачі проходять навчання з використання системи;

9. *Підтримка та вдосконалення*: Після впровадження команда продовжує підтримку системи, вирішує виникаючі питання та вдосконалює систему з урахуванням відгуків користувачів.

Цей процес може варіюватися в залежності від конкретних вимог та контексту розробки. Важливою частиною розробки інформаційного забезпечення є забезпечення безпеки даних та захисту системи від несанкціонованого доступу. Також слід звернути увагу на відповідність системи нормативним вимогам та законодавству, особливо в контексті обробки персональних даних.

Слід зазначити, що кожний процес має безліч особливостей, від якого залежить майбутня функціональність інтелектуальної пропускної системи на основі математичного забезпечення.

На етапі аналізу вимог команда розробників спільно з замовником (клієнтом) визначає всі вимоги до системи. Це може включати:

- список функцій, які повинна виконувати система (ідентифікація, реєстрація, контроль доступу, зберігання даних тощо).
- специфікації апаратного забезпечення (типи камер, датчиків, картрідерів тощо).
- вимоги до безпеки (шифрування даних, захист від вторгнень).
- вимоги до інтерфейсу користувача.

На етапі проектування визначається загальна структура системи. Розробники вирішують, які компоненти будуть включені до системи, як вони взаємодіятимуть, як будуть розподілятися функції між серверами та клієнтами, як забезпечити масштабованість та надійність системи.

На етапі розробки ПЗ розробники пишуть код для програмного забезпечення системи. Вони реалізовують функції ідентифікації (розпізнавання обличчя, сканування відбитків пальців), алгоритми керування доступом, засоби зберігання даних та інші необхідні функції.

При розробці баз даних створюється відповідна база даних для зберігання інформації про користувачів, права доступу, розклад роботи тощо. Важливо враховувати вимоги до безпеки та ефективності зберігання даних.
При розробці інтерфейсу створюється також відповідний інтерфейс, через який оператори та користувачі можуть взаємодіяти з системою. Це може бути веб-інтерфейс, додаток для смартфона або інше.

На етапі апаратного забезпечення встановлюються та налаштовуються не тільки пристрої описані вище, які будуть використовуватися в системі. Це можуть бути звукові детектори, сигнали подачі звуку тривоги, штучний інтелект який повідомляє про підозрілу активність.

Також система піддається ретельному тестуванню на різних сценаріях використання. Виявлені помилки та недоліки мають бути обов’язково виправлені перед впровадженням.

Після успішного тестування система готова до впровадження, де виконується встановлення та налаштування системи на вже існуючих реальних об'єктах. Оператори та користувачі проходять навчання з використання системи.

Після впровадження команда продовжує підтримувати систему, вирішує виникаючі питання та вдосконалює систему на основі зворотного зв'язку користувачів (відгуки в соціальних мережах, СМС або службу підтримки).

Слід зазначити, що всі ці процеси є ітеративними і в теорії вони можуть мати багато різних командних робіт залежно від їх функціоналу, тобто під час розробки можуть виникати нові вимоги, помилки чи зміни, які потребують коригування. Уважне планування, спілкування між командами розробників та замовником, а також дотримання стандартів безпеки та якості

є ключовими для успішної розробки інформаційного забезпечення інтелектуальної пропускної системи.

Розробка інтелектуальної пропускної системи включає роботу зі змінами та конфігурацією. Конфігураційний менеджмент допомагає керувати різними версіями системи, змінами у програмному коді та налаштуваннях. Конфігурація в контексті розробки інформаційного забезпечення інтелектуальних пропускних систем відноситься до налаштування та управління параметрами системи та її компонентів. Даний процес допомагає забезпечити належну роботу системи та забезпечити відповідність вимогам замовника.

На малюнку нижче зазначений загальновживаний код, який використовують для створення програмного забезпечення для інтелектуальної пропускної системи використовуючи мову програмування Python та її бібліотеки для роботи з даними (див. рис. 2.10):

```
from cv2 import cv2
import clx.xms
import requests
client = clx.xms.Client(service_plan_id='your_service id', token='token_id')
create = clx.xms.api.MtBatchTextSmsCreate()
create.sender = 'sender no.'
create.recipients = {'recipients no.'}
create.body = 'This is a test message from your Sinch account'
detector = cv2.CascadeClassifier("path")
cap = cv2.VideoCapture(0, cv2.CAP_DSHOW)
counter = 0
while True:
    ret, img = cap.read()
    if ret:
        gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
        faces = detector.detectMultiScale(gray, 1.1, 4)
        for face in faces:
            x, y, w, h = face
            if(face.any() and counter ==0):
                try:
                batch = client.create_batch(create)
                except (requests.exceptions.RequestException, clx.xms.exceptions.ApiException) as ex:
                    print('Failed to communicate with XMS: %s' % str(ex))
            cv2.rectangle(img, (x, y), (x+w, y+h), (255, 0, 0), 2)
        cv2.imshow("Face", img)
        counter = 1
    key = cv2.waitKey(1)
    if key == ord("q"):
        break
cap.release()
cv2.destroyAllWindows()
```

Рис. 2.10. *Приклад створення програмного забезпечення для інтелектуальної*

пропускної системи за допомогою мови програмування Python та її фреймворків.

Після створення програмного забезпечення разом з кодом, необхідно провести його тест. Але варто зазначити, що будь-який член команди може в будь-який момент змінити будь-яку частину системи. В загальному, існує тільки одна офіційна версія розроблюваної системи. Якщо знадобиться створити для чогось її гілку, варто залишити її лише на кілька годин.

Щоразу треба збирати й тестувати нову версію системи і вводити її в дію. Чим більший розрив між офіційною версією системи і тієї, що знаходиться на комп'ютері розробника ПЗ або тестувальника, тим більші ризики і тим дорожче це для проекту.

На наступному малюнку продемонстровано як поетапне планування, створення ПЗ та тестування інтелектуальної пропускної системи працюють разом. Замовники люблять бути партнерами в процесі розроблення програмного забезпечення і активно сприяти незалежно від рівня досвіду, а менеджери мають бути зосередженими на зв'язках і відношеннях (див. рис. 2.11):

Рис. 2.11. *Погодження розробників та замовників щодо створення*

програмного забезпечення для інтелектуальної пропускної системи.

Для організації інформаційної взаємодії різноманітних інформаційних систем між собою, а також з різними групами користувачів дані потрібно відповідним чином однотипово описати в усіх системах на різних рівнях, тобто вирішити проблему їх інформаційної сумісності в найширшому розумінні. Цього досягають створенням інформаційного забезпечення, під яким розуміють сукупність форм документів, нормативної бази та реалізованих рішень щодо обсягів, розміщення і форм існування інформації, яка використовується в інформаційній системі при її функціонуванні.

Методичні та інструктивні матеріали – це сукупність державних стандартів, галузевих керівних методичних матеріалів і розроблених проектних рішень щодо створення й супроводження інформаційного забезпечення.

Системи класифікації і кодування – це перелік описів і систем супроводження класифікаторів техніко-економічної інформації на економічному об’єкті.

Більш детальну інформацію щодо структури інформаційного забезпечення інтелектуальної пропускної системи, стосовно вищеописаних пунктів, продемонстровано на малюнку нижче (див. рис. 2.12):

Рис. 2.12. *Структура інформаційного забезпечення інтелектуальної*

пропускної системи.

Основні принципи створення інформаційного забезпечення: цілісність, вірогідність, контроль, захист від несанкціонованого доступу, єдність і гнучкість, стандартизація та уніфікація, адаптивність, мінімізація введення і виведення інформації (однократність введення інформації, принцип введення – виведення тільки змін).

Цілісність – здатність даних задовольняти принцип повного узгодження, точність, доступність і достовірне відображення реального стану об'єкта.

Існують два підходи до створення інформаційного забезпечення:

- аналіз сутностей;
- синтез атрибутів.

Аналіз сутностей є спадним підходом, або «згори – вниз», який поділяє процес створення на чотири стадії:

- моделювання уявлень користувачів;
- об'єднання уявлень;
- складання і аналіз моделі (схеми);
- реальне (фізичне) проектування.

Синтез атрибутів є зростаючим підходом, або «знизу – вгору», оскільки він починається із синтезу атрибутів найнижчого рівня, з яких формуються сутності та зв'язки верхнього рівня. Виділяють чотири стадії для цього підходу:

- класифікація атрибутів;

- композиція сутностей;
- формування зв’язків;
- графічне уявлення.

Кожний з цих підходів має свої переваги й недоліки і визначається виходячи із потреб проектування інформаційного забезпечення. Для створення великих інформаційних забезпечень, у яких є структура, найбільш прийнятний аналіз сутностей, для автономних невеликих інформаційних забезпечень без структури – атрибутний (локальний).

Інформаційне забезпечення не можна успішно спроектувати без загального планування «згори – вниз» і детального проектування «знизу – вгору». Погодження двох підходів, в свою чергу, не можна досягти без відповідної методики, загальні аспекти якої ми розглядаємо.

Вимоги до інформаційного забезпечення (ГОСТ 24.104–85 «Автоматизовані системи управління. Основні вимоги») такі:

1. Інформаційне забезпечення має бути достатнім для виконання всіх функцій інформаційного забезпечення, які автоматизуються.
2. Для кодування інформації, яка використовується тільки в цій системі, мають бути застосовані класифікатори, які є у замовника.
3. Для кодування в інформаційному забезпеченні вихідної інформації, яка використовується на вищому рівні, мають бути використані класифікатори цього рівня, крім спеціально обумовлених випадків.
4. Інформаційне забезпечення має бути суміщене з інформаційним забезпеченням систем, які взаємодіють з нею, за змістом, системою кодування, методами адресації, форматами даних і формами подання інформації, яка отримується і видається інформаційною системою.
5. Форми документів, які створюються інформаційною системою, мають відповідати вимогам стандартів УСД чи нормативно-технічним документам замовника інформаційного забезпечення.

6. Форми документів і відеокадрів, які вводяться, виводяться чи коригуються через термінали інформаційного забезпечення, мають бути погоджені з відповідними технічними характеристиками терміналів.
7. Сукупність інформаційних масивів інформаційного забезпечення має бути організована у вигляді бази даних на машинних носіях.
8. Форми подання вихідної інформації інформаційного забезпечення мають бути погоджені із замовником (користувачем) системи.
9. Терміни і скорочення, які застосовуються у вихідних повідомленнях, мають бути загальноприйнятими в цій проблемній сфері й погоджені із замовником системи.
10. У інформаційному забезпеченні мають бути передбачені необхідні заходи щодо контролю і оновлення даних в інформаційних масивах, оновлення масивів після відмови будь-яких технічних засобів, а також контролю ідентичності однойменної інформації в базах даних.

Ефективне функціонування інформаційної системи об'єкта можливе лише при відповідній організації інформаційної бази – сукупності впорядкованої інформації, яка використовується при функціонуванні інформаційного забезпечення і поділяється на ***зонішньо-*** і ***внутрішньомашинну (машинну) бази*** (ГОСТ 34.003–90).

Зовнішньомашинна інформаційна база – частина інформаційної бази, яка являє собою сукупність повідомлень, сигналів і документів, призначених для безпосереднього сприйняття людиною без застосування засобів обчислювальної техніки.

Внутрішньомашинна інформаційна база – частина інформаційної бази, яка є сукупністю інформації, що використовується в ІС на носіях даних.

Така зовнішньомашинна ІБ має багато модифікацій від подання у вигляді повідомлень на паперовому носії, запитів на екрані дисплея та домовного спілкування з ЕОМ.

Внутрішньомашинна ІБ пройшла три етапи еволюції.

Перший етап характеризується роз'єднаним фондом даних:

- Програми розв'язання кожної окремої задачі становили одне ціле з масивами, які оброблялися.
- Використання якого-небудь масиву для іншої задачі забезпечувалось індивідуально пристосуванням до форм подання даних, структур елементів масивів і т ін.
- Опис даних не потрібний, оскільки структура була раніше відома.
- Коригування масивів виконувалось індивідуальними засобами.
- Задача розв'язувалася в пакетному режимі, користувач отримував результати винятково у вигляді машинограм і виробничих документів через групу підготовки і оформлення даних.

Другий етап – централізований фонд даних:

- Дані відокремлені від процедур їх обробки і організовані в бібліотеки масивів загального користування. Подання інформації, формати елементів даних і структура масивів уніфіковані і не залежать від конфігурації пам'яті та її організації.
- Опис даних відокремлено як від програм, так і від самих даних, тому дані й програми їх обробки стають значною мірою незалежними. Це полегшує зміну структур даних і програм. Але реорганізація бібліотеки і її окремих груп компонентів потребує зміни програми обробки.

Третій етап – організація баз даних – характеризується:

- Об'єднання не лише інформації, а й апаратно-програмних засобів її поповнення, коригування і видачі користувачеві.
- Повне відокремленням функцій нагромадження, ведення і реорганізації

даних від функцій їх обробки. Дані коригуються поза рівнем програм користувача за допомогою власного апарату бази даних.

- Поява логічного буфера, системи управління базою даних, розв'язки між програмами користувача і базою даних.
- Можливість оперативної реалізації довільних запитів у режимі безпосереднього зв'язку з ЕОМ.
- Високий ступень централізації загальносистемних масивів, яка передбачає спільне використання загальних даних.
- Різноманітність даних і зв'язаність в довільні логічні структури.
- Наявність потужного програмного забезпечення і мовних засобів.

Основною задачею є визначення потрібної кількості баз даних і оптимального розподілу інформації між ними з урахуванням того, що **економічний об'єкт** – це динамічна система, яка перебуває в постійному розвитку. Використовуючи пріоритет виробничих функцій, необхідно побудувати таку базу даних для інтелектуальної пропускної системи. Так, навколо поняття «Модель виробу» формуються дві оболонки: внутрішня являє конструкторську документацію, зовнішня – технологічну і управлінську інформацію. Більше інформації зображено на малюнку нижче (див. рис. 2.13):

Рис. 2.13. *Модель виробу з внутрішньою та зовнішньою оболонками.*

Однак варто зауважити, що виникла виникла наступна проблема: визначити, чи потрібна одна база даних, кілька локальних, взаємозв'язана розподілена база даних, локальні файли чи їх комбінації і т.п. При цьому враховується інформація, що використовується для реалізації багатьох функцій, особливо в оперативному режимі, активна інформація, тобто така, що використовується багаторазово.

Описуючи організацію інформаційної бази для інтелектуальної пропускної системи, потрібно дати опис логічної і фізичної структур бази даних.

Документ складається з двох частин:

- Опис внутрішньомашинної інформаційної бази.
- Опис зовнішньомашинної інформаційної бази.

Кожна частина складається з таких розділів:

- Логічна структура.
- Фізична структура (для зовнішньомашинної інформаційної бази).
- Організація ведення інформаційної бази.

При організації раціонального варіанта внутрішньомашинної інформаційної бази даних для інтелектуальної пропускної системи, яка найбільш повно відбиває специфіку об'єкта управління, перед розробниками постають вимоги до організації масивів, які можуть бути суперечливими. До них належать:

- Повнота подання даних.
- Мінімальний склад даних.
- Мінімізація часу вибірки даних.
- Незалежність структури масивів від програмних засобів їх організації.
- Динамічність структури інформаційної бази.

Найбільш суперечливою з них є вимога повноти подання даних, мінімізація складу даних і мінімізація часу вибірки даних. Оптимальним є повне взаємне врахування всіх вимог, що випливають з процесів, які автоматизуються.

Останнім часом склалися такі основні підходи до побудови внутрішньомашинної інформаційної бази:

- Проектування масиву як відображення змісту окремого документа.
- Проектування масивів для окремих процесів управління.
- Проектування масивів для комплексів процесів управління, які реалізуються.
- Проектування бази даних.
- Проектування кількох баз даних.

Кожний з цих підходів має свої переваги і недоліки, а вибір залежить від обчислювальної техніки, яка використовується, програмних засобів і специфіки процесів, що автоматизуються.

Проектування масивів інформаційного забезпечення для інтелектуальної пропускної системи передбачає визначення їх складу, змісту, структури і вибір раціонального способу їх подання в пам'яті обчислювальної системи.

Поняття складу і змісту масивів передбачає визначення оптимальної кількості масивів і переліку атрибутів (полів), які у них містяться.

Під структурою масиву розуміємо формат записів у масиві, розмір полів і їх розміщення в машинному записі, ключові атрибути і впорядкування масиву за ними.

Вибираючи раціональний спосіб подання масиву в пам'яті визначають такий спосіб зберігання даних, за якого забезпечувалися б мінімальний обсяг пам'яті для розміщення масиву, висока швидкість пошуку даних, а також

можливість збільшення і оновлення масиву. Кожний масив

характеризується обсягом, способом організації, стабільністю і ступенем активності.

З точки зору використання масивів на різних етапах технологічного процесу обробки даних виділяють такі типи масивів: вхідні (первинні), основні (базові), робочі (проміжні) й вихідні (результатні).

Вхідні масиви – це проміжна ланка між первинними інформаційними повідомленнями і основними масивами. Зміст і розміщення даних у вхідному масиві аналогічні змісту й розміщенню їх у первинному інформаційному повідомленні.

Основні масиви створюються на основі вхідних, постійно зберігаються і містять основні дані про об'єкти управління і процеси виробництва. Кожний основний масив містить усю сукупність інформації, яка всебічно характеризує однорідні об'єкти і потрібна для реалізації функцій управління. За змістом ці масиви ми можемо класифікувати на такі групи: нормативні, розцінoчні, планово-договірні, регламентуючі, довідково-табличні й постійно-облікові.

Необхідність створення таких масивів зумовлена необхідністю забезпечення принципу одноразового формування масивів, внесення змін і усунення дублювання. Це в свою чергу призводить до різкого збільшення його розміру і ускладнення використання в процесі реалізації тих чи інших процесів, оскільки часто потрібна лише частина інформації основного масиву, а це вимагає створення робочих масивів.

Робочі масиви призначені для роботи програм, які реалізують розв'язання конкретних задач процесів управління і містять обмежене коло атрибутів одного чи кількох основних масивів. Робочі масиви організуються в момент розв'язання задачі й лише на час її розв'язання, після чого їх анулюють.

Вихідні масиви формуються в процесі розв'язання задачі й використовуються для модифікації основних масивів і виведення вихідних (результатних) інформаційних повідомлень.

Основні масиви можуть мати вигляд локальних масивів чи організовані в базу даних (БД) під керуванням системи управління базою даних (СУБД).

Взаємозв'язок користувача з базою даних інтелектуальної пропускної системи зображено на малюнку нижче (див. рис. 2.14):

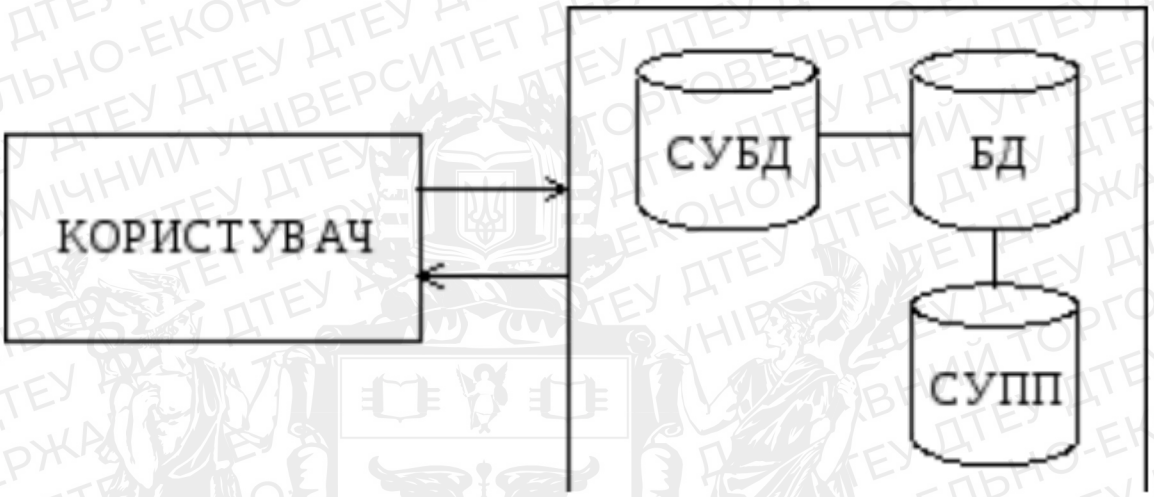

Рис. 2.14. *Схема взаємозв'язку користувача з базою даних інтелектуальної пропускної системи.*

База даних – іменована сукупність даних, що відображає стан об`єктів та їх відношення у визначеній проблемній сфери (закон України "Про Національну програму інформатизації" (74/98-ВР від 04.02.98)).

Система управління базами даних – це сукупність програм і мовних засобів, які призначені для управління даними в базі даних і забезпечують взаємодію її з прикладними програмами.

Масив даних – це конструкція даних, компоненти якої ідентичні за своїми характеристиками і є значеннями функції від фіксованої кількості цілочислових аргументів.

Файл – це ідентифікована сукупність примірників повністю описаного в конкретній програмі типу даних, розміщених зовні програми в зовнішній пам'яті та доступних програмі, за допомогою спеціальних операцій.

На наступних малюнках (рис. 2.15-2.16) я створив базу даних для своєї інтелектуальної пропускної системи яка має інформаційне забезпечення:

```python
import sqlite3

# Підключення до бази даних (створюється, якщо не існує)
conn = sqlite3.connect('access_control_system.db')
cursor = conn.cursor()

# Створення таблиці для користувачів
cursor.execute('''
    CREATE TABLE IF NOT EXISTS users (
        id INTEGER PRIMARY KEY,
        first_name TEXT NOT NULL,
        last_name TEXT NOT NULL,
        access_level INTEGER NOT NULL
    )
''')

# Створення таблиці для подій доступу
cursor.execute('''
    CREATE TABLE IF NOT EXISTS access_events (
        id INTEGER PRIMARY KEY,
        user_id INTEGER,
        timestamp DATETIME DEFAULT CURRENT_TIMESTAMP,
        success BOOLEAN,
        FOREIGN KEY (user_id) REFERENCES users(id)
    )
''')
```

Рис 2.15. *Створення бази даних інформаційного забезпечення інтелектуальної пропускної системи*.

```
# Додавання прикладу користувача
cursor.execute( _sql: '''
    INSERT INTO users (first_name, last_name, access_level) VALUES (?, ?, ?)
''', _parameters: ('John', 'Doe', 1))

# Додавання прикладу події доступу
cursor.execute( _sql: '''
    INSERT INTO access_events (user_id, success) VALUES (?, ?)
''', _parameters: (1, True))

# Збереження змін та закриття підключення
conn.commit()
conn.close()
```

Рис. 2.16. *Створення бази даних для інформаційного забезпечення інтелектуальної пропускної системи.*

Таким чином, я створив локальну базу даних для ефективного інформаційного забезпечення. Я створив її використовуючи мову програмування Python і базу даних SQLite, оскільки це легка та вбудована СКБД, яка добре підходить для невеликих проектів. Для взаємодії з базою даних на Python, можна використовувати бібліотеку sqlite3. Весь програмний код був створений в окремому файлі під назвою **database.py** в папці проекту.

Висновки до розділу

Проектування інтелектуальної пропускної системи - це комплексний процес, який вимагає детального підходу та знань з різних областей, таких як інформаційна технологія, ідентифікація, безпека та бізнес-процеси. Воно вимагає об'єднання технічних, безпекових та організаційних аспектів. Успіх залежить від тісного співробітництва між різними командами та спеціалістами.

Моделювання системи перед її фактичною розробкою допомагає визначити потреби, пріоритети та можливі ризики. Аналіз потреб користувачів дозволяє створити систему, яка справді задовольняє їх вимоги.

Використання математичних моделей допомагає прогнозувати роботу системи, оцінювати продуктивність та виявляти можливі проблеми на етапі проектування, і також розробка інформаційного забезпечення важлива також, адже вона включає в себе розробку баз даних, розробку інтерфейсів користувача та забезпечення захисту даних. Важливо збалансувати функціональність та безпеку.

Безпека є ключовим аспектом інтелектуальних пропускних систем. Забезпечення захисту даних, управління доступом та виявлення вторгнень важливо для забезпечення довіри користувачів до системи.

Загалом, проектування та розробка інтелектуальної пропускної системи - це велике та складне завдання, яке вимагає збалансованого підходу до технічних, безпекових та бізнес-аспектів. Детальне моделювання, математичне та інформаційне забезпечення допомагають забезпечити ефективну та безпечну роботу системи з урахуванням потреб користувачів та бізнес-вимог.

Інтелектуальна пропускна система без пунктів - це система безпеки, яка використовує різні технології для ідентифікації та контролю доступу без необхідності фізичного пункту входу.

Моделювання інтелектуальної пропускної системи без пунктів передбачає поєднання різних технологій та методів для створення ефективної та безпечної системи контролю доступу без необхідності фізичних точок входу.

Математичне забезпечення інтелектуальної пропускної системи без пунктів грає критичну роль у забезпеченні безпеки та ефективності системи. Вона може містити використання математичних моделей для розпізнавання облич, біометричних даних (відбитків пальців, рис та ін.) та інших об'єктів для ідентифікації осіб чи предметів. Також вона може містити використання

математичних алгоритмів у машинному навчанні для навчання системи розпізнавання на основі даних та для постійного вдосконалення її роботи.

В цілому, математичне забезпечення системи безпунктової безпеки базується на різноманітних математичних методах і алгоритмах, які дозволяють системі ефективно розпізнавати, аналізувати та захищати дані та середовище безпеки.Інформаційне забезпечення інтелектуальної пропускної системи без пунктів - це критичний аспект, оскільки воно забезпечує обробку, зберігання та передачу даних у системі безпеки.

Інформаційне забезпечення інтелектуальної пропускної системи вимагає комплексного підходу до забезпечення безпеки, доступності та правильної обробки даних, що включає розробку спеціалізованих програмних засобів та врахування вимог щодо захисту особистих даних та регулятивних стандартів.

РОЗДІЛ 3

РОЗРОБКА ІНТЕЛЕКТУАЛЬНОЇ ПРОПУСКНОЇ СИСТЕМИ

3.1. Програмна реалізація інтелектуальної пропускної системи.

Як було зазначено в попередніх розділах та підрозділах моєї дипломної роботи, інтелектуальні пропускні системи, що базуються на механізмах розпізнавання образів, використовують передові технології штучного інтелекту, однак все залежить від типів, які дана використовує. Зокрема машинного навчання та комп'ютерного зору, для ідентифікації та автентифікації осіб.

Основні складові програмної реалізації інтелектуальної пропускної системи на основі розпізнавання образів включають такі етапи:

1. *Збір даних*: починаючи зі збору даних про обличчя або інші біометричні параметри (відбитки пальців, риси очей тощо), система використовує відеокамери, сканери або інші пристрої для отримання образів.

2. *Підготовка та оброблення даних*: отримані зображення обробляються для покращення якості та відокремлення параметрів, які будуть використовуватися для подальшого аналізу.

3. *Виділення ознак*: цей етап включає в себе виділення унікальних характеристик з обличчя або іншого біометричного зразка, таких як контур, риси, точки, що розрізняють одну особу від іншої. Ці ознаки можуть бути різними для кожної системи розпізнавання.

4. *Створення моделі*: тут звичайно використовується точна техніка машинного навчання, такі як нейронні мережі або класифікатори, щоб навчити систему розпізнавати та класифікувати зображення на основі отриманих ознак.

5. *Розпізнавання та автентифікація*: зображення, яке надходить до

системи, порівнюється з вже збереженими даними у базі. Наявність співпадінь або відмінностей визначає, чи доступ дозволено.

6. *Підтвердження доступу*: якщо особа ідентифікована успішно, система видає дозвіл на доступ або відмовляє у доступі відповідно до налаштувань.

Технології, що використовуються для реалізації цих систем, включають:

1. *Машинне навчання*: для навчання моделей розпізнавання на основі великої кількості даних.

2. *Глибинне навчання*: використовує нейронні мережі для автоматичного визначення ознак та класифікації.

3. *Комп'ютерний зір*: використання алгоритмів для обробки зображень та виділення ознак.

Ці системи можуть бути використані у пропускних системах на входах офісів, громадських будівель, на території підприємств для контролю доступу, а також в системах безпеки для впізнавання та відслідковування осіб. Такі системи надають високий рівень безпеки та автоматизують процес контролю доступу.

Для більш детального пояснення щодо програмної реалізації інтелектуальної пропускної системи, я продемонструю вам наступні зображення, на яких продемонстровано основні блоки коду, який я писав особисто для програмного забезпечення своєї власної програми для реалізації інтелектуальної пропускної системи з детальним поясненням у вигляді коментарів всередині створення програмного забезпечення (див. рис. 3.1 – 3.3):

```python
import threading

import cv2
from deepface import DeepFace # імпорт основних бібліотек та фреймворків для коректного створення програми

cap = cv2.VideoCapture(0, cv2.CAP_DSHOW) # ініціалізація змінної, яка буде фіксувати особу на камеру

cap.set(cv2.CAP_PROP_FRAME_WIDTH, value: 640) # параметри камери фіксації за шириною
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, value: 480) # параметри камери фіксації за висотою

counter = 0 # створення змінної лічильника, який буде фіксувати людей (за замовчуванням - 0)

face_match = False # створення змінної, яка буде виконувати функцію фіксування особи в момент її появи на камеру

reference_img = cv2.imread("photo.jpg") # створення змінної, яка посилається на фото в якій буде відбуватися фіксація

# створення функції, яка перевіряє особу на верифікацію
1 usage
def check_face(frame):
    global face_match
    try:
        if DeepFace.verify(frame, reference_img.copy())['verified']:
            face_match = True
        else:
            face_match = False

    except ValueError:
        face_match = False
```

Рис 3.1

```python
# цикл та логіка програми, які перевіряють параметри обличчя особи в момент її потрапляння в поле зору сканування камери
while True:
    ret, frame = cap.read()

    if ret:
        if counter % 30 == 0:
            try:
                threading.Thread(target=check_face, args=(frame.copy(),)).start()
            except ValueError:
                pass
        counter += 1

# логічний висновок програми коли особу вдалося ідентифікувати
        if face_match:
            cv2.putText(frame, text: "The person is detected.", org: (20, 450), cv2.FONT_HERSHEY_SIMPLEX, fontScale: 2, color: (
        else:
            cv2.putText(frame, text: "The person is not detected.", org: (20, 450), cv2.FONT_HERSHEY_SIMPLEX, fontScale: 2, co

        cv2.imshow( winname: "video", frame)

# вихід з програми та завершення її функціоналу
    key = cv2.waitKey(1)
    if key == ord("q"):
        break

cv2.destroyAllWindows()
```

Рис 3.2.

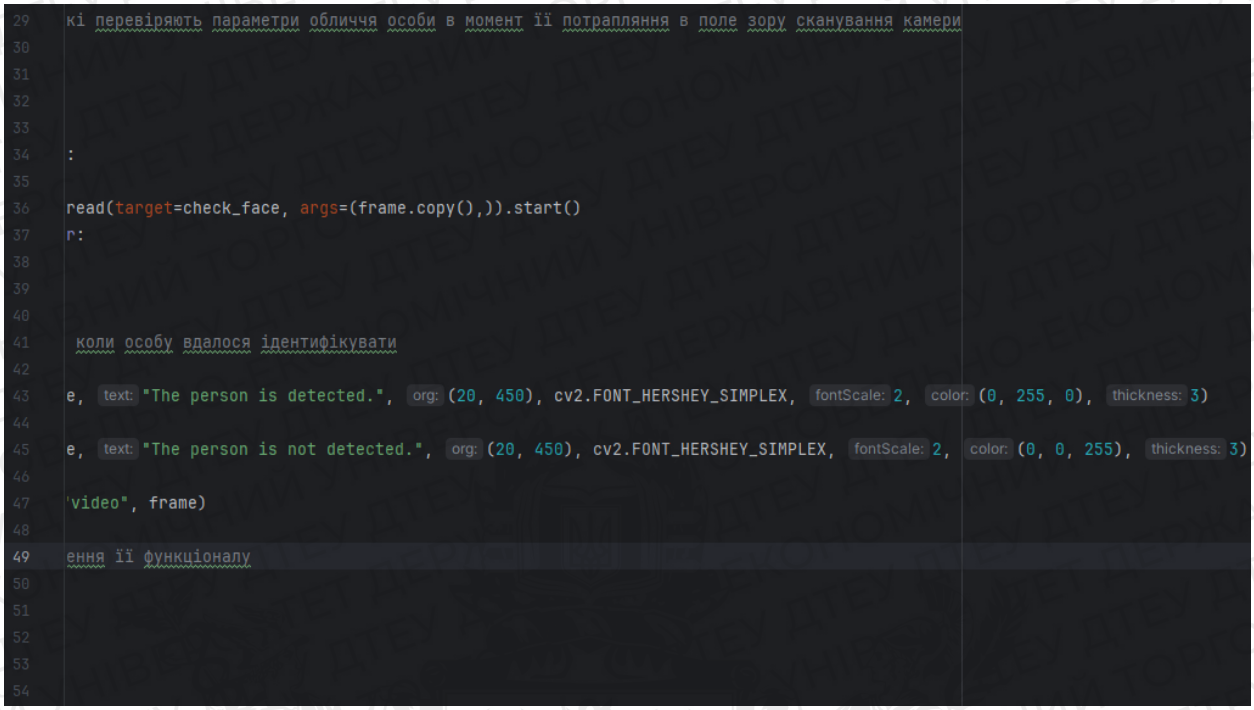

Рис 3.3.

На зображеннях вище ви можете спостерігати основні блоки коду програмного забезпечення інтелектуальної пропускної системи.

3.2. Розробка інтерфейсу пропускної системи

Інтерфейс пропускної системи повинен бути дружелюбним та ефективним. Це включає в себе створення програмного забезпечення, яке керує процесом ідентифікації облич. Інтерфейс повинен мати можливість реєстрації нових користувачів, а також швидку та точну перевірку ідентичності наявних користувачів.

Розробка інтерфейсу системи ідентифікації облич включає в себе кілька ключових етапів, спрямованих на забезпечення зручності користування, швидкості і точності ідентифікації. Ось кілька основних

аспектів розробки інтерфейсу такої системи:

1. *Дизайн користувацького інтерфейсу (UI)*: успішний інтерфейс повинен бути інтуїтивно зрозумілим для користувачів. Це означає простоту та легкість взаємодії з системою. UI має включати в себе елементи, які дозволяють користувачам з легкістю реєструвати своє обличчя та отримувати підтвердження ідентифікації.

2. *Адаптивний дизайн*: інтерфейс повинен бути адаптивним до різних пристроїв та розмірів екранів, щоб забезпечити консистентність та зручність взаємодії для користувачів, які використовують систему на різних пристроях.

3. *Безпека і конфіденційність*: важливо враховувати аспекти безпеки та конфіденційності при розробці інтерфейсу. Збереження та захист даних облич користувачів є критично важливим аспектом, тому інтерфейс повинен мати високі стандарти забезпечення безпеки даних.

Розробка інтерфейсу системи ідентифікації облич - це складний процес, який вимагає сполучення знань з дизайну, технологій розпізнавання облич та врахування потреб користувачів. Але варто звернути увагу, що дизайни є різними і можна створити свій власний але при цьому ефективний та зручний інтерфейс для пропускних систем на базі ідентифікації облич.

Для наступного прикладу, я хочу звернути вашу увагу на розробку власного інтерфейсу, який я створив власноруч створюючи програмне забезпечення для пропускної системи (див. рис. 3.4 – 3.5):

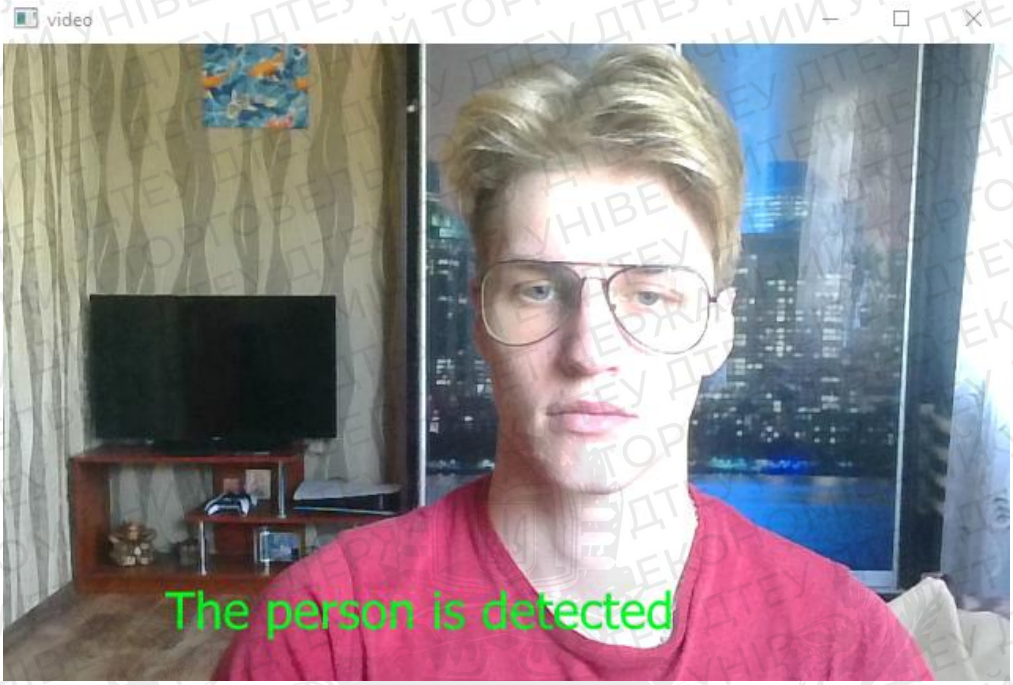

Рис. 3.4. *Програма повертає значення надпису зеленого кольору якщо особу вдалося ідентифікувати (обличчя дивиться прямо на камеру для сканування).*

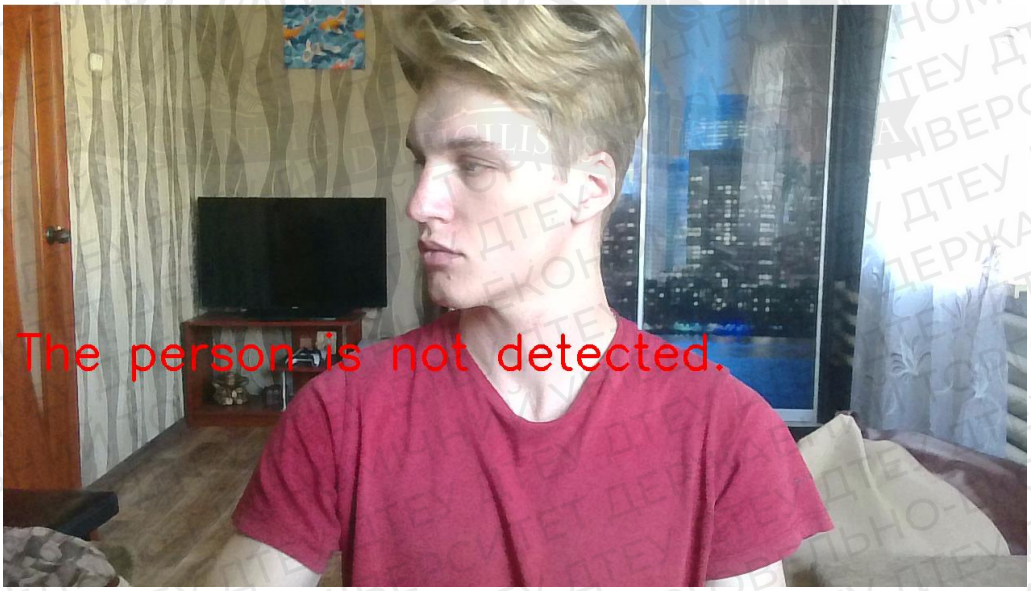

Рис. 3.5. *Програма повертає значення надпису червоного кольору та попереджає що особу не ідентифіковано (обличчя не дивиться прямо на камеру для сканування).*

Розробка інтерфейсу для пропускної системи на базі ідентифікації облич - це складний та багатоплановий процес. Інтерфейс повинен бути максимально зручним для користувачів. Це означає простоту взаємодії та інтуїтивний дизайн для забезпечення швидкості та зручності в процесі реєстрації та ідентифікації. Збереження та захист даних користувачів облич є надзвичайно важливим. Інтерфейс повинен мати високі стандарти безпеки для запобігання можливих загроз та забезпечення конфіденційності особистих даних. Також розробка інтерфейсу - це постійний процес. Після випуску системи важливо продовжувати вдосконалювати та оновлювати інтерфейс, враховуючи зміни технологій та потреб користувачів.

Загальний висновок полягає в тому, що успішний інтерфейс для пропускної системи на базі ідентифікації облич потребує глибокого розуміння потреб користувачів, високих стандартів безпеки та постійного вдосконалення на основі зворотного зв'язку та технологічних інновацій.

3.3. Тестування роботи пропускної системи

Тестування роботи пропускної інтелектуальної системи, заснованої на розпізнаванні образів, виконується для оцінки її точності, швидкості та загальної ефективності. Цей процес допомагає визначити, наскільки система може відтворити реальні умови та розпізнавати різні типи образів. Додатково, тестування роботи пропускної інтелектуальної системи на основі розпізнавання образів дозволяє оцінити її продуктивність та надійність в реальних умовах та визначити її придатність для конкретних завдань, пов'язаних з розпізнаванням образів.

Основні етапи тестування такої системи включають у себе:

1. *Налаштування системи*: перш за все, систему потрібно підготувати до тестування, включаючи завантаження даних, налаштування параметрів моделі та підготовку алгоритмів розпізнавання.

2. *Визначення метрик*: обираються метрики, за якими буде

вимірюватися продуктивність системи, такі як точність розпізнавання, швидкість відповіді, чутливість та специфічність розпізнавання.

3. *Тестові дані*: створення набору тестових даних, які представляють різноманітні образи, які система повинна розпізнати. Ці дані використовуються для перевірки роботи системи.

4. *Виконання тестів*: тестування проводиться шляхом подання образів на вхід системі та аналізу реакції. Це включає аналіз результатів, порівняння їх з правильними відповідями та вимірювання метрик ефективності.

5. *Оцінка результатів*: результати тестування оцінюються з урахуванням метрик та можливих помилок чи обмежень системи. Це допомагає визначити сильні та слабкі сторони системи.

6. *Підтвердження та вдосконалення*: на основі результатів тестування приймаються рішення щодо підтвердження або вдосконалення роботи системи. Якщо виявляються недоліки, вони виправляються, а система може бути перетестована для підтвердження покращень.

Тестування інтелектуальних систем, зокрема тих, що базуються на розпізнаванні образів, є складним та важливим процесом, який допомагає забезпечити їхню надійність, точність та придатність до реального використання.

Для наступної демонстрації я хочу звернути вашу увагу на свій власний тест для свого програмного забезпечення, який я створив щоб перевірити програму на валідність відповідної ефективності. Для цього я створив додатковий файл під назвою test.py для мови програмування Python і створив код, який допомагає протестувати програмне забезпечення мого власного коду для інтелектуальної пропускної системи на базі розпізнавання образів (див. рис. 3.6 – 3.7):

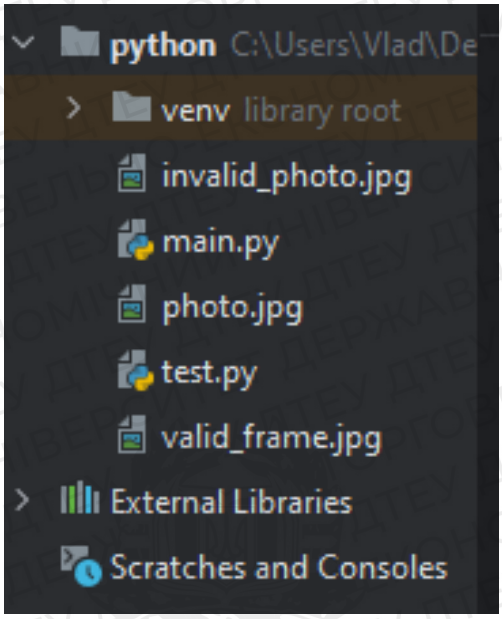

Рис. 3.6. *Створення файлу для тестування test.py мови програмування Python в середовищі, де буде відбуватися тестування програмного забезпечення з файлу main.py.*

main.py test.py

```python
import unittest
import cv2
from main import check_face

class TestFaceDetection(unittest.TestCase):

    def test_face_check_with_valid_image(self):
        # передаємо валідне зображення та перевіряємо, чи функція поверне очікувані результати
        reference_img = cv2.imread("photo.jpg")  # передаємо реальне зображення
        frame = cv2.imread("video")  # передаємо тип зображення (відео)
        result = check_face(frame, reference_img)
        self.assertTrue(result)  # очікується, що результат буде True

    def test_face_check_with_invalid_image(self):
        # передаємо невалідне зображення та перевіряємо, чи функція поверне очікувані результати
        reference_img = cv2.imread("invalid_photo.jpg")  #передаємо "invalid_photo.jpg" як невалідне зображення
        frame = cv2.imread("valid_frame.jpg")  # передаємо "valid_frame.jpg" як реальне зображення
        result = check_face(frame, reference_img)
        self.assertFalse(result)  # очікується, що результат буде False

if __name__ == '__main__':
    unittest.main()
```

Рис. 3.7. *Створення коду для тестування в файлі test.py використовуючи модулі, класи та логічні оператори мови програмування Python.*

На мою думку, можливо було б використати бібліотеку **unittest** для написання тестів для своїх функцій з файлу main.py. Проте, оскільки мій код в основному працює з **OpenCV** та бібліотекою **DeepFace**, цілком складно провести автоматизоване тестування, оскільки воно потребує наявності реальних зображень для перевірки функціоналу.

Як альтернатива, я спробував провести тестування функцій і методів, які не вимагають наявності реальних зображень, які можна було б автоматизовано перевірити. На зображенні 11.2 я навів приклад тестів за допомогою бібліотеки unittest для функцій check_face() та перевірки логіки програми.

Тут я спробував підготувати два тести для функції check_face(), один з валідним, а інший з невалідним зображенням.

Як висновок, тестування пропускної системи, яка ґрунтується на розпізнаванні образів, є ключовим етапом для забезпечення її ефективності та надійності в реальних умовах використання. Цей процес дозволяє оцінити рівень точності, швидкості та загальної продуктивності системи.

Перш за все, налаштування системи включає завантаження даних, параметрів моделі та алгоритмів розпізнавання. Визначення метрик, таких як точність, швидкість реакції та інші, стає важливим для оцінки продуктивності системи.

Формування репрезентативного набору тестових даних є ключовим кроком для перевірки системи в різних умовах та сценаріях. Тестування включає подачу образів на вхід системи та аналіз реакції, порівняння результатів з правильними відповідями та вимірювання метрик ефективності.

Оцінка результатів тестування допомагає виявити сильні та слабкі сторони системи. На основі цього виробляються рішення щодо підтвердження або вдосконалення роботи системи. Виявлені недоліки виправляються, що може призвести до подальшого тестування для перевірки вдосконалень.

У цілому, тестування пропускних систем на основі розпізнавання

образів допомагає забезпечити їхню точність та ефективність в реальних умовах, визначає їхню придатність для виконання конкретних завдань та сприяє подальшому вдосконаленню та розвитку цих систем.

Висновки до розділу

Розробка програмної реалізації інтелектуальної пропускної системи є складним та многогранним процесом, який вимагає дотримання кількох етапів для створення функціональної та ефективної системи. Перш за все, програмна реалізація включає розробку та впровадження алгоритмів машинного навчання або глибокого навчання, що дозволяють системі розпізнавати образи. Цей процес вимагає кодування та програмування для оптимізації швидкості та точності розпізнавання образів.

Розробка програмної частини системи також включає інтеграцію алгоритмів та моделей у внутрішню структуру системи. Це вимагає не лише створення алгоритмів розпізнавання, а й їхню оптимізацію для ефективної роботи системи в реальному часі.

Крім того, програмна реалізація вимагає впровадження різноманітних технологій та створення інтерфейсів для взаємодії з користувачем. Це може включати створення веб-сайту, мобільного додатка чи інших інтерфейсів, що спрощують процес реєстрації та введення даних у систему.

Програмна реалізація інтелектуальної пропускної системи вимагає постійного вдосконалення, випробувань та оптимізації, щоб забезпечити її продуктивність, точність та надійність у реальних умовах використання. Такий підхід дозволяє створити працездатну систему, яка може успішно впроваджуватися та використовуватися у різних сферах, де необхідне розпізнавання образів та контроль доступу.

Також, розробка інтерфейсу пропускної системи є не менш важливою складовою процесу створення функціональної та зручної для користувача системи контролю доступу. Цей етап включає не лише дизайн, а й реалізацію

інтерфейсу, який відповідає потребам користувачів та сприяє зручній та ефективній взаємодії з системою.

Важливим аспектом розробки інтерфейсу є врахування потреб користувачів, їхніх вподобань та звичок. Ергономіка та зручність використання інтерфейсу грають ключову роль у покращенні користувацького досвіду.

Крім того, розробка інтерфейсу також включає в себе тестування та вдосконалення, з метою забезпечення оптимального функціонування системи та виправлення можливих проблем.

У цілому, ефективний інтерфейс пропускної системи відображає сумісність з потребами користувачів, сприяє зручності та простоті взаємодії, що є важливими факторами для успішного впровадження та використання системи контролю доступу.

І останнє, але не менш важливе значення має розробка тестування роботи пропускної системи, яка є критично важливою для забезпечення її ефективності, надійності та придатності до використання в реальних умовах. Цей процес дозволяє оцінити рівень точності, швидкості та загальної продуктивності системи, а також виявити її сильні та слабкі сторони.

Тестування пропускної системи включає в себе використання різних методів та наборів тестових даних для аналізу реакції системи на різноманітні умови. Важливо оцінювати якість розпізнавання образів, точність реакції системи на вхідні дані та швидкість обробки цих даних.

Під час тестування необхідно акцентувати увагу на перевірці відповідності системи вимогам, які були поставлені перед її розробкою. Також важливо враховувати можливість виявлення помилок та недоліків системи, що дозволить їх виправити та вдосконалити роботу системи.

Завдяки тестуванню можна отримати об'єктивну інформацію про продуктивність системи та її придатність для конкретних завдань, пов'язаних з контролем доступу та розпізнаванням образів. Цей процес є невід'ємною частиною розробки пропускної системи, яка сприяє її оптимізації та

вдосконаленню, підвищуючи загальний рівень її функціональності та користувацького досвіду.

ВИСНОВКИ

Як один із підсумкових висновків моєї дипломної роботи, маю зауважити, що інтелектуальна пропускна система на основі механізмів розпізнавання образів є потужним інструментом для забезпечення безпеки та контролю доступу. Для її успішної реалізації необхідно використовувати багато ключових компонентів.

До даних типів систем використовуються технології, які дозволяють ідентифікувати особу за зовнішніми ознаками обличчя. Вони можуть використовувати методи глибокого навчання для точного розпізнавання осіб. Особливу увагу треба звернути також на біометричні сканери, які виконують функції датчиків для збору біометричних даних, таких як відбитки пальців, сканування сетчатки ока або інші біометричні параметри для ідентифікації особи.

Важливу роль відіграють відеоспостереження та камери з високою роздільною здатністю, адже ці компоненти служать для візуального моніторингу та фіксування образів для подальшого аналізу.

На мою думку, найголовнішу роль відіграють методи машинного навчання та штучного інтелекту. Використання алгоритмів машинного навчання для навчання системи розпізнавання образів та аналізу зібраних даних для постійного вдосконалення.

Загалом, інтелектуальна пропускна система на основі механізмів розпізнавання образів вимагає точних технологій, захисту даних, швидкості та ефективності для успішної реалізації контролю доступу та забезпечення безпеки.

У майбутньому інтелектуальні пропускні системи на основі механізмів розпізнавання образів відіграють ключову роль у сферах безпеки, технологій та соціального життя.

Вони будуть широко використовуватися в промисловості, бізнесі та громадських просторах для забезпечення безпеки та контролю доступу до

будівель, об'єктів і територій. У транспорті вони можуть використовуватися для автоматичної ідентифікації пасажирів в автомобілях, на вокзалах, в аеропортах та в громадському транспорті. Вони можуть бути використані для безпечного доступу до медичних установ, лабораторій, зберігання медичних даних та лікарських препаратів. В інтелектуальних будівлях вони забезпечать безпеку, включаючи контроль доступу, моніторинг та автоматизацію різних систем.

В цілому, інтелектуальні пропускні системи на основі розпізнавання образів стануть невід'ємною частиною нашого повсякденного життя, допомагаючи забезпечити безпеку, зручність та ефективність у різних сферах діяльності.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Книга "Computer Vision: Algorithms and Applications" (Р. Сзеліскі та ін.).

2. Книга "Handbook of Face Recognition" (Л. С. Лі та А. К. Джейн).

3. Файн В. С. Розпізнавання зображень, М. 1970.

4. Duda, R.O., Hart, P.E., & Stork, D.G. (2012). Pattern Classification (2nd ed.). Wiley.

5. Bishop, C.M. (2006). Pattern Recognition and Machine Learning. Springer.

6. Theodoridis, S., & Koutroumbas, K. (2009). Pattern Recognition (4th ed.). Academic Press.

7. Deep Learning Book by Ian Goodfellow, Yoshua Bengio, and Aaron Courville (http://www.deeplearningbook.org).

8. "A Few Useful Things to Know About Machine Learning" by Pedro Domingos.

9. Ивахненко А. Г., Лапа В. Г. Кибернетические предсказывающие устройства. — К.: «Наукова думка», 1965. — 216 с. — ISBN 978-5-458-61159-6.

10. "The Wolfram Language Image Identification Project". www.imageidentify.com. Retrieved 2017-03-22.

11. Bengio, Y. Artificial Neural Networks and their Application to Speech/Sequence Recognition // McGill University Ph.D. thesis.. — 1991.

12. Morgan, Nelson; Bourlard, Hervé; Renals, Steve; Cohen, Michael; Franco, Horacio. Hybrid neural network/hidden markov model systems for continuous speech recognition // International Journal of Pattern Recognition and Artificial Intelligence. — 1993-08-01.

13. . Weng, N. Ahuja and T. S. Huang,. Learning recognition and segmentation of 3-D objects from 2-D images // Proc. 4th International Conf. Computer Vision, Berlin, Germany, pp. 121-128. — May, 1993.