

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ РАДІОФІЗИКИ ЕЛЕКТРОНІКИ ТА КОМП'ЮТЕРНИХ СИСТЕМ

Кафедра радіотехніки та радіоелектронних систем

До захисту допущено:

«На правах рукопису»

Завідувач кафедри _____ Ігор АНІСІМОВ

18 травня 2023 р.

КВАЛІФІКАЦІЙНА МАГІСТЕРСЬКА РОБОТА

на тему:

«Система моніторингу території з використанням комп'ютерного зору»

Виконав:

студент 2-го курсу магістратури

денної форми навчання

спеціальності 172 Телекомунікації та радіотехніка

ОНП «Інформаційна безпека телекомунікаційних систем і мереж»

Азарний Дмитро Ігорович _____

Науковий керівник:

к.т.н., доц. Жиров Геннадій Борисович _____

Рецензент:

д. т. н., проф. Вишнівський Віктор Вікторович _____

Засвідчую, що у цій магістерській роботі

немає запозичень з праць інших авторів без

відповідних посилань

Студент _____

Робота допущена до захисту в ЕК рішенням кафедри радіотехніки та радіоелектронних систем від «18» травня 2023 р., протокол № 18.

Завідувач кафедри радіотехніки та радіоелектронних систем,

доктор фіз.-мат. наук, професор

Анісімов Ігор Олексійович _____

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1. АНАЛІЗ СУЧАСНИХ ТЕХНОЛОГІЙ ВИЯВЛЕННЯ ПОРУШНИКІВ.....	6
1.1.Аналіз існуючих методів детектування.....	6
1.2. Аналіз алгоритму існуючих систем детектування.....	8
1.2.1 Системи на основі відеоспостереження.....	8
1.2.2 Системи на основі інфрачервоних датчиків.....	8
1.2.3 Системи на основі мікрохвильових датчиків.....	9
1.2.4 Системи на основі ультразвукових датчиків.....	10
1.3 Приклади доступних систем безпеки на ринку.....	11
1.3.1 MotionCam Outdoor.....	11
1.3.2 StarterKit Cam.....	12
1.4.Висновки.....	13
РОЗДІЛ 2. АНАЛІЗ АЛГОРИТМІВ РОЗПІЗНАВАННЯ.....	14
2.1 Метод гнучкого порівняння графів.....	14
2.2 Нейронні мережі.....	15
2.3 Моделі активного зовнішнього вигляду.....	16
2.4 Метод каскадів Хаара (метод Віюли-Джонса).....	18
2.5 Висновок.....	20
РОЗДІЛ 3. ВИБІР ПРОГРАМНОЇ ТА ТЕХНІЧНОЇ БАЗИ	22
3.1 Загальні вимоги для побудови системи моніторингу.....	22
3.2 Вибір бібліотеки комп'ютерного зору	23
3.2.1 MATLAB Computer Vision Toolbox.....	23
3.2.2 SimpleCV.....	24
3.2.3 Caffe.....	25
3.2.4 TensorFlow.....	25
3.2.5 PyTorch.....	26
3.2.6 OpenCV.....	27

3.3 Вибір основної платформи для побудови системи.....	28
3.3.1 Аналіз наявних мікрокомп'ютерів.....	29
3.4 Аналіз вибору системи сповіщення.	29
3.4.1 Facebook Messenger.....	30
3.4.2 Slack.....	31
3.4.3 Discord.....	32
3.4.4 Skype.....	33
3.4.5 Telegram.....	33
3.5 Висновки.....	34
РОЗДІЛ 4. БУДОВА СИСТЕМИ МОНІТОРИНГУ.....	36
ВИСНОВКИ	40
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	42

ВСТУП

Актуальність дослідження. В сучасному світі, де науковий та технічний прогрес стрімко розвивається, немає меж для застосування комп'ютерних технологій в різних сферах життєдіяльності. Однією з таких сфер є моніторинг територій з метою забезпечення безпеки і контролю за порушеннями правопорядку. У цій галузі комп'ютерний зір відіграє важливу роль, дозволяючи автоматизувати процеси виявлення об'єктів та аналізу даних.

Комп'ютерний зір, який є частиною широкого спектру технологій штучного інтелекту, дозволяє автоматизувати процеси збору, обробки та аналізу великої кількості даних.

Один з прикладів застосування комп'ютерного зору - це системи відеоспостереження зі вбудованими алгоритмами розпізнавання образів. За допомогою таких систем можна виявляти порушення правил дорожнього руху, наприклад, переходи на червоне світло або паркування на тротуарах, а також допомагати при розслідуванні злочинів. Також комп'ютерний зір може використовуватися для виявлення змін у природних екосистемах, наприклад, для моніторингу стану лісів, земель або водних ресурсів.

Мета і завдання дослідження. Метою дипломної роботи є підвищення ефективності систем моніторингу території середньої цінової категорії. Для досягнення поставленої мети необхідно розв'язати актуальне науково-технічне завдання щодо розробки системи моніторингу території з використанням комп'ютерного зору з можливістю її швидкої модернізації.

Основним завданням системи є виявлення людини на відео з камери, якщо ця людина є об'єктом моніторингу. При виявленні людини система має надсилати повідомлення зі знімком екрану у чаті Telegram та СМС повідомлення через GSM модем.

Для досягнення цієї мети були поставлені наступні задачі:

- дослідити та порівняти існуючі методи виявлення людини на відео з камери;
- розробити алгоритм виявлення людини на відео з камери з використанням комп'ютерного зору;
- розробити програмне забезпечення для роботи системи моніторингу території;
- провести експериментальну перевірку розробленої системи.

Об'єктом дослідження є алгоритми комп'ютерного зору.

Предметом дослідження є програмні засоби для включення технології комп'ютерного зору до системи моніторингу території.

Апробація наукових результатів. Основні положення магістерської роботи оприлюднені на XVIII Міжнародній науково-практичній конференції: Військова освіта та наука: сьогодення та майбутнє, яка відбулася 25 листопада 2022 р. у Військовому інституті Київського національного університету імені Тараса Шевченка.

Загальний обсяг роботи - __ сторінок друкованого тексту. Основний текст викладено на __ сторінках.

У розділі 1 розглянуто сучасні системи виявлення порушників.

У розділі 2 проаналізовано алгоритми розпізнавання осіб.

У розділі 3 проаналізовані існуючі програмні оболонки для роботи із технологією комп'ютерного зору та програми для створення ботів.

У розділі 4 наведено частини коду для побудови системи моніторингу та функціонал ботів для взаємодії з користувачем, описано логіку роботи готової системи.

У висновках підсумовано основні результати дослідження.

РОЗДІЛ 1

АНАЛІЗ СУЧАСНИХ ТЕХНОЛОГІЙ ВИЯВЛЕННЯ ПОРУШНИКІВ

1.1 Аналіз існуючих методів детектування

Перед розглядом конкретних методів детектування, важливо визначити загальні принципи та можливості систем безпеки. Сьогодні на ринку існує багато різноманітних технологій та систем, які можуть бути використані для виявлення порушень безпеки на території об'єкта. Для ефективного застосування таких систем, необхідно визначити їх основні переваги та недоліки. Таким чином, у даному розділі будуть проаналізовані існуючі методи детектування та їхні можливості.

Відеоспостереження: системи відеоспостереження використовують камери для запису зображень із певних ділянок території. Можливе використання технологій розпізнавання осіб та автомобільних номерних знаків.

Датчики руху: ці пристрої виявляють рух у заданій зоні та запускають тривогу. Датчики можуть бути активовані шляхом зміни температури, змін тиску, зміни магнітного поля і т.д.

Інфрачервоні датчики: ці пристрої використовують інфрачервоне випромінювання виявлення руху об'єктів у зоні огляду. Інфрачервоні датчики можуть використовуватися для виявлення людей, тварин та транспортних засобів.

Аудіо датчики: ці пристрої використовуються для виявлення звуків у заданій зоні. Вони можуть виявляти звуки, пов'язані з порушеннями безпеки, такі як звук скла, що ламається.

Системи контролю доступу: ці системи використовуються для обмеження доступу до певних ділянок території. Вони можуть включати системи ідентифікації, такі як сканери відбитків пальців, системи розпізнавання осіб і т.д.

Детектори диму та вогню: ці пристрої виявляють дим та/або вогонь та запускають тривогу. Вони можуть бути особливо важливими для виявлення пожеж у будинках та інших приміщеннях.[1]

Вуличне освітлення: вуличне освітлення може відігравати важливу роль у виявленні порушників. Хороше освітлення може забезпечити яскравість і видимість на території у темний час доби та зробити її менш привабливою для злочинців.

Геоелектричні датчики: ці пристрої використовуються для виявлення підземних тунелів, каналів та інших підземних об'єктів.

Радіочастотні ідентифікатори (RFID): ці пристрої використовуються для ідентифікації об'єктів або персоналу, які мають RFID мітки. Це може бути корисним для відстеження переміщень усередині будівлі або на території.

Ультразвукові датчики: ці пристрої використовують для виявлення руху об'єктів на території, особливо в умовах низької видимості.

Мікрохвильові датчики: ці пристрої використовуються для виявлення руху об'єктів, аналогічно до інфрачервоних датчиків, але можуть бути більш точними і працювати в умовах низької видимості.

Системи контролю за периметром: такі системи можуть включати різні типи датчиків і камер для виявлення порушень периметра.

Системи розпізнавання номерних знаків: такі системи використовуються для розпізнавання номерних знаків транспортних засобів, що може бути корисним для відстеження переміщень на території.

Безпілотні літальні апарати (Дрони): дрони можуть використовуватися для моніторингу території з повітря, особливо в важкодоступних місцях або на великих територіях.

1.2 Аналіз алгоритму роботи існуючих систем детектування

1.2.1 Системи на основі відеоспостереження

Системи охорони на основі відеоспостереження слідкують за територією, що охороняється, за допомогою камер. Камери можуть бути розміщені як усередині приміщень, так і на відкритих територіях, і можуть забезпечувати спостереження як у режимі реального часу, так і запис відео на зберігання для подальшого перегляду.

Для роботи такої системи камери повинні бути обладнані спеціальними функціями виявлення руху та аналізу зображення. При виявленні руху камера може зафіксувати зображення, а потім система аналізує зображення на наявність об'єктів або людей, що рухаються на території. Якщо система виявляє порушення, вона може спрацювати сигналізацію та/або сповіщати оператора охорони.

Сучасні системи відеоспостереження можуть бути обладнані додатковими функціями, такими як розпізнавання осіб, автоматична класифікація об'єктів, аналіз поведінки людей та багато іншого. Такі функції дозволяють значно підвищити ефективність системи та зменшити кількість хибних спрацьовувань.

Системи відеоспостереження можуть бути з'єднані з іншими системами охорони, наприклад, із системами виявлення руху або датчиками дверей та вікон. Це дозволяє створювати комплексні системи охорони, які можуть виявляти різні види порушень та швидко реагувати на них.

Однак слід зазначити, що системи відеоспостереження не є універсальним рішенням для всіх типів об'єктів, що охороняються, і вимагають комплексного підходу до реалізації охорони.[2].

1.2.2 Системи на основі інфрачервоних датчиків

Системи на основі інфрачервоних датчиків, також відомі як ІЧ-бар'єри, працюють на основі виявлення змін в інфрачервоному випромінюванні, які відбуваються під час руху об'єктів через певну ділянку простору.

Для роботи такої системи на території розміщуються спеціальні інфрачервоні датчики, які можуть бути як провідними, так і бездротовими. Кожен датчик у парі має властивість випромінювати інфрачервоні промені у бік іншого датчика. У момент коли хтось перетинає промені датчиків, система реєструє зміни в інфрачервоному випромінюванні і спрацьовує сигналізація.

Системи на основі інфрачервоних датчиків можуть бути встановлені на певному рівні над землею, що дозволяє працювати в різних умовах, таких як висока вологість, дощ, сніг або туман. Більш просунуті системи можуть бути обладнані фільтрами, які унеможливають помилкові спрацьовування від тварин, комах та інших джерел.

Такі системи можуть бути використані для охорони периметру будівель, складів, парків, аеропортів, кордонів та інших об'єктів. Вони ефективні в умовах, коли необхідно забезпечити швидкий та точний відгук на порушення периметра чи зони. Однак, варто зазначити, що ІЧ-бар'єри можуть бути обійдені за допомогою засобів маскування або блокування променів датчиків, тому вони мають бути використані у поєднанні з іншими методами охорони.[3].

1.2.3 Системи на основі мікрохвильових датчиків

Системи на основі мікрохвильових датчиків працюють за принципом випромінювання високочастотних радіохвиль та аналізу їхнього відображення від об'єктів, що знаходяться в зоні виявлення

Мікрохвильові датчики випромінюють короткі імпульси радіохвиль у зону виявлення. Якщо об'єкт знаходиться в зоні дії датчика, ці імпульси відбиваються від нього і повертаються до датчика, де вони аналізуються. Якщо виявляється рух об'єкта, система запускає сигналізацію.

Мікрохвильові датчики працюють незалежно від умов довкілля, таких як температура, вологість, дощ або сніг, і можуть виявляти рух на відстані до кількох десятків метрів.

Такі системи часто використовуються для охорони будівель, складів, аеропортів, територій з великою кількістю техніки та інших об'єктів, де необхідно виявляти рух на великих відстанях та в різних умовах довкілля.

Однак, варто зазначити, що мікрохвильові датчики можуть давати помилкові спрацьовування у разі відображення хвиль від об'єктів, що рухаються, таких як автомобілі, поїзди, вантажівки та інші, які знаходяться в зоні дії датчика. Тому при встановленні системи необхідно враховувати такі фактори та налаштовувати датчики, щоб унеможливити помилкові спрацьовування.[4].

1.2.4 Системи на основі ультразвукових датчиків

Системи на основі ультразвукових датчиків використовують звукові хвилі з частотою, що перевищує верхню межу чутності людини (близько 20 кГц), щоб виявляти об'єкти, що рухаються.

Ультразвукові датчики генерують короткі імпульси звукових хвиль у зоні виявлення. Якщо об'єкт знаходиться в зоні дії датчика, то хвилі відбиваються від нього і повертаються до датчика, де вони аналізуються. Якщо виявляється рух об'єкта, система запускає сигналізацію.

Такі системи можуть використовуватися як для внутрішньої, так і зовнішньої охорони. Вони часто використовуються для виявлення об'єктів, що рухаються, таких як люди або автомобілі, і можуть працювати за будь-яких погодних умов. Вони також можуть бути встановлені на огорожах, вікнах та дверях для виявлення спроб вторгнення.

Однак, слід зазначити, що ультразвукові датчики можуть мати обмежену ефективність у приміщеннях з великою кількістю перешкод або де є багато об'єктів, які можуть відображати звукові хвилі, наприклад, магазини або склади. Також, подібно до інших типів датчиків, ультразвукові датчики можуть давати помилкові спрацьовування, що може бути викликано, наприклад, присутністю тварин або сильним вітром. Тому перед встановленням таких систем необхідно провести ретельний аналіз місцевості та налаштувати датчики для мінімізації хибних спрацьовувань.[5].

1.3 Приклади доступних систем безпеки на ринку

1.3.1 MotionCam Outdoor

Ajax MotionCam Outdoor - це бездротовий детектор руху з камерою, призначений для захисту від несанкціонованого проникнення на відкритих територіях. Цей пристрій використовує технології для виявлення руху та запису відео фрагментів у високій якості.

Основні характеристики пристрою:

Бездротове підключення: MotionCam Outdoor працює на радіочастоті 868 МГц, що забезпечує стабільне бездротове підключення до 2 км дальністю передачі даних.

Виявлення руху: пристрій оснащений подвійною системою виявлення руху - по тепловізору та по руху, що забезпечує точне визначення факту знаходження людини або тварини на території, а також знижує кількість помилкових спрацьовувань.

Камера: MotionCam Outdoor оснащена високоякісною 3-мегапіксельною камерою з роздільною здатністю 2048x1536, яка здатна записувати відео у форматі H.264 із частотою 20 кадрів на секунду.

Інфрачервоне підсвічування: вбудовані інфрачервоні діоди дозволяють пристрою працювати в темний час доби, забезпечуючи чітку та якісну картинку навіть за відсутності освітлення.

Сирена: вбудована сирена потужністю 110 дБ спрацьовує у разі виявлення руху та здатна відлякувати потенційних зловмисників.

Стійкість до погодних умов: пристрій має захист від пилу та води, що відповідає стандарту IP54, що дозволяє використовувати його на відкритих територіях у будь-яких погодних умовах.

Тривалий час роботи: вбудований акумулятор ємністю 5200 мАг забезпечує до 5 місяців роботи без підзарядки.

Необхідність підключення централі Ajax Hub для обробки даних.

Ajax MotionCam Outdoor є універсальним та надійним пристроєм для забезпечення безпеки на відкритих територіях, таких як паркування, території заміських будинків, дачних ділянок та інших об'єктів.[6].

1.3.2 StarterKit Cam

StarterKit Cam від Ajax Systems - це комплект системи безпеки, що включає кілька пристроїв для забезпечення безпеки всередині приміщення. Комплект складається з центрального пристрою (Ajax Hub), дверного/віконного датчика (Ajax DoorProtect), датчика руху (Ajax MotionProtect) та пульта керування (Ajax SpaceControl).

Ajax Hub – це основний пристрій системи, який підключається до інтернету та керує всіма підключеними пристроями. Він оснащений вбудованим засобом зв'язку (Wi-Fi, Ethernet, 2G/3G/4G) і може працювати в автономному режимі до 16 годин у разі відключення живлення.

Ajax DoorProtect - це датчик, який встановлюється на двері або вікно і повідомляє про спробу злому. Він працює на батарейках, має захист від відкриття, знімання та неправильної установки, а також підтримує роботу в умовах низьких температур.

Ajax MotionProtect - датчик руху, який використовує інфрачервоний датчик для виявлення руху всередині приміщення. Він також працює на батарейках та може виявляти рух на відстані до 12 метрів.

Ajax SpaceControl - це пульт дистанційного керування системою безпеки, який дозволяє вмикати та вимикати систему, а також викликати сигнал тривоги.

StarterKit Cam включає всі пристрої StarterKit, а також камеру Ajax MotionCam. MotionCam - датчик руху, обладнаний камерою високої чіткості, яка знімає відео при спрацьовуванні датчика руху. Він також працює на батарейках та може виявляти рух на відстані до 12 метрів.

Всі пристрої комплекту підключаються до Ajax Hub за допомогою бездротової технології Jeweller, що забезпечує надійний зв'язок на відстані до 2 км та захищає від злому та перешкод. Крім того, система Ajax має власну

мобільну програму для віддаленого управління та моніторингу системи, що дозволяє оперативно реагувати на оповіщення про можливі проблеми в системі безпеки.[7].

1.4 Висновки

Аналізуючи сучасні системи моніторингу території та компонентну базу для їх створення було виявлено декілька суттєвих недоліків. Перш за все це обмеженість датчиків та зони охорони за яку вони відповідають. Більшість систем працюють за принципом інфрачервоного датчика та його варіацій, що робить систему вразливу до хибних спрацювань та температурних коливань. Навіть у розглянутих системах Ajax заявлена стійкість до найпоширенішого джерела хибних спрацювань – тварин, насправді обмежена розмірами тварин, адже починаючи з собак середнього розміру можуть виникати спрацювання системи. Інша проблема сучасних датчиків це їх висока ціна. Також проблема обмеженості датчиків впливає на проблему ціни, адже для повного захисту бажаного периметру треба використати велику кількість датчиків.

Головна проблема це відсутність гнучкості у застосуванні сучасних систем. Можливість зміни цілі детектування повністю відсутня у всіх розглянутих варіантах. При необхідності обмежувати систему чи розширити її функціонал може бути необхідно створювати нову систему з нуля, що тягне за собою попередньо розглянуті проблеми.

Отже, враховуючи зазначені вище недоліки було вирішено побудувати систему моніторингу території з використанням комп'ютерного зору, яка буде більш економічною, за наявні аналоги на ринку, та може бути налаштована в залежності від потреб користувача.

РОЗДІЛ 2

АНАЛІЗ АЛГОРИТМІВ РОЗПІЗНАВАННЯ

2.1 Метод гнучкого порівняння графів

Сутність методу полягає у еластичному порівнянні графів, що описують зображення осіб. Особи в даному методі являють собою графи зі зваженими вершинами і ребрами. На стадії розпізнавання один еталонний граф залишається незмінним, а решта видозмінюється заради найточнішого співставлення з еталонним. В системах розпізнавання графи можуть бути представлені як прямокутними решітками, так і структурами, утвореними антропометричними точками обличчя. На рисунку 2.1 зображено приклад еталонного графу, та видозміненого графу [8].

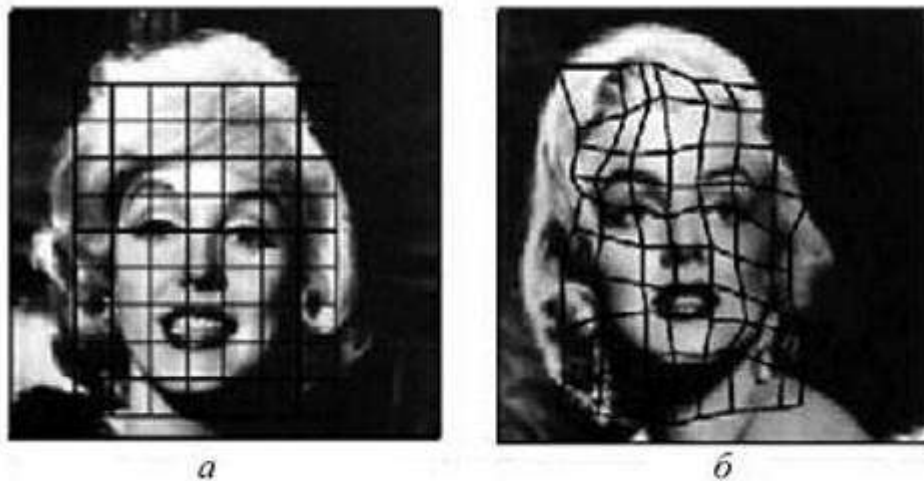


Рис.2.1 Еталонний та видозмінений граф

У вершинах графа обчислюються значення ознак, найчастіше для цього використовують комплексні значення габорівських фільтрів, що обчислюються в локальній області вершини графа шляхом згортки значень яскравості пікселів з фільтрами Габора.

Ребра графів порівнюються відстанями між сусідніми вершинами графу. Різниця між графами розраховується за допомогою функції деформації, що

враховує як різницю між значеннями ознак, обчисленими в вершинах, так і ступінь деформації ребер графа.

Деформація графів відбувається шляхом зміщення кожної з вершин на певну дистанцію в певному напрямку щодо вихідного положення та вибору такого розташування, при якому різниця в значеннях ознак в вершинах деформованих графів та відповідних вершинах еталонного графа буде якомога меншою. Така процедура відбувається для кожної вершини графа до того часу, поки не буде здобута мінімальна сумарна відмінність між ознаками видозміненого і еталонного графів. Така операція повинна бути виконана для всієї бази даних еталонних осіб. Результатом розпізнавання буде еталон з найточнішим значенням функції деформації [9].

Недоліками методу є дуже високий час розрахунків процедури розпізнавання, низька швидкість тренування нових еталонів.

Перевагами є досить висока точність розпізнавання при відхиленнях обличчя до 15 градусів від еталонного положення.

2.2 Нейронні мережі

На теперішній час існує кілька видів нейронних мереж. Нейронна мережа на багат шаровому перцептроні є найбільш використовуваною, вона дозволяє проводити класифікацію зображень на основі попереднього навчання.

Нейронні мережі навчаються на певному наборі прикладів. Суть навчання мережі зводиться до налагоджування ваг зв'язків між нейронами в процесі рішення оптимізаційної задачі за допомогою методу градієнтного спуску. З плином навчання нейронна мережа виділяє ключові ознаки, визначає їх важливість та будує взаємозв'язки. Навчена нейронна мережа може застосовувати набутий в процесі навчання досвід для детектування раніше невідомих образів за рахунок набутих здібностей.

При вирішенні задачі розпізнавання осіб, найкращі результати видає згортова нейронна мережа, що є наступним рівнем розвитку після

багатошарового перцептрона, за рахунок можливості врахування більшої кількості параметрів зображення. Згорткова нейронна мережа частково забезпечує стійкість до зміни масштабу, зсувів, поворотів, зміни ракурсу й іншим спотворенням. На рисунку 2.2 зображено архітектуру згорткової нейронної мережі [10].

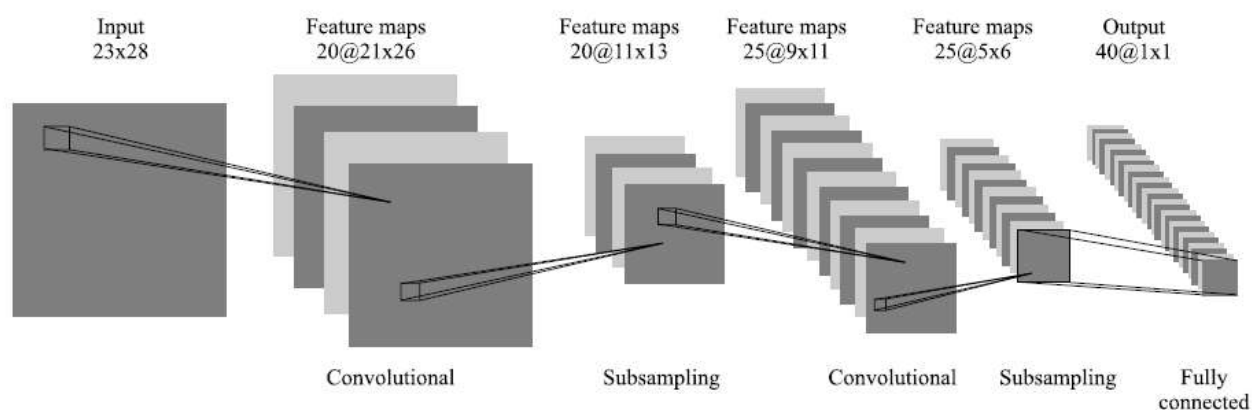


Рис.2.2 Архітектура згорткової нейронної мережі

Тестування згорткової нейронної мережі на базі даних, що містить фронтальні зображення осіб з малими змінами освітлення, поворотів, та емоцій показало точність до 96%. Свій розвиток нейронні мережі продовжили в розробці DeepFace, що придбав Facebook для процедури розпізнавання осіб користувачів своєї соцмережі. Всі особливості архітектури носять закритий характер.

До недоліків можна віднести те, що додавання в базу даних нової особи вимагає повного навчання мережі з нуля, що займає велику кількість часу. Складність в налаштуванні параметрів для навчання [11].

2.3 Моделі активного зовнішнього вигляду

Моделі активного зовнішнього вигляду - це математичні моделі із зображенням зовнішнього вигляду та форми сімейства предметів. Кожна модель має такі параметри, модифікація яких призводить до генерації різних

екземплярів даного об'єкта. У випадку з обличчям зміна параметрів моделі може означати відкриття або закриття рота, або поворот голови [12].

Використання моделі для визначення даного об'єкта на зображенні - це збіг параметрів моделі, щоб різниця між об'єктом на зображенні і початковою моделлю була мінімальною. Початкове положення, яке є точкою входу алгоритму зазвичай встановлюється за допомогою окремого детектора (наприклад детектор обличчя). На рисунку 2.3 зображено приклад виділення точок обличчя для даного методу.

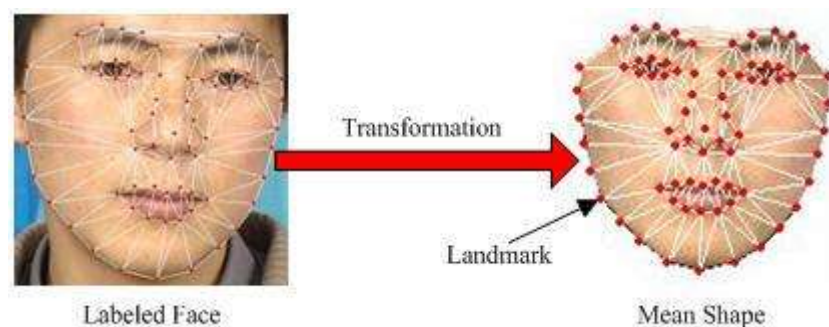


Рис.2.3 Виділення обличчя за допомогою антропометричних точок

Якщо предмет (наприклад, обличчя) має змінну форму та зовнішній вигляд, це необхідно промодельовати обидва ці явища. Для цього потрібно використовувати дві моделі. Одна модель це опис можливих форм предмета, а інша - опис можливого вигляду. Модель створюється на основі набору фотографій об'єкта, всі точки позначаються вручну. На рисунку 2.4 зображено антропометричні точки обличчя по яким будується модель [13].



Рис.2.4 Антропометричні точки обличчя

При вирішенні задачі локалізації об'єкту виконується пошук параметрів, що представляють собою зображення. При певному наближенні до еталонного зображення приймається рішення про наявність чи відсутність шуканого об'єкту.

2.4 Метод каскадів Хаара (метод Віоли-Джонса)

Для методу Віоли-Джонса головним інструментом є примітиви Хаара, що являють собою подрібнення заданої прямокутної області на набори прямокутних підобластей різного типу . На рисунку 2.5 зображено приклад примітивів Хаара.

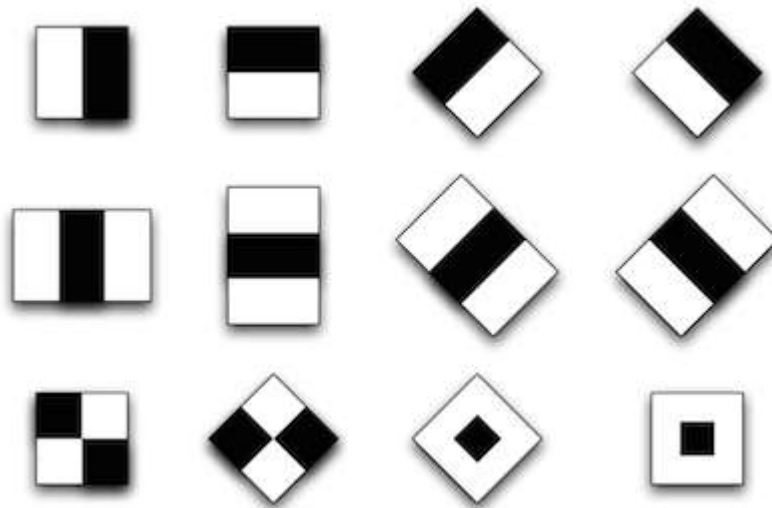


Рис.2.5 Примітиви Хаара

Ознака Хаара складається із сусідніх областей прямокутної форми. Вони розташовуються на зображенні, потім підсумовуються інтенсивності пікселів в областях, потім розраховується різниця між їх сумами. Ця різниця являє собою значення певної ознаки, певного розміру.[14]

В початковій версії методу Віоли-Джонса використовувалися лише примітиви без поворотів, а для розрахунку значення ознаки сума додавання яскравостей пікселів в першій підобласті віднімалася від суми додавання яскравостей в другій підобласті. Для покращення методу були впроваджені примітиви нахилені на 45 градусів та примітиви несиметричних конфігурацій, що дозволило збільшити швидкість розрахунку.

На етапі розпізнавання в методі Віоли-Джонса прямокутник заданого розміру пересувається по зображенню, та для всіх областей, над якими проходить вікно, розраховується ознака Хаара. Присутність чи відсутність шуканого предмета у вікні визначається різницею між розрахованим значенням ознаки і певним порогом. Примітиви Хаара слабо підходять для задачі навчання та класифікації, для опису об'єктів з необхідною точністю необхідна більша кількість ознак. Через це в методі Віоли - Джонса ознаки Хаара згруповані в каскадний класифікатор [15].

На рисунку 2.6 зображено каскад системи, що розпізнає обличчя.

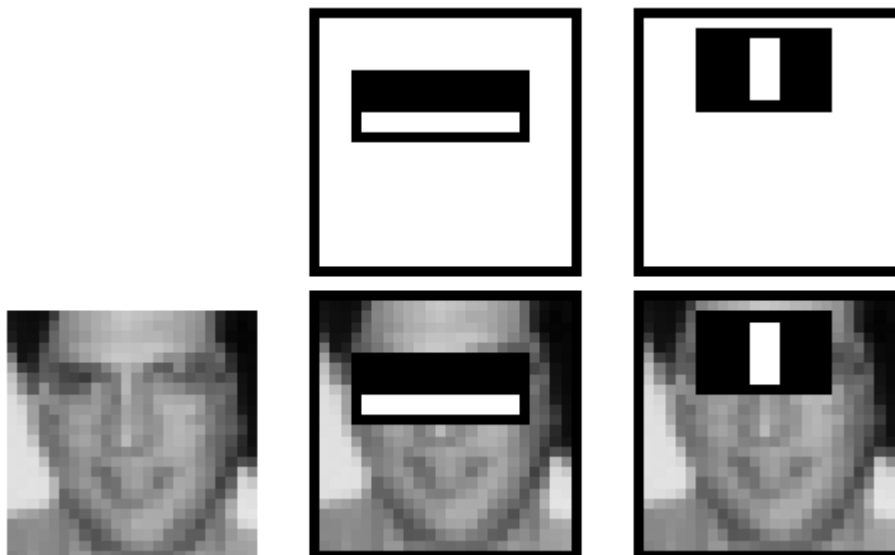


Рис.2.6 Каскад Хаара для розпізнавання обличчя

У методу Віюли-Джонса є свої переваги та недоліки.

До недоліків можна віднести зниження стійкості навчання при зміні освітлення, при зміні масштабу зображення, невелика швидкість розрахунку, слабка точність при великій кількості критеріїв пошуку.

До переваг можна віднести високу швидкість розрахунку окремої ознаки, можливість натренувати вузько спеціалізовану модель, простота реалізації.[16]

2.5 Висновок

Очевидно, що методи розпізнавання не обмежуються наведеними прикладами, їх значно більша кількість. Але для реалізації проекту потрібно обрати один. Обрано було метод Віюли-Джонса, бо його реалізація є відносно простою, існує багато матеріалів пов'язаних з ним, а інструменти для роботи можна знайти в відкритому доступі в OpenCV [17].

OpenCV є відкритою бібліотекою комп'ютерного зору, машинного навчання та обробки зображень. У сучасних системах вона відіграє важливу роль у вирішенні задач в режимі реального часу. За допомогою OpenCV можна обробляти зображення та відео для розпізнавання об'єктів, облич, та навіть

рукописного почерку. Інтеграція з іншими бібліотеками, такими як NumPy Python, дозволяє ефективно працювати з масивами даних OpenCV для подальшого аналізу. Для ідентифікації візерунка зображень та виконання різних функцій використовується векторний простір, а над цими ознаками виконуються математичні операції.[18]

РОЗДІЛ 3

ВИБІР ПРОГРАМНОЇ ТА ТЕХНІЧНОЇ БАЗИ

3.1 Загальні вимоги для побудови системи моніторингу

Для ефективної розробки системи моніторингу з використанням комп'ютерного зору необхідно враховувати інші фактори. Наприклад, важливо вибрати правильну камеру з оптимальними параметрами роздільної здатності, чутливості до світла та швидкості зйомки.

Також необхідно підібрати потрібне програмне забезпечення для аналізу зображень з камери та встановити необхідні алгоритми комп'ютерного зору. При цьому, варто розглянути як готові рішення, так і можливість розробки власних алгоритмів.

Окремої уваги заслуговує питання зберігання інформації, зібраної системою моніторингу. Важливо обрати надійну і безпечну систему зберігання даних, яка забезпечує їх захист від несанкціонованого доступу та зберігає дані відповідно до законодавства.

Крім того, необхідно передбачити можливість інтеграції з іншими системами безпеки та сповіщення у разі виявлення небезпечних ситуацій. Наприклад, систему моніторингу можна поєднати зі звуковою або світловою сигналізацією, що дозволить швидко реагувати на можливі загрози.

Також варто враховувати можливості розвитку проекту в майбутньому, оскільки система моніторингу може бути розширена з використанням нових технологій, наприклад, штучного інтелекту або дронів. Всі ці фактори необхідно розглянути та врахувати при розробці проекту системи моніторингу з використанням комп'ютерного зору.

3.2 Вибір бібліотеки комп'ютерного зору

3.2.1 MATLAB Computer Vision Toolbox

Computer Vision Toolbox надає алгоритми, функції та програми для проектування та тестування комп'ютерного бачення, 3D-бачення та систем обробки відео. Існує можливість виконувати виявлення та відстеження об'єктів, а також виявлення, вилучення та зіставлення ознак. Можливо автоматизувати робочі процеси калібрування для однокамерних, стереокамер і камер типу «риб'яче око». Для 3D-бачення набір інструментів підтримує візуальний режим і хмару точок SLAM, стереобачення, структуру з руху та обробку хмари точок. Програми для комп'ютерного бачення автоматизують робочі процеси маркування наземної правди та калібрування камери.

Можливо навчити власні детектори об'єктів за допомогою алгоритмів глибокого та машинного навчання, таких як YOLO, SSD і ACF. Для семантичної сегментації та сегментації екземплярів можна використовувати алгоритми глибокого навчання, такі як U-Net і Mask R-CNN. Набір інструментів містить алгоритми виявлення об'єктів і сегментації для аналізу зображень, які занадто великі, щоб поміститися в пам'ять. Попередньо підготовлені моделі дозволяють виявляти обличчя, пішоходів та інші звичайні об'єкти.

Можливо прискорити свої алгоритми, запустивши їх на багатоядерних процесорах і графічних процесорах. Алгоритми Toolbox підтримують генерацію коду C/C++ для інтеграції з існуючим кодом, прототипування робочого столу та розгортання системи вбудованого бачення.

Перевагами даного пакету є:

Набір інструментів та функцій для аналізу, проектування та впровадження алгоритмів комп'ютерного зору.

Надання комплексного набору алгоритмів та інструментів для обробки зображень, розпізнавання образів, комп'ютерного зору та машинного навчання.

Є інтегрованим з MATLAB, що дозволяє працювати зі зображеннями та аналізувати дані в зручному середовищі.

Можна легко використовувати готові функції для розв'язання стандартних завдань комп'ютерного зору.

Надає можливість створювати та налагоджувати власні алгоритми комп'ютерного зору.

Недоліки:

MATLAB та Computer Vision Toolbox є комерційними продуктами, що може бути дорогим для деяких користувачів.

Іноді складно відшукати необхідну функцію, оскільки у наборі є багато інструментів.

Для ефективного використання MATLAB та Computer Vision Toolbox необхідні відповідні знання та навички.[19]

3.2.2 SimpleCV

SimpleCV — це фреймворк з відкритим вихідним кодом, набір бібліотек і програмного забезпечення, які можна використовувати для розробки додатків комп'ютерного бачення. Він дозволяє працювати із зображеннями чи відеопотоками, які надходять із веб-камер, та IP-камер або мобільних телефонів. SimpleCV можна використовувати безкоштовно, і оскільки він є відкритим кодом, також можливо змінити код, якщо забажаєте. Він написаний на Python і працює на Mac, Windows і Ubuntu Linux. Його розробили інженери Sight Machine і ліцензовано за ліцензією BSD.

Переваги:

Легкий та інтуїтивно зрозумілий інтерфейс.

Безкоштовний та відкритий.

Легка інтеграція з Python та іншими бібліотеками.

Є великою кількістю документації та прикладів коду.

Можливість використовувати для прототипування та швидкої розробки.

Недоліки:

Менш потужний та обмежений у функціоналі порівняно з більшістю інструментів для роботи з комп'ютерним зором.

Недостатня підтримка від спільноти порівняно з іншими бібліотеками.

Менш ефективний порівняно з низькорівневими бібліотеками, такими як OpenCV.[20].

3.2.3 Caffe

Caffe - це відкрита бібліотека машинного навчання, яка використовується для розробки глибоких нейронних мереж. Вона написана на мові програмування C++, але має інтерфейси для взаємодії з Python та MATLAB.

Переваги:

Відкрита та безкоштовна.

Підтримує широкий набір алгоритмів машинного навчання та глибоких нейронних мереж.

Має швидку реалізацію завдяки використанню оптимізованої бібліотеки BLAS.

Легко інтегрується з Python та MATLAB.

Має велику спільноту користувачів та підтримку від розробників.

Недоліки:

Складний та незручний інтерфейс.

Вимагає високої кваліфікації в галузі машинного навчання та програмування.

Менш ефективний порівняно з низькорівневими бібліотеками, такими як TensorFlow та PyTorch.

Недостатня документація та приклади коду.[21].

3.2.4 TensorFlow

TensorFlow - це відкрита програмна бібліотека, розроблена Google, яка призначена для розв'язання завдань машинного навчання. Вона дозволяє будувати та тренувати нейронні мережі для автоматичного розпізнавання та класифікації зображень, досягаючи якості, що наближається до людського сприйняття. TensorFlow використовується як для досліджень, так і для розробки

власних продуктів Google. Основний API для роботи з бібліотекою реалізований для мови програмування Python, а також існують реалізації для R, C#, C++, Haskell, Java, Go та Swift.

Переваги:

Відкрита та безкоштовна.

Підтримує широкий набір алгоритмів машинного навчання та глибоких нейронних мереж.

Має високу продуктивність та ефективність завдяки використанню обчислювальних графів.

Легко інтегрується з Python та іншими бібліотеками, такими як Keras та Pandas.

Має велику спільноту користувачів та підтримку від Google.

Недоліки:

Складний та незручний інтерфейс для початківців.

Вимагає високої кваліфікації в галузі машинного навчання та програмування.

Менш гнучкий та менш зручний для прототипування порівняно з іншими бібліотеками.

Обмежена підтримка від спільноти порівняно з іншими бібліотеками.[22].

3.2.5 PyTorch

PyTorch - це відкрита бібліотека машинного навчання, розроблена компанією Facebook. Вона дозволяє розробляти та навчати глибокі нейронні мережі для розв'язання різноманітних задач машинного навчання.

Переваги:

Відкрита та безкоштовна.

Підтримує широкий набір алгоритмів машинного навчання та глибоких нейронних мереж.

Має простий та зрозумілий інтерфейс.

Легко інтегрується з Python та іншими бібліотеками, такими як NumPy та SciPy.

Має підтримку для роботи з графічними процесорами та обчислювальними кластерами.

Недоліки:

Не має такої високої продуктивності, як TensorFlow, в особливості для великих датасетів.

Не має такої великої спільноти користувачів, як TensorFlow.

Іноді може виникати проблема зі сумісністю між різними версіями PyTorch.

Є деякі проблеми з документацією та прикладами коду, зокрема, для початківців.[23].

3.2.6 OpenCV

OpenCV - це відкрита бібліотека алгоритмів комп'ютерного зору, обробки зображень та чисельних алгоритмів загального призначення. Вона має відкритий код і доступна на різних мовах програмування, включаючи C/C++, Python, Java, Ruby, Matlab, Lua та інші. OpenCV може бути використана як у комерційних, так і в академічних цілях згідно з умовами ліцензії BSD.

У жовтні 2009 року було випущено друге велике оновлення - OpenCV 2. Воно включало значні зміни в інтерфейсі C++, спрямовані на спрощення, поліпшення безпеки, додавання нових функцій та покращення продуктивності, зокрема для багатоядерних систем. У травні 2016 року компанія Intel придбала провідного розробника OpenCV - Itseez, що дало новий імпульс розвитку бібліотеки.

Переваги:

Відкрита та безкоштовна.

Підтримує широкий набір функцій та алгоритмів обробки зображень та комп'ютерного зору.

Має підтримку для роботи з різними мовами програмування, такими як Python, C++, Java, та інші.

Може працювати з різними форматами зображень та відео.

Має велику спільноту користувачів та активно розвивається.

Недоліки:

Іноді може бути складним для початківців у використанні.

Не має такої продуктивності та ефективності, як TensorFlow та PyTorch, в особливості для навчання глибоких нейронних мереж.

Іноді можуть виникати проблеми з оптимізацією та швидкістю виконання для певних функцій та алгоритмів.

Може бути складним для роботи з відеопотоками та стрімінговою обробкою зображень.

3.3 Вибір основної платформи для побудови системи

Для успішної реалізації проекту системи моніторингу необхідно обрати відповідну платформу, що забезпечуватиме ефективну роботу програмного коду. Вибір платформи обмежений вимогами до запуску програмного коду, тому необхідно ретельно підійти до питання вибору оптимального варіанту з необхідними характеристиками для ОС Windows, Linux або MacOS.

Один з можливих варіантів - використання мікрокомп'ютера. Ця платформа є оптимальним вибором для проектів, які потребують компактного розміру, низької вартості та низького споживання енергії. Мікрокомп'ютер може бути зібраний з необхідними компонентами, такими як процесор, пам'ять, зберігання даних та інтерфейси вводу/виводу.

Використання мікрокомп'ютера забезпечить необхідний рівень продуктивності для виконання завдань системи моніторингу, а також надасть можливість модернізувати платформу в майбутньому, що підвищить функціональність та збільшить обсяги зберігання даних. Окрім того, використання мікрокомп'ютера спрощує процес збірки та налаштування системи моніторингу, що значно знижує ризик помилок та збільшує ефективність процесу розгортання системи.

3.3.1 Аналіз наявних мікро комп'ютерів

Raspberry Pi: один з найпопулярніших міні-комп'ютерів з великою громадою користувачів та підтримкою спільноти. Raspberry Pi підтримує різні ОС та може використовуватись для різних завдань.

BeagleBone: міні-комп'ютер на базі процесора ARM з великим набором функцій та вбудованими сенсорами.

Odroid: міні-комп'ютер на базі процесора ARM з підтримкою ОС Android та Linux.

Intel NUC: міні-комп'ютер на базі процесора Intel, який працює на Windows та Linux.

ASUS Tinker Board: міні-комп'ютер на базі процесора ARM з підтримкою ОС Android та Linux.

NVIDIA Jetson Nano: міні-комп'ютер із високопродуктивним процесором NVIDIA та підтримкою фреймворків машинного навчання, таких як TensorFlow та PyTorch.

Google Coral Dev Board: міні-комп'ютер із підтримкою Google TensorFlow та високопродуктивним процесором Coral Edge TPU.

Всі з перелічених комп'ютерів мають приблизно однакові характеристики, відрізняючись лише у незначних для поставленої задачі деталях. Тому оптимальним вибором буде використання Raspberry Pi, як найпоширенішого з них, та з найбільшою спільнотою користувачів.[24]

3.4 Аналіз вибору системи сповіщення

Одним з основних критеріїв при виборі системи сповіщення є зручність у використанні, простота у налагодженні зв'язку, можливість автоматизації, та максимально зрозумілий інтерфейс, цим критеріям відповідають смс повідомлення та месенжери.

3.4.1 Facebook Messenger

Facebook Messenger Bot API – це інструмент розробки, що надається Facebook, який дозволяє створювати ботів для Facebook Messenger. Ці роботи можуть бути використані для автоматичного спілкування з користувачами через Facebook Messenger.

Нижче наведено деякі переваги та недоліки Facebook Messenger Bot API:

Переваги:

Величезна аудиторія: Facebook Messenger має понад 1,3 мільярда користувачів, що робить його одним із найпопулярніших месенджерів у світі. Це означає, що створення роботів для Facebook Messenger може бути дуже ефективним способом досягнення великої аудиторії.

Багаті можливості: Facebook Messenger Bot API надає багатий набір можливостей, які можуть бути використані для створення ботів, таких як відправлення повідомлень, робота із зображеннями, відео та аудіофайлами, керування картками та багато іншого.

Інтеграція з іншими програмами: Facebook Messenger Bot API інтегрований з іншими програмами Facebook, такими як Facebook Pages, що дозволяє створювати ботів, які можуть спілкуватися зі сторінками Facebook.

Недоліки:

Складність у розробці: Facebook Messenger Bot API може бути складним у використанні та розробці, особливо для новачків у розробці ботів. Це пов'язано з великою кількістю функцій та налаштувань, які потрібно враховувати під час створення бота.

Обмеження у використанні: Facebook Messenger Bot API має ряд обмежень, таких як обмеження на кількість повідомлень, що відправляються в день, що може призвести до блокування бота.

Низька відкритість: Facebook є закритою платформою, що означає, що доступ до деяких функцій може бути обмежений. Крім того, Facebook має

правила використання, які потрібно дотримуватись при створенні ботів, щоб уникнути блокування.

3.4.2 Slack

Slack Bot API – це інтерфейс програмування додатків (API) для створення ботів у Slack. Slack – це платформа комунікації для бізнесу, де користувачі можуть спілкуватися через текстові повідомлення, голосові та відеодзвінки.

Нижче наведено деякі переваги та недоліки Slack Bot API:

Переваги:

Інтеграція з іншими програмами: Slack Bot API інтегрований з безліччю інших програм, таких як Google Drive, Trello, GitHub та багато інших. Це означає, що ви можете створювати ботів, які можуть працювати з іншими сервісами та інтегрувати їх у ваш робочий процес.

Широкі можливості: Slack Bot API має безліч функцій, які можуть бути використані для створення ботів, таких як надсилання повідомлень, управління каналами, ролей та користувачів, робота з файлами та багато іншого.

Велика спільнота: Slack має велику спільноту користувачів та розробників, що робить її привабливою для створення ботів та спілкування з аудиторією.

Недоліки:

Ціна: Slack є платною платформою, тому використання Slack Bot API може бути дорожчим, ніж інші API для створення ботів, таких як Telegram Bot API.

Обмежений функціонал: Slack Bot API має обмеження у функціональності порівняно з іншими API, такими як Telegram Bot API або Facebook Messenger Platform. Наприклад, Slack Bot API не підтримує створення групових чатів.

Складність у розробці: Slack Bot API може бути більш складним у використанні та розробці, особливо для новачків у розробці ботів. Він також може мати більше обмежень і правил, які треба виконувати, щоб уникнути блокування бота.

3.4.3 Discord

Discord Bot API – це інтерфейс програмування додатків (API) для створення ботів у Discord. Discord – це платформа спілкування для геймерів та спільнот, де користувачі можуть спілкуватися через текстові повідомлення, голосові та відеодзвінки.

Нижче наведено деякі переваги та недоліки Discord Bot API:

Переваги:

Широкі можливості: Discord Bot API має безліч функцій, які можуть бути використані для створення ботів, таких як надсилання повідомлень, управління каналами, ролей та користувачів, робота з голосовими каналами та багато іншого.

Велика спільнота: Discord має велику спільноту користувачів та розробників, що робить її привабливою для створення ботів та спілкування з аудиторією.

Доступний: Discord Bot API відкритий та гнучкий, що означає, що розробники можуть створювати ботів з використанням будь-якої мови програмування та інтегрувати їх з іншими сервісами та API.

Недоліки:

Складність у розробці: Discord Bot API може бути більш складним у використанні та розробці, особливо для новачків у розробці ботів. Він також може мати більше обмежень і правил, які слід слідувати, щоб уникнути блокування бота.

Боти можуть бути заблоковані: Discord має сувору політику щодо спаму та неправомірних дій ботів. Якщо ваш бот порушує правила Discord, він може бути заблокований або вилучений без попередження.

Недоступність для деяких країн: Discord заблоковано в деяких країнах, що обмежує аудиторію для ваших роботів і може створювати труднощі у використанні Discord Bot API.

3.4.4 Skype

Skype Bot API – це інтерфейс програмування для створення ботів у Skype. Він дозволяє розробляти ботів, які можуть взаємодіяти з користувачами Skype через текстові повідомлення, голосові та відеодзвінки.

Нижче наведено деякі переваги та недоліки Skype Bot API:

Переваги:

Інтеграція з Microsoft: Skype Bot API надається Microsoft, що означає, що він інтегрований з іншими продуктами та сервісами Microsoft, такими як Office 365 та Azure. Це може спростити розробку та використання ботів для бізнес-цілей.

Розширені можливості: Skype Bot API має безліч функцій, які можуть бути використані для створення ботів, таких як надсилання повідомлень, керування контактами, відправлення файлів та багато іншого.

Широка аудиторія: Skype має широку аудиторію користувачів, що робить його привабливим для розробки ботів та спілкування з розробниками.

Недоліки:

Обмежена платформа: Skype Bot API обмежується використанням лише у Skype, що робить його менш універсальним, ніж інші API.

Обмежений функціонал: Skype Bot API має обмеження у функціональності порівняно з іншими API, такими як Telegram Bot API або Facebook Messenger Platform. Наприклад, Skype Bot API не підтримує створення групових чатів.

Складність у розробці: Skype Bot API може бути більш складним у використанні та розробці, особливо для новачків у розробці ботів. Він також може мати більше обмежень і правил, які слід слідувати, щоб уникнути блокування бота.

3.4.5 Telegram

Telegram Bot API - це інтерфейс програмування додатків (API) створення ботів в Telegram.

Нижче наведено деякі переваги та недоліки Telegram Bot API:

Переваги:

Простота створення та налаштування ботів: Telegram Bot API має інтуїтивно зрозумілий та простий інтерфейс для створення та налаштування ботів. Для створення бота вам не потрібно мати глибокого знання програмування або використовувати складні інструменти.

Широкі можливості: Telegram Bot API надає різноманітні функції, які можна використовувати для створення різних ботів, таких як надсилання повідомлень, керування групами, відправлення зображень та багато іншого. Крім того, API підтримує безліч мов програмування, таких як Python, Java, Node.js та інші.

Висока продуктивність: Telegram Bot API швидко та ефективно обробляє запити від користувачів, що дозволяє ботам відповідати на повідомлення та виконувати команди миттєво.

Недоліки:

Обмежений функціонал: Telegram Bot API має деякі обмеження щодо функціональності, такі як неможливість використання відео або звуку в ботах. Крім того, багато функцій доступні лише для привілейованих користувачів.

Боти можуть бути заблоковані: Telegram має сувору політику щодо спаму та неправомірних дій ботів. Якщо ваш бот порушує правила Telegram, він може бути заблокований або вилучений без попередження.

Обмежений доступ до даних: Telegram Bot API не надає повний доступ до даних користувачів, що може бути недостатньо для деяких бізнес-цілей. Наприклад, ви не можете отримати доступ до історії чату користувача або його контактів.[25]

3.5 Висновки

Проаналізувавши компонентну базу для розробки системи було обрано наступні програмні та апаратні засоби:

OpenCV в якості бібліотеки комп'ютерного зору. Основними факторами вибору були досвід у використанні в минулому, можливість використовувати навчені іншими членами спільноти моделі, простоту у використанні.

Telegram messenger як засіб повідомлення користувача про детектування. Дану платформу було обрано через її популярність серед населення, велику кількість документації для створення ботів, простоту їх створення, специфічну орієнтованість інших платформ.

Raspberry Pi в якості основи для побудови системи як найкращу за характеристиками ціна-якість. Але за відсутності платформи було використано домашній ноутбук.

РОЗДІЛ 4

БУДОВА СИСТЕМИ МОНІТОРИНГУ

Для початку роботи було встановлено та налагоджено OpenCV для роботи із зображенням з відеокамери. Далі навчено модель для розпізнавання власного обличчя, логіка роботи зображена на рисунку 4.1.

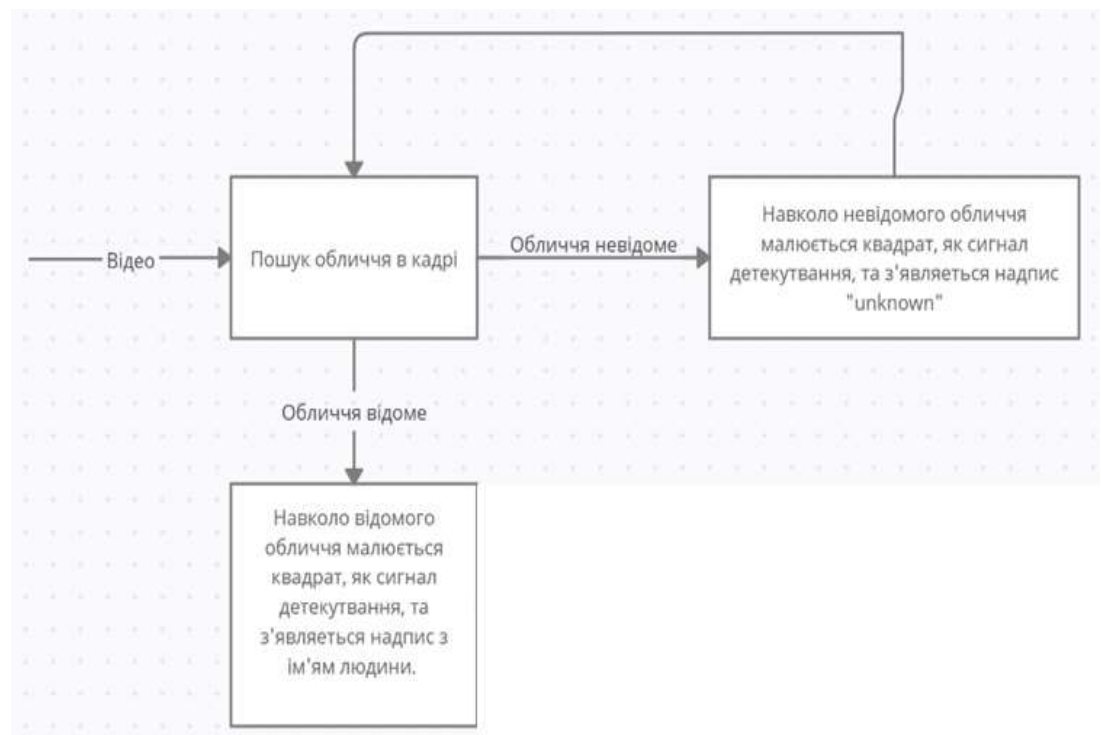


Рис. 4.1 Логіка роботи моделі розпізнавання

Наступним кроком було створено телеграм бота, та інтегровано в навчену модель. Функціонал телеграм бота наступний: при знаходженні невідомої особи у кадрі система робить знімки екрану та надсилає в чат, після цього знімки видаляються. Основний функціонал наведено на рисунку 4.2.

```

90     #If someone in your dataset is identified, print their name on the screen
91     if currentname != name:
92         start_time = time.time()
93         currentname = name
94         print(currentname)
95     if time.time() - start_time < 5:
96         screenshot = pyautogui.screenshot()
97         filename = 'screenshot.png'
98         screenshot.save(filename)
99         with open(filename, 'rb') as f:
100             bot.send_document(chat_id='-940964130', document=f)
101             os.remove(filename)
102     # update the list of names
103     names.append(name)

```

Рис.4.2 Основний функціонал телеграм бота

На рисунку 4.3 зображено приклад, на якому зображено успішне детектування та відправлення знімку екрану в чат. Для полегшення тестування системи було змінено тригер спрацювання з невідомої особи на відому.

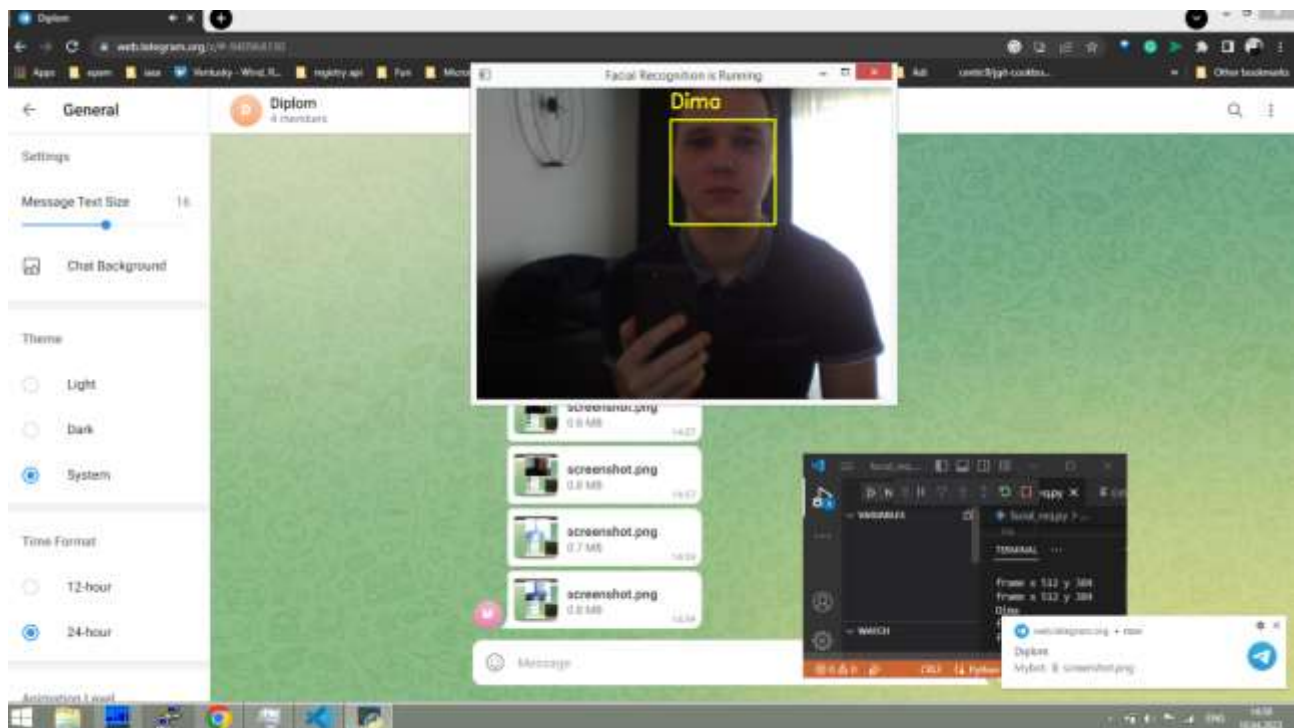


Рис. 4.3 Відпрацювання детектування та сповіщення через чат телеграм

Як додаткове джерело сповіщення про детектування було вирішено використовувати смс повідомлення. Для цього було підключено смартфон на ОС

Android з увімкненим режимом розробника в якості gsm модему, але не до пристрою, де відбувається детектування.

Для автоматизації відправки повідомлень було створено другого бота телеграм, що спрацьовує при знаходженні в чаті файлу із знімком екрану, тобто після спрацювання детектування особи та відпрацювання першого бота. На рисунках 4.4 та 4.5 зображено команди, що надсилаються модему, та функціонал другого бота відповідно.

```
import serial
import time

# Налаштування для підключення до модему через COM-порт
ser = serial.Serial('COM4', 9600, timeout=5)

# Очікування готовності модему
ser.write(b'AT\r\n')
response = ser.read(1000)
print(response)

# Встановлення текстового режиму SMS
ser.write(b'AT+CMGF=1\r\n')
response = ser.read(1000)
print(response)

# Встановлення адресата
ser.write(b'AT+CMGS="+380961112011"\r\n')
time.sleep(1)

# Ввід тексту повідомлення
ser.write(b'Hello, World! \x1a')
response = ser.read(1000)
print(response)

# Закінчення з'єднання з модемом
ser.close()
```

Рис. 4.4 AT команди для gsm модему

```

import telegram
import subprocess
from telegram.ext import Updater, CommandHandler, MessageHandler, Filters

# Функція для запуску скрипту
def run_script():
    subprocess.run(["python", "C:/test/run.py"])

# Функція для обробки повідомлень з файлами
def handle_file(update, context):
    file = context.bot.get_file(update.message.document.file_id)
    if file.file_path.endswith('.jpg') or file.file_path.endswith('.png'):
        run_script()

# Функція для обробки помилок
def error(update, context):
    print(f"Update {update} caused error {context.error}")

# Ініціалізація бота
from telegram.utils.request import Request

request = Request(con_pool_size=8)
bot = telegram.Bot(token='-----', request=request)
updater = Updater(bot=bot, use_context=True)

# Встановлення обробки повідомлень
updater.dispatcher.add_handler(MessageHandler(Filters.document, handle_file))
updater.dispatcher.add_error_handler(error)

# Запуск бота
updater.start_polling()
updater.idle()

```

Рис. 4.5 Функціонал другого боту

Як результат повна система працює наступним чином. У відео ведеться пошук обличчя людей, якщо особа невідома – нічого не стається. Якщо відбувається детектування – система робить серію знімків екрану, що надсилає в чат телеграм, при появі цих знімків в чаті подається команда на gsm модем щодо відправки смс повідомлення на заданий номер.

Варто зазначити, що при необхідності можна виконувати детектування та відправлення смс на одній машині, таким чином можна сповіщати користувача про детектування незалежно від наявності інтернет з'єднання на пристрої та від стану роботи серверів Telegram.

ВИСНОВКИ

В ході виконання дипломної роботи було успішно розв'язано актуальне науково-технічне завдання щодо розробки системи моніторингу території з використанням комп'ютерного зору та можливістю її швидкої модернізації.

Перед початком роботи був проведений аналіз ринку, в результаті якого були виявлені слабкі місця існуючих на ринку систем безпеки. Також було проаналізовано матеріально-технічну базу для розробки власної системи моніторингу.

Для реалізації системи була обрана бібліотека комп'ютерного зору OpenCV, яка забезпечує ефективний аналіз відео та виявлення обличч людей. Цей вибір дозволив забезпечити потрібну функціональність системи.

З метою сповіщення користувача про виявлення об'єктів моніторингу був обраний месенджер Telegram, який надає можливість надсилати знімки екрану та інформацію про подію. Крім того, як резервний варіант сповіщення, було обрано SMS повідомлення, яке також може бути використане для повідомлення користувача.

В результаті роботи була повністю реалізована система моніторингу території, яка була піддана випробуванням. При виявленні об'єктів моніторингу, зокрема обличч людей, система успішно надсилає знімки екрану в чат Telegram та відправляє SMS повідомлення на попередньо встановлений номер.

Окрім цього, реалізована система має потенціал для швидкої модернізації, що відкриває перспективи для подальшого розвитку. Наприклад, можливість вдосконалення алгоритмів виявлення, розширення функціональності для виявлення інших типів об'єктів або інтеграція з іншими системами безпеки.

Експериментальна перевірка розробленої системи підтвердила її працездатність та надійність. Система успішно виявляла обличчя людей на відео та надсилала сповіщення у вигляді знімків екрану через месенджер Telegram та SMS повідомлення.

Отже, виконана робота відповідає поставленим завданням дослідження, а сама розроблена система моніторингу території з використанням комп'ютерного зору здатна підвищити ефективність моніторингових процесів в середньоціновому сегменті. Її використання може бути корисним для забезпечення безпеки на різних об'єктах, таких як приватні домогосподарства, невеликі підприємства або громадські місця.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Азарний Д.І. Розпізнавання обличь для системи розумного будинку / Д.І. Азарний, Г.Б. Жиров // Військова освіта та наука: сьогодення та майбутнє: XVIII Міжнародна науково-практична конференція, 25 листопада 2022 р.: тези доповіді. – Київ: ВІ КНУ, 2022. – С. 233.
2. <https://euristiq.com/real-time-video-monitoring/>(Last accessed: 12.04.2023).
3. <https://www.instructables.com/Face-detection-and-tracking-with-Arduino-and-OpenC/> (Last accessed: 01.04.2023).
4. <https://opencv.org/courses/> (Last accessed: 14.03.2023).
5. Backstop Media, Rick Waldron. Make: JavaScript Robotics Building NodeBots with Johnny-Five, Raspberry Pi, Arduino, and BeagleBone. Oreilly, 2018. 272с.
6. <https://ajax.systems/ua/products/motioncam-outdoor/>(Last accessed: 18.04.2023).
7. <https://ajax.systems/ua/products/starterkit-cam/>(Last accessed: 18.04.2023).
8. <https://www.tutorialspoint.com/opencv/index.htm> (Last accessed: 25.02.2023).
9. <https://www.geeksforgeeks.org/opencv-overview/> (Last accessed: 28.03.2023).
10. Forsyth, D. A., & Ponce, J. Computer vision: a modern approach. Prentice Hall. 2012. 800с.
11. Szeliski, R. . Computer vision: algorithms and applications. Springer.2011.812с.
12. <https://www.howtogeek.com/427897/how-doesfacial-recognition-work/>(Last accessed: 10.03.2023).
13. https://docs.opencv.org/3.4/db/d28/tutorial_cascade_classifier.html(Last accessed: 25.02.2023).
14. P. Viola and M.J. Jones, «Robust real-time face detection», International Journal of Computer Vision, vol. 57, no. 2, 2004., pp.137–154
15. Р.Гонсалес, Р.Вудс, «Цифрова обробка зображення», ISBN 5-94836-028-8, 2005. – 1072 с.

16. Yan Šochman, Jiří Matas, «AdaBoost», Center for Machine Perception, Czech Technical University, Prague, 2010
17. Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *Pattern Analysis and Machine Intelligence*, IEEE Transactions on, 35(8):1872–1886, 2013.
18. N. Srivastava, G. E. Hinton et al., Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014. URL: <https://www.cs.toronto.edu/~hinton/absps/JMLRdropout.pdf>
19. <https://www.mathworks.com/products/computer-vision.html>(Last accessed: 11.03.2023).
20. <http://simplecv.org/>(Last accessed: 11.03.2023).
21. <https://caffe.berkeleyvision.org/>(Last accessed: 11.03.2023).
22. <https://www.tensorflow.org/>(Last accessed: 11.03.2023).
23. <https://pytorch.org/>(Last accessed: 11.03.2023).
24. <https://www.pcmag.com/picks/the-best-windows-mini-pcs/>(Last accessed: 18.04.2023).
25. https://en.wikipedia.org/wiki/Comparison_of_user_features_of_messaging_platforms/(Last accessed: 19.04.2023).