

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Приладобудівний факультет

Кафедра автоматизації та систем неруйнівного контролю

До захисту допущено:

Завідувач кафедри

_____ **Юрій КИРИЧУК**

«__» _____ 20__ р.

Дипломна робота

на здобуття ступеня бакалавра

за освітньо-професійною програмою «Комп'ютерно-інтегровані системи та технології в приладобудуванні»

спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології»

на тему: «Нейромережевий модуль для сервісного робота»

Виконав (-ла):

студент (-ка) IV курсу, групи ПК-01

Сакута Анастасія Анатоліївна _____

Керівник:

доцент, к.т.н., доцент,

Галаган Роман Михайлович _____

Рецензент:

старший викладач, доктор філософії (PhD)

Дорожинська Ганна Василівна _____

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент (-ка) _____

Київ – 2024 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Приладобудівний факультет
Кафедра автоматизації та систем неруйнівного контролю

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 151 «Автоматизація та комп'ютерно-інтегровані технології»

Освітньо-професійна програма «Комп'ютерно-інтегровані системи та технології в приладобудуванні»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Юрій КИРИЧУК

«__» _____ 20__ р.

ЗАВДАННЯ

на дипломну роботу студенту

Сакута Анастасія Анатоліївна

1. Тема роботи «Нейромережевий модуль для сервісного робота», керівник роботи Галаган Роман Михайлович, доцент, к.т.н., доцент, затверджені наказом по університету від «28» травня 2024 р. №2121-с
2. Термін подання студентом роботи 04 червня 2024р.
3. Вихідні дані до роботи: програмне забезпечення - Python, бібліотеки - OpenCV, DeepFace, gTTS, os, time, плата керування - Raspberry pi
4. Зміст роботи: РЕФЕРАТ, АБСТРАКТ, СПИСОК СКОРОЧЕНИХ НАЗВ, ВСТУП, 1 РОЗДІЛ 1: СЕРВІСНІ РОБОТІВ ТА РОБОТОТЕХНІКА, 2 РОЗДІЛ 2: ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ РОБОТОТЕХНІЧНИХ СИСТЕМ ТА ОГЛЯД БІБЛІОТЕК PУTHON, 3 РОЗДІЛ 3: НЕЙРОМЕРЕЖЕВИЙ МОДУЛЬ ДЛЯ СЕРВІСНОГО РОБОТА, ВИСНОВКИ, СПИСОК ЛІТЕРАТУРИ, ДОДАТОК А: ЗАГАЛЬНИЙ КОД ПРОГРАМНОГО МОДУЛЯ, ДОДАТОК Б: БЛОК-СХЕМА АЛГОРИТМУ КОДУ.
5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): Загальна кількість ілюстрацій у роботі - 10, разом з плакатами та презентацією – 16

6. Дата видачі завдання 14 травня 2024 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання етапів роботи	Примітка
1	Дослідження матеріалів про робототехніку	15.04.24	
2	Дослідження матеріалів про нейромережі та їх застосунок у ПЗ для роботів	28.04.24	
3	Пошук аналогів створюваного модулю	05.05.24	
4	Опис принципу роботи робота та підбір модулів для коду	08.05.24	
5	Написання окремих модулів, що відповідають за одну з функцій робота	13.05.24	
6	Створення алгоритму роботи повноцінного коду	16.05.24	
7	Вибір створених програмних модулів та поєднання їх у кінцеве програмне забезпечення	20.05.24	
8	Тестування роботи програмного модуля	27.05.24	
9	Оформлення роботи та підведення висновків	29.05.24	

Студент

Анастасія САКУТА

Керівник

Роман ГАЛАГАН

РЕФЕРАТ

Дана робота присвячена розробці програмного забезпечення, а саме нейромережевого модуля, для розпізнавання облич, емоцій та спілкування з людьми, а також для подальшого створення роботизованої системи (сервісного робота), що базується на використанні плати Raspberry Pi. В основі даного забезпечення лежить використання мови програмування Python та застосування таких бібліотек, як OpenCV, DeepFace - для обробки відеопотоку та розпізнаванню емоцій, для озвучуванню тексту – бібліотека gTTS. Аналізуючи функціонал даного програмного забезпечення можна припустити, що даний модуль з деяким вдосконаленням може використовуватись для більш складних сервісних роботів, що в майбутньому зможуть значно спростити життя людям. Загальна кількість сторінок 64, кількість ілюстрацій у роботі - 10, разом з плакатами – 15 та додатково є презентація.

ABSTRACT

This work is devoted to the development of software, namely a neural network module, for recognising faces, emotions and communicating with people, as well as for the further creation of a robotic system (service robot) based on the use of the Raspberry Pi board. This software is based on the Python programming language and the use of such libraries as OpenCV, DeepFace for video stream processing and emotion recognition, and the gTTS library for text-to-speech. Analysing the functionality of this software, it can be assumed that this module, with some improvements, can be used for more complex service robots that will greatly simplify people's lives in the future. The total number of pages is 64, the number of illustrations in the work is 10, together with posters - 15, and there is additionally a presentation.

ЗМІСТ

РЕФЕРАТ	4
ABSTRACT	5
СПИСОК СКОРОЧЕНИХ НАЗВ	8
ВСТУП	10
1 РОЗДІЛ 1: СЕРВІСНІ РОБОТІВ ТА РОБОТОТЕХНІКА	12
1.1 Визначення робототехніки.....	12
1.2 Що ж таке робот?	12
1.3 Компоненти робота.....	13
1.4 Класифікації.....	15
1.5 Сервісний робот	18
1.6 Різниця між сервісним і промисловим роботом	19
1.7 Сфери використання	20
1.8 Переваги та недоліки сервісних роботів.....	22
1.9 Приклади використання сервісних роботів.....	23
1.10 ПЗ та операційні системи роботів	25
1.11 Висновки	26
2 РОЗДІЛ 2: ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ РОБОТОТЕХНІЧНИХ СИСТЕМ ТА ОГЛЯД БІБЛІОТЕК PYTHON	27
2.1 Програмне забезпечення: визначення, суть, види	27
2.2 Нейронні мережі.....	29
2.2.1 Визначення нейронної мережі	29
2.2.2 Як працюють нейронні мережі?	30
2.2.3 Робота нейронної мережі.....	30

2.2.4 Навчання нейронної мережі	33
2.2.5 Види нейронних мереж	34
2.3 Бібліотеки Python	35
2.3.1 Бібліотека OpenCV	36
2.3.2 Бібліотека DeepFace	37
2.3.3 Бібліотека gTTS	38
2.3.4 Бібліотека SpeechRecognition	39
2.3.5 Бібліотека pyttsx3	39
2.3.6 Бібліотека API ChatGPT	40
2.3.7 Бібліотека Google Cloud Dialogflow	41
2.3.8 Бібліотека ChatterBot	42
2.4 Висновок	42
3 РОЗДІЛ 3: НЕЙРОМЕРЕЖЕВИЙ МОДУЛЬ ДЛЯ СЕРВІСНОГО РОБОТА	44
3.1 Основні модулі	44
3.1.1 Визначення емоцій та обробка зображень	44
3.1.2 Розпізнавання голосу	46
3.1.3 Озвучування тексту	47
3.1.4 Інтеграція чат-ботів зі штучним інтелектом	49
3.2 Реалізація програмного забезпечення та його алгоритм роботи	50
ВИСНОВКИ	52
СПИСОК ЛІТЕРАТУРИ	54
ДОДАТОК А: ЗАГАЛЬНИЙ КОД ПРОГРАМНОГО МОДУЛЯ	59
ДОДАТОК Б: БЛОК-СХЕМА АЛГОРИТМУ КОДУ	60

СПИСОК СКОРОЧЕНИХ НАЗВ

ABB – Asea Brown Boveri

API – Інтерфейс прикладного програмування (Application Programming Interface)

ASIMO – Advanced Step in Innovative Mobility

CNN – Convolutional Neural Network (Згорткова нейронна мережа)

FPGA – Field-Programmable Gate Array

GPU – Graphics Processing Unit

GTTS – Google Text-to-Speech

IFR – International Federation of Robotics (Міжнародна Федерація Робототехніки)

KRL – KUKA Robot Language

LSTM – Long Short-Term Memory (Довга короткочасна пам'ять)

MLP – Multilayer Perceptron (Багатошаровий перцептрон)

MSE – Mean Squared Error (середня квадратична похибка)

NLP – Обробка природної мови (natural language processing)

nsss – NS (NextStep) Speech Synthesizer

OpenCV – Open Source Computer Vision Library

Pyttsx3 – Python Text-to-Speech, Version 3

QNX – Quick Unix

RAPID – Robot Application Programming Interface for Developers

ReLU – Rectified linear unit (випрямлена лінійна одиниця)

RNN – Recurrent Neural Network (Рекурентна нейронна мережа)

ROS – Robot Operating System

RPA – Robotic Process Automation

RTOS – Real-Time Operating System

sapi5 – Speech Application Programming Interface, Version 5

SIMD – Single Instruction Multiple Data (одночасне виконання однієї інструкції над кількома даними)

SNN – Simulated neural networks (симульовані нейронні мережі).

TTS – Text-to-speech

ПЗ – програмне забезпечення

ШІ – штучний інтелект

ШНМ – штучні нейронні мережі

ВСТУП

У сучасному світі, де технології стрімко розвиваються, робототехніка займає центральне місце як у нашому повсякденному житті, так і у різних галузях промисловості. Завдяки поєднанню інженерних рішень із передовими технологіями, сервісні роботи стають все більш розповсюдженими і невід'ємною частиною нашого соціального середовища.

Нейромережі в цьому контексті виступають не лише як інструмент обробки даних, але й як ключовий компонент у створенні ефективних та розумних роботів. Їх здатність до аналізу величезних обсягів інформації, виявлення залежностей, та швидкість навчання роблять їх незамінними в областях, де потрібен високий ступінь автоматизації та автономності.

У рамках даного проекту, ми пропонуємо розробку нейромережевого модулю для сервісного робота, який має за мету спростити та поліпшити взаємодію між людьми та роботом. Наш проект спрямований на реалізацію функцій розпізнавання облич та усної реакції на емоції. Такий підхід дозволяє створити інтерактивну та ефективну систему взаємодії з користувачем, яка може використовуватися у різних сферах: від розважальних до промислових застосувань.

За основу розробки взято мову програмування Python, яка завдяки своїй простоті та широкій підтримці відкритих бібліотек стала найбільш популярною у сфері машинного навчання та обробки даних. Для обробки відеопотоку та розпізнавання облич людей використовуються бібліотека OpenCV, що дозволяє здійснювати швидко та ефективно обробку зображень та відео, та бібліотека DeepFace для розпізнавання осіб та їх емоцій на фотографіях з високою точністю. Для озвучування тексту, а точніше реакцій на емоції використовується бібліотека gTTS.

На початок проекту планувалось зробити більший функціонал модуля, та під час роботи було прийнято спростити код.

Загалом, цей проект відкриває широкі перспективи для застосування нейромережевих технологій у сфері робототехніки та робить взаємодію з роботами

більш інтуїтивною та ефективною. Наша робота може стати важливим кроком у розвитку індустрії сервісних роботів та сприяти створенню нових рішень, які полегшать життя людям та підвищать рівень комфорту в їхньому повсякденності.

1 РОЗДІЛ 1: СЕРВІСНІ РОБОТІВ ТА РОБОТОТЕХНІКА

1.1 Визначення робототехніки

Робототехніка – напрямок інженерії, що допомагає втілити в життя фантастичні фільми про майбутнє. Це одна з передових галузей у інженерії. Робототехніка тільки починає набирати популярність в Україні та світі, проте у світі використання цієї галузі зустрічається частіше. Наприклад, лідерами у цій галузі є США, Китай, Німеччина, Японія та Південна Корея. [1] Причина ж повільного розвитку робототехніки – нестача фінансування проєктів у цій сфері.

Робототехніка – наука практичного застосування, яка має зв'язок з проектуванням, конструюванням та експлуатацією роботів. Іншими словами, робототехніка, походячи від слова "робот" і "техніка", це наука, що створює автоматизовані технічні системи, тобто роботів. Вона спрямована на розробку роботів і робототехнічних систем для автоматизації складних технологічних процесів і операцій, включаючи виконання небезпечних і важких завдань, які раніше виконували люди. Термін "робототехніка" був введений письменником-фантастом Айзеком Азімовим у 1940-х роках. [2]

Робототехніка вирішує ряд техніко-економічних проблем для підприємств:

- Редукція робочих місць
- Економія на виробництві
- Високоякісна продукція
- Швидкість та точність виконання завдань
- Автоматизація процесів чи виробництва, така як у Тесла в США
- Підвищення продуктивності
- Швидке впровадження нових технологій у надекстремальних умовах

1.2 Що ж таке робот?

Робот — це автоматичний пристрій, який використовується для виконання завдань або надання підтримки в різних сферах. Їхня розробка спрямована на

автоматизацію та спрощення рутинних чи складних завдань, що колись вимагали людської присутності.

Письменник-фантаст Айзек Азімов встановив три закони робототехніки для регулювання поведінки "розумних" машин. Перший закон стверджує, що робот не може завдати шкоди людині або, своєю бездіяльністю, допустити, щоб людині було завдано шкоди. Другий закон вимагає, щоб робот слухався наказів людини, за винятком ситуацій, які суперечать першому закону. Третій закон стверджує, що робот повинен захищати самого себе, якщо це не суперечить першим двом законам. [2,3]

Наукова фантастика часто стає реальністю, так сталося і у випадку роботів. Королівське товариство та Британська академія об'єднали зусилля для формулювання одного принципу, що керуватиме системами штучного інтелекту (ШІ) - "Поліпшувати добробут людини". Цей принцип визначає, що ШІ має допомагати і підвищувати якість життя людей, забезпечуючи їхній добробут та комфорт. [2,4]

Роботи мають потенціал революціонізувати багато аспектів сучасного світу, від оптимізації промислової ефективності до покращення нашого повсякденного життя.

1.3 Компоненти робота

Загальна структура робота складається з п'яти основних систем [5,6]:

- Система прийняття рішень (система керування, по суті «мозок» системи)
- Інформаційно-вимірювальна (сенсорна) система
- Система зв'язку
- Система енергопостачання
- Виконавча система
- Механіка робота (каркас, «скелет»)

Система прийняття рішень, відома як інформаційно-керівна система, відіграє ключову роль у функціонуванні робота, будучи його "мозком". Основні завдання цієї системи включають керування виконавчими механізмами, опрацювання та аналіз сигналів від сенсорних систем та організацію взаємодії з людьми. Зазвичай цей

"мозок" представлений мікропроцесором, мікроконтролером або програмованою логічною схемою (FPGA), які функціонують на певній програмі. Таким чином, система прийняття рішень не лише складається з електронних компонентів, але й визначається програмним забезпеченням, що визначає його поведінку. [5,6]

Інформаційно-вимірювальна система робота відповідає за сприйняття зовнішнього світу і визначається як сукупність "органів чуття". Ця система надає роботу здатність орієнтуватися в просторі. Ці "органи чуття" відомі також як датчики або сенсори. Така система є ключовою складовою робота, оскільки вона забезпечує збір інформації про навколишнє середовище, необхідну для прийняття рішень та взаємодії з навколишнім середовищем. [5,6]

Система зв'язку забезпечує взаємодію робота з іншими пристроями і оператором, що дозволяє формулювати завдання, контролювати функціонування, діагностувати несправності та проводити регламентні перевірки. Це забезпечує людині (оператору) постійний контроль над роботом. Оператор втручається, коли автономна робота неможлива або недоцільна, наприклад, у невідомих умовах, які не передбачені програмою, або в ситуаціях, що вимагають прийняття рішення людиною. Також втручання оператора може бути необхідним, коли робот діє відповідно до жорсткої програми та потребує команди від людини для виконання певних дій. [5,6]

Система енергопостачання включає в себе різноманітні компоненти, такі як елементи живлення, стабілізатори живлення, різні генератори електроенергії та батареї. Наприклад, до складу можуть входити різні типи генераторів, такі як бензинові, дизельні або гібридні, які забезпечують постійний потік електроенергії для роботи системи. Крім того, для забезпечення стабільності напруги та потужності можуть бути використані стабілізатори живлення. У деяких випадках використовуються альтернативні джерела енергії, такі як сонячні батареї, які перетворюють сонячне випромінювання на електричну енергію. Всі ці компоненти спільно забезпечують надійне живлення для робототехнічних систем, що забезпечує їх безперебійну роботу. [5,6]

Виконавча система робота забезпечує всі його рухи та визначає його можливості виконання фізичних дій чи переміщення в просторі. Вона включає в себе різноманітні компоненти, такі як маніпулятори, ходова частина, пристрої позиціонування робочих елементів тощо. Електричні двигуни є основою більшості механізмів виконавчої системи. Маніпулятори відповідають за переміщення та управління об'єктами, тоді як ходова частина дозволяє роботів пересуватись у просторі. Пристрої позиціонування робочих елементів забезпечують точність та контроль над рухами. Усі ці компоненти спільно забезпечують роботів можливість взаємодіяти з оточуючим середовищем та виконувати завдання, які він був призначений виконувати. [5,6]

Механіка робота, його скелет, складається з різноманітних елементів конструкції, які відіграють ключову роль у забезпеченні його функціональності. Ця конструкція включає в себе основні компоненти, такі як колеса, ноги та корпус, до яких кріпляться та з'єднуються всі раніше розглянуті складові робота. Важливо зазначити, що конструкція визначає як зовнішній вигляд, так і функціональність робота. Для роботів, які виконують фізичні рухи, механіка часто є так само складною або навіть більш складною, ніж електроніка, оскільки вона відповідає за забезпечення маневреності та стійкості пристрою під час виконання різноманітних завдань. [5,6]

Система робота складається з декількох ключових компонентів, які взаємодіють між собою для забезпечення правильної роботи пристрою. Ці складові включають інформаційно-вимірювальну систему, яка сприймає оточуючий світ, систему прийняття рішень, що визначає відповідні дії, систему зв'язку для взаємодії з іншими пристроями та оператором, систему енергопостачання для живлення, виконавчу систему, що здійснює фізичні дії, та механіку робота, яка визначає його структуру та маневреність. Ці компоненти спільно працюють, щоб забезпечити ефективну та надійну роботу пристрою у різних сценаріях використання.

1.4 Класифікації

Існує безліч різноманітних класифікацій роботів. Розглянемо основні з них.

Класифікація роботів за міжнародним стандартом ISO 8373:2021 [6] визначає два основних типи: промислові та сервісні. Промислові роботи призначені для автоматизації виробництва та використовуються у виробничих цехах, тоді як сервісні роботи призначені для обслуговування та виконання рутинних завдань за межами виробничих просторів. Однак у сучасному світі існують інші типи роботів, такі як побутові, дослідницькі, вчительські, військові, медичні, аптечні, роботи для забезпечення безпеки, соціальні, розважальні, андроїди та інші. Ця розширена класифікація допомагає фахівцям у точній ідентифікації різних типів роботів та їх використання у різних сферах діяльності. [5, 40, 41]

Міжнародна Федерація Робототехніки (IFR) [7] розширює класифікацію сервісних роботів, враховуючи їхнє призначення. Особисті сервісні роботи призначені для використання в повсякденному житті, такі як кухонні помічники, роботи-пилососи, доглядальниці та вихованці. Вони спрощують побутові обов'язки і полегшують життя людей. Професійні сервісні роботи призначені для надання різноманітних послуг і забезпечення доходу, наприклад, роботи-консультанти, адміністратори, гіді, кур'єри та роботи-діагности. Ці роботи використовуються в різних сферах, де вони здатні ефективно виконувати рутинні завдання та забезпечувати високий рівень обслуговування. Розширена класифікація допомагає краще розуміти різноманітність сервісних роботів і їхнє використання у побуті та професійній діяльності, сприяючи розвитку сучасних технологій та поліпшенню якості життя. [5, 40, 41]

З історії створення:

Роботи класифікуються за трьома поколіннями залежно від їхньої функціональності та рівня автономності.

Перше покоління роботів, розроблене в 1960-х, обмежувалося виконанням програмованих послідовностей дій і не могло пристосовуватися до змін у середовищі. Ці роботи потребували жорсткого програмування оператором і перепрограмування при зміні завдань. Хоча їх простота дозволяла використовувати їх в окремих процесах, їх функціональність була обмежена. [5,8]

Друге покоління роботів, яке з'явилося пізніше, мали гнучкі програми, що дозволяли їм адаптуватися до змін у середовищі. Вони вміли вибирати оптимальні алгоритми, але все ще потребували участі людини в програмуванні. [5,8]

Третє покоління роботів, найбільш розвинене, має самонавчальні системи, що дозволяють їм адаптуватися і вдосконалюватися без втручання людини. Ці роботи можуть розуміти мову, вести діалог з людьми, навчатися на досвіді та вирішувати інтелектуальні завдання. Вони пристосовані до роботи в умовах неповної інформації та можуть функціонувати автономно. [5,8]

Системи програмного керування роботами можна розділити на різні типи в залежності від рівня гнучкості їхньої програми.

Перший тип - це жорсткопрограмовані системи, які є найбільш поширеними. У таких системах фіксований набір команд, який залишається незмінним протягом всього процесу роботи. Вони не мають можливості адаптуватися до змінних умов, тому не можуть коригувати свої дії у реальному часі. [5]

Другий тип - це адаптивні системи, які отримують інформацію про зовнішнє середовище за допомогою датчиків. Такі системи можуть адаптуватися до змін умов, що дозволяє їм працювати більш ефективно і точно. [5,9]

Третій тип - це гнучкопрограмовані системи, які швидко змінюють програму ґрунтуючись тільки на меті і інформації про зовнішнє середовище, яка надходить від датчиків. Ці системи є найбільш гнучкими і адаптивними, оскільки вони можуть реагувати на зміни умов і самостійно коригувати свою роботу в реальному часі з метою досягнення поставленої мети. [5,9]

Роботи за типом управління можна класифікувати на три категорії: автономні, напівавтономні та керовані.

Автономні роботи: Ці роботи діють повністю самостійно без прямого втручання людини під час виконання завдань. Вони використовують вбудовані алгоритми та сенсори для сприйняття оточуючого середовища та прийняття рішень. [5]

Напівавтономні роботи: У цих роботів є певний рівень автономності, але вони все ж таки потребують деякого рівня контролю або втручання людини. Наприклад, людина може задавати певні параметри або визначати загальну стратегію дій робота, але сам робот виконує конкретні завдання автономно. [5]

Керовані роботи: Ці роботи повністю залежать від людського керівництва. Керовані роботи виконують дії лише під час отримання відповідних команд або сигналів від оператора. Вони не мають власної автономності і не можуть приймати рішень без прямого керівництва людини. [5]

Класифікація роботів за способом пересування включає такі типи: Колісні використовують колеса для переміщення і можуть мати різну кількість коліс для забезпечення стійкості та маневреності. Гусеничні роботи мають гусеничну систему, яка дозволяє їм працювати на нерівному або складному місцевості, наприклад, в лісах або на піщаних дюнах. Роботи, що крокують, моделюють рухи людини і застосовуються для андроїдів. Для забезпечення більшої стабільності частіше використовуються три- або чотириногі конструкції. До групи літаючих роботів належать дрони, безпілотні літаки і ракети. Плаваючі роботи призначені для переміщення на воді або під нею за допомогою сили вітру або гвинтів. Робот-гексапод має шість рухливих ланок, що дозволяють йому виконувати різноманітні рухи та маневри. Однак для забезпечення його руху достатньо лише трьох робочих ланок, оскільки вони працюють синхронно та координовано, забезпечуючи стійкість та ефективність пересування робота. Ця конструкція дозволяє економити ресурси та забезпечує оптимальний баланс між функціональністю та витратами енергії. [5]

1.5 Сервісний робот

Сервісний робот - це автономна машина або механізм, призначений для виконання різноманітних корисних завдань для людей. Ці роботи можуть виконувати рутинні, трудомісткі або небезпечні операції, які важко або неприємно виконувати людині. Вони можуть використовуватися в різних сферах, включаючи медицину, готельний бізнес, логістику, побутові послуги та багато інших. Сервісні роботи

можуть мати різні ступені автономності, від повної автономії, коли вони працюють без участі людини, до часткової автономії, коли вони взаємодіють з людьми під час виконання завдань. Основна мета сервісних роботів - полегшити життя людей, звільнивши їх від монотонних або небезпечних робіт і покращити ефективність різних сфер людської діяльності.

Згідно з принципами Міжнародної організації зі стандартизації, сервісний робот має за мету полегшити життя людини, звільнивши її від виконання різних корисних завдань. Ці роботи працюють автономно, завдяки вбудованій системі керування, яка може бути налаштована на різні режими, від повної автономії до часткової. [10, 11]

На відміну від традиційних роботів, що використовуються в промислових цілях, сервісні роботи в першу чергу призначені для підтримки людей у повсякденному житті. У широкому розумінні сервісний робот охоплює широкий спектр застосувань і типів роботів, починаючи від роботів для логістики та доставки, соціальних роботів, роботів-прибиральників, роботів для дезінфекції, роботів-кухарів/кухонних роботів, роботів-офіціантів/ресторанних роботів, сільськогосподарських роботів і підводних роботів. Незважаючи на те, що ринок сервісних роботів знаходиться на набагато більш ранній стадії розвитку, ніж традиційні промислові роботи, в цьому просторі зростає кількість зусиль для сприяння впровадженню сервісних роботів. Інтенсивність конкуренції та стадія розробки значно варіюється залежно від сфери застосування. [12]

1.6 Різниця між сервісним і промисловим роботом

Промислові та сервісні роботи мають різні основні сфери застосування, хоча деякі вони можуть перетинатися. Промислові роботи використовуються переважно для виробничих завдань або в тих сферах, де робота може бути небезпечною для людей. Вони зазвичай замінюють людей у цих ролях. У свою чергу, сервісні роботи мають більш різноманітне застосування, включаючи галузі охорони здоров'я, готельно-ресторанного бізнесу та прибирання. Їхні завдання часто є менш небезпечними, але можуть охоплювати широкий спектр діяльності, включаючи

виробничі процеси. Важливо відзначити, що сервісні роботи призначені допомагати людям, а не повністю замінити їх на робочих місцях. [13]

Промислові роботи вже мають довгу історію та є відносно зрілою технологією. Їхній розвиток завжди був тісно пов'язаний з виробництвом, що є основним стимулом для їхнього вдосконалення. Наприклад, перші промислові роботи з'явилися ще у 1950-х роках. Натомість, сервісні роботи є молодішою галуззю, і вони з'явилися приблизно у той же час. У порівнянні з промисловими аналогами, впровадження сервісних роботів може здаватися менш простим на перший погляд. Це може бути пов'язано з тим, що їхнє застосування часто вимагає розв'язання складних завдань інтеракції з різними середовищами та умовами, що призводить до повільнішого їх поширення і використання.[13]

Промислові роботи, як правило, мають обмежену мобільність і гнучкість, оскільки вони запрограмовані на виконання обмеженого набору завдань. Зазвичай вони призначені для виконання конкретних дій в межах виробничого процесу. У порівнянні з ними, сервісні роботи, які спеціально створені для взаємодії з людьми, мають більшу універсальність та безпечність. Вони розроблені з урахуванням потреб та безпеки користувачів, що робить їх більш придатними для використання у відкритих просторах або серед людей. Більшість сервісних роботів також характеризуються вищою рівнем мобільності, що дозволяє їм легше адаптуватися до різних умов і виконувати різноманітні завдання. [13]

1.7 Сфери використання

Робототехніка розширюється на різні сфери життя, включаючи промисловість, медицину, обслуговування, транспорт, освіту та інші. Вона включає в себе використання роботів та автоматизованих систем, що активно впроваджуються для полегшення та прискорення процесів. Промислові роботи, наприклад, роботи-маніпулятори та автономні робочі одиниці, відіграють ключову роль у виробничих процесах, сприяючи підвищенню ефективності та швидкості виробництва. [11, 14]

У сфері медицини роботи стають невід'ємною частиною хірургічних операцій, забезпечуючи неймовірну точність та меншу інвазію, що виходить за межі здібностей навіть найкращого хірурга. В області обслуговування вони виявляються незамінними помічниками, де роботи-офіціанти та автоматизовані системи підтримують обслуговування у ресторанах, готелях і магазинах. Технології автономної доставки, які поширюються все ширше, забезпечують ефективну та оперативну розноску товарів. А в домашньому господарстві роботи-пилотяги використовуються для підтримання чистоти і порядку в приміщеннях. [10, 11]

Професійне прибирання отримує новий стандарт з появою роботів-прибиральників та роботів-дезінфекторів, особливо під час пандемії Covid-19. Це відчутно в усіх сферах, де можливе використання таких технологій. Чистота та дезінфекція значною мірою допомагають стримувати поширення інфекцій. [15]

Роботи відправляються в умови, недоступні для людей, як от найглибший океан чи найвища гора. Вони виконують завдання з перевірки та технічного обслуговування в суворих умовах та небезпечних місцях, де людська праця була б складною. [15]

Будівельна галузь також зазнає змін завдяки роботам. Зведення та знесення споруд вимагає значної робочої сили, яку не завжди легко забезпечити. Мобільні роботи, такі як ті, що використовуються для будівельної документації, транспорту, 3D-друку, свердління чи навіть зварювання, стають більш ефективними та менш залежними від погодних умов порівняно з людською працею. [15]

Важливість соціальної взаємодії та освіти зросла під час пандемії Covid-19 та заходів соціального дистанціювання. Спеціальні соціальні роботи стали невід'ємною частиною цього процесу, надаючи можливість взаємодії без необхідності фізичного контакту. У сфері освіти роботи служать платформою для експериментів та навчання з робототехніки, спрощуючи доступ до цієї області без потреби спеціалізованої підготовки. [15]

Щодо догляду в домашніх умовах, роботи допомагають людям із віковими обмеженнями або інвалідністю в їхньому повсякденному житті. Вони надають

підтримку у пересуванні та маніпуляціях, зменшуючи залежність від сторонньої допомоги та сприяючи самостійності. [15]

1.8 Переваги та недоліки сервісних роботів

Переваги використання сервісних роботів:

- Ці роботи відзначаються вищою гнучкістю в роботі.
- Вони посилюють збір даних і ефективно проводять аналіз.
- Роботи сприяють самопомозі в різних справах, полегшуючи життя людини.
- Вони зменшують навантаження та робочий стрес людей.
- Надійність робіт підвищується завдяки ефективності в програмуванні та виконанні протоколів.
- Роботи допомагають у більшій ясності та ефективності надання послуг.

Недоліки використання сервісних роботів:

- Ці роботи зменшують кількість людських робочих місць.
- Іноді можуть виникати ситуації, коли ці роботи працюють в неправильному напрямку.
- Помилки програмування можуть виникати під час обробки роботами.
- Вони потребують постійного оновлення та модернізації їхнього програмного та апаратного забезпечення
- Вартість експлуатації висока.
- Обслуговування цих роботів є витратним і вимагає своєчасної заміни.

Кожен випадок може мати свої унікальні особливості, що впливають на результат. Наприклад, у сфері промисловості, де потрібна велика точність та швидкість, використання роботів-маніпуляторів може значно підвищити продуктивність та знизити ризики для працівників. У сфері медицини, автономні роботи можуть забезпечити більш точне та менш інвазивне хірургічне втручання, що призводить до поліпшення результатів лікування та зниження часу відновлення пацієнтів. Однак, важливо враховувати, що успішне впровадження роботів потребує

великої уваги до деталей, технічної підготовки та забезпечення безпеки відповідно до конкретних умов і вимог. [16]

1.9 Приклади використання сервісних роботів

У листопаді 2000 року компанія Honda Motor Co., Ltd. оголосила про створення нового гуманоїдного робота під назвою ASIMO (рис. 1.1). Цей робот використовує нову технологію ходьби, розроблену Honda, що надає йому можливість ходити з унікальною схожістю до людини. Унікальні можливості ASIMO були вперше представлені у штаб-квартирі Honda в Токіо. [17]

Boston Dynamics Spot [18] - це робот на чотирьох ногах, розроблений компанією Boston Dynamics. Він має компактну конструкцію і вражаючі можливості, включаючи сканування та картографування оточуючого середовища, уникнення перешкод, сходження по сходах, відкривання дверей та інші завдання. Spot може використовуватися як у промислових, так і в побутових цілях, допомагаючи в різних ситуаціях, від пошуку та рятування до інспекційних робіт та допомоги в побуті.



Рисунок 1.1 - ASIMO[17]

SoftBank Robotics Pepper - це робот, призначений для взаємодії з людьми в різних сферах, таких як бізнес, освіта, торгівля тощо. Він має за мету створення дружнього

та привітного інтерфейсу для комунікації з користувачами, надання їм інформації, допомоги та розваги. Pepper (рис. 1.2) здатний розпізнавати людські емоції, реагувати на команди та взаємодіяти з оточуючим середовищем. Його основна функція - створювати позитивний та персоналізований досвід для користувачів. [19]

Роботи Zena Rx від Aethon (рис. 1.3) надійно доставляють аптечні, лабораторні та інші клінічні матеріали.[20] Робот Zena Rx від Aethon є частиною серії роботів, які спеціалізуються на автономному транспортуванні ліків та інших медичних припасів у лікарнях та інших медичних установах. Ці роботи призначені для полегшення та автоматизації процесу доставки ліків і медичних матеріалів до пацієнтів та медичного персоналу. Завдяки своїй автономності, роботи Zena Rx здатні самостійно навігувати в медичних приміщеннях, уникати перешкод та забезпечувати ефективну та безпечну доставку потрібних матеріалів за запитом. Це дозволяє медичному персоналу зосередитися на більш важливих завданнях, забезпечуючи в той же час швидку та надійну обслуговування пацієнтів.



Рисунок 1.2 - SoftBank Robotics Pepper

[19]



Рисунок 1.3 - Zena Rx від Aethon[20]

Існує ще безліч інших і різноманітних сервісних роботів, що використовуються у різних сферах та мають різні призначення. Вище ж зазначено одні з найпопулярніших роботів, які вважають вдалим проектом та які використовуються і зараз.

1.10 ПЗ та операційні системи роботів

Операційні системи та програмне забезпечення для робототехнічних систем є ключовими елементами в розвитку робототехніки. Вони забезпечують роботу, координацію та управління роботами, роблячи їх більш ефективними, гнучкими та надійними. Найпопулярніші з операційних систем - ROS (Robot Operating System), ROS 2, VxWorks, QNX, Linux, тощо.

ROS - це одна з найпопулярніших відкритих операційних систем для розробки роботів. ROS надає інструменти та бібліотеки для створення комплексних робототехнічних програм. Вона підтримує безліч роботів та сенсорів і широко використовується в наукових дослідженнях та комерційних проектах. Також серед особливостей можна вирізнити: підтримку розподілених обчислень, багатомовність (підтримка C++, Python), велику спільноту розробників, численні пакети для різних задач (навігація, маніпуляція, сприйняття). [21]

ROS2 - це наступне покоління ROS, яке пропонує поліпшену продуктивність, безпеку і можливість роботи в реальному часі. ROS 2 підтримує більш широке коло додатків, включаючи промислові роботи. Особливості: поліпшена підтримка багатоплатформності, більш висока безпека, підтримка систем реального часу, краща підтримка багаторівневих систем. [22]

VxWorks [23] - промислова операційна система реального часу (RTOS - real-time operating system), розроблена компанією Wind River та яка використовується в робототехніці для забезпечення високої надійності та продуктивності. VxWorks широко застосовується в критично важливих системах, таких як робототехніка, авіація та космічна техніка. Особливості: надійність, висока продуктивність, підтримка багатоядерних процесорів, широке застосування в аерокосмічній та оборонній промисловості.

QNX [24] - інша RTOS, яка використовується для створення систем з високими вимогами до надійності та продуктивності. QNX часто використовується в автомобільній промисловості, медичних пристроях та робототехніці. Особливості:

мікроядерна архітектура, висока надійність, підтримка стандартів безпеки, використання в критичних системах.

Linux [25] є популярною операційною системою для робототехніки завдяки своїй гнучкості, надійності та підтримці широкого кола апаратних платформ. Багато робототехнічних фреймворків, таких як ROS, працюють на базі Linux. Linux - відкрите ПЗ, велика спільнота розробників, підтримка широкого спектру апаратних платформ, можливість налаштування під конкретні завдання.

1.11 Висновки

Отже, інтеграція робототехніки у різні сфери життя та подальший розвиток інтелектуальних можливостей роботів відкриває безмежні можливості для вдосконалення та підвищення ефективності різних процесів.

Існує кілька видів роботів, найпопулярніші з них – сервісні та промислові. Також існують різноманітні класифікації робототехнічних систем, які розрізняють їхні підвиди або ж виокремлюють інші типи роботів залежно від їх призначення або будови. Наведено основні відмінності між промисловим і сервісним роботом, оскільки їх дуже часто плутають.

Сервісні роботи наразі дуже популярні, а тому є багато прикладів їх успішного інтегрування майже у всі сфери діяльності: починаючи від побуту та закінчуючи проведенням складних операцій. Наведено основні переваги і недоліки сервісних роботів. Найпоширеніші та найуспішніші проекти, на мою думку – це ASIMO від Honda, Spot від Boston Dynamics, Pepper від SoftBank Robotics та Zena Rx від Aethon.

Основні операційні системи для робототехнічних систем включають ROS (Robot Operating System), ROS 2, VxWorks, QNX та Linux. Зазначено, що найбільшою проблемою у розвитку сервісних роботів наразі є їх інтелектуалізація. Саме тому метою моєї роботи буде створення "мозку" робота, тобто програмного забезпечення. В процесі виготовлення робототехнічної системи найбільше часу та уваги приділяється саме розробці такого програмного модуля.

2 РОЗДІЛ 2: ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ РОБОТОТЕХНІЧНИХ СИСТЕМ ТА ОГЛЯД БІБЛІОТЕК PYTHON

2.1 Програмне забезпечення: визначення, суть, види

У загальному випадку програмне забезпечення (ПЗ) - це набір програм і даних, які вказують комп'ютеру, як виконувати певні завдання. Зазвичай воно включає супровідну програмну документацію. Програмне забезпечення відрізняється від апаратного забезпечення, яке складає фізичну частину системи та виконує безпосередню роботу. [26] Для робототехнічних систем теж створюються ПЗ і їх визначення схоже на те, що вказано для комп'ютера. Програмне забезпечення для роботів представляє собою набір програмних команд або інструкцій, що передаються механічним пристроям і електронним системам, відомим як роботи, для виконання певних завдань. Це програмне забезпечення використовується для автономного виконання різноманітних завдань. На ринку існує безліч програмних систем і фреймворків, спрямованих на полегшення програмування роботів. [27]

Програмне забезпечення – це одна з найважливіших складових, якій потрібно приділяти значну увагу при створенні робота. Воно є аналогом людського розуму і внутрішнього "Я", яке визначає, коли і як виконувати ті чи інші дії.

Кожен тип робота потребує спеціалізованого програмного забезпечення, яке враховує його унікальні функції та операційне середовище. Промислові роботи, наприклад, повинні мати високоточні алгоритми для виконання складних виробничих процесів, таких як складання, зварювання або фарбування. Їх програмне забезпечення повинно забезпечувати точність, надійність і безпеку під час роботи.

Сервісні роботи, з іншого боку, орієнтовані на взаємодію з людьми та виконання повсякденних завдань. Їх програмне забезпечення повинно включати алгоритми машинного навчання та обробки зображень для розпізнавання об'єктів, навігації в середовищі, уникнення перешкод та адаптації до нових умов. Такі роботи повинні бути інтуїтивно зрозумілими і безпечними для користувачів.

Класифікація програмного забезпечення (ПЗ) для роботів може бути здійснена за кількома критеріями: за функціональністю, за рівнем абстракції, за типом робототехнічної системи, за сферою застосування та за типом взаємодії з користувачем. У попередньому розділі ми розглянули кілька видів операційних систем.

Типи програмного забезпечення для роботизованих систем включають [28]:

Операційні системи робота: наприклад, ROS [21]. Як зазначалось вище, є відкритою системою, яка надає набір інструментів та бібліотек для розробки програмного забезпечення для роботів. Вона включає в себе такі компоненти, як системи управління роботами, візуалізацію даних, моделі об'єктів та багато іншого. ROS широко використовується в дослідженнях з робототехніки та індустрії. Програмне забезпечення для автономного програмування: дозволяє програмувати промислових роботів без фізичного підключення до робота. [28]

Програмне забезпечення для автономного програмування: Цей тип ПЗ дозволяє розробникам створювати програми для роботів, які можуть працювати автономно без постійного з'єднання з комп'ютером. Вони можуть бути програмовані для виконання різних завдань, включаючи навігацію, розпізнавання об'єктів та взаємодію з оточуючим середовищем. [28]

Симулятори: Ці інструменти дозволяють розробникам тестувати програмне забезпечення для роботів у віртуальних середовищах, що дозволяє економити час і кошти, які зазвичай потрібні для фізичного тестування роботів. Вони надають можливість відтворювати реальні умови та сценарії для тестування алгоритмів і програмного забезпечення. [28]

Проміжне програмне забезпечення: Це ПЗ включає в себе різноманітні інструменти для забезпечення взаємодії та координації між різними компонентами робота, такими як датчики, актуатори та обчислювальні блоки. Воно може включати в себе середовища розробки, бібліотеки комунікації та інші компоненти, які полегшують розробку робототехнічних систем. [28]

Програмне забезпечення для планування мобільних роботів: Це ПЗ допомагає програмувати рухи та навігацію мобільних роботів у просторі. Воно може включати в себе алгоритми для планування маршруту, визначення оптимальних шляхів та уникнення перешкод. [28]

Роботизована автоматизація процесів (RPA - Robotic Process Automation): Це технологія програмного забезпечення, яка дозволяє створювати, розгортати та керувати програмними роботами, які автоматизують рутинні операції та взаємодіють з цифровими системами для підвищення ефективності бізнес-процесів. [28]

Ці програмні засоби забезпечують широкий спектр можливостей для розробників робототехнічних систем, дозволяючи їм створювати складні, високопродуктивні та надійні робототехнічні рішення.

2.2 Нейронні мережі

2.2.1 Визначення нейронної мережі

Нейронна мережа - це алгоритм або модель машинного навчання, яка функціонує на зразок людського мозку. Вона використовує процеси, що наслідують роботу біологічних нейронів, для розв'язання задач, ідентифікації важливих патернів та прийняття рішень на основі вхідних даних. Нейронна мережа складається з шарів вузлів, або штучних нейронів, які включають вхідний шар, один або кілька прихованих шарів та вихідний шар. Кожен вузол з'єднується з іншими, має власну вагу і поріг. Якщо вихідні дані вузла перевищують певний поріг, вузол активується і передає дані на наступний рівень мережі. Якщо ж ні, дані не передаються далі. [29]

Нейронні мережі використовують навчальні дані для самонавчання та підвищення точності. Після налаштування вони стають потужними інструментами в інформатиці та штучному інтелекті, здатними швидко класифікувати та кластеризувати дані. Завдання з розпізнавання мови або зображень, які вручну займали б години, можуть бути виконані за хвилини. Одним із найвідоміших прикладів застосування нейронних мереж є пошуковий алгоритм Google. [29]

Нейронні мережі також називають штучними нейронними мережами (ШНМ) або симульованими нейронними мережами (SNN). Вони є частиною машинного навчання і лежать в основі моделей глибокого навчання. [29, 40, 41]

2.2.2 Як працюють нейронні мережі?

Опишемо процес роботи нейронної мережі для класифікації електронної пошти. Спочатку, вхідний рівень приймає дані, такі як вміст листа, інформація про відправника та тему. Ці дані множаться на ваги та проходять через приховані шари мережі. Під час навчання мережа вивчає шаблони, що вказують на те, чи є лист спамом. Вихідний рівень використовує двійкову функцію активації, щоб передбачити, чи є лист спамом (1) або ні (0). Протягом процесу навчання мережа постійно уточнює свої ваги за допомогою зворотного поширення помилки, що дозволяє їй краще розрізняти між спамом та не спамом, і демонструє практичне застосування нейронних мереж у реальному житті, таких як фільтрація електронної пошти. [30] У контексті нейронної мережі для класифікації електронної пошти у прихованому шарі можна використати різні активаційні функції. Однією з найпоширеніших активаційних функцій для цього застосування є ReLU (Rectified Linear Unit) або її модифікації. ReLU - це нелінійна активаційна функція, яка відображає негативні значення в нуль, а позитивні значення залишає без змін.

Використання ReLU дозволяє моделі набувати нелінійність, що може бути корисним у виявленні складних шаблонів в даних

2.2.3 Робота нейронної мережі

Нейромережа - це як мозок, що навчається розпізнавати шаблони і робити прогнози на основі наданих даних. Зазвичай нейромережа у своєму складі має вхідний шар, один або декілька прихованих шарів і вихідний шар, що складається з шарів штучних нейронів, які з'єднуються (рис. 2.1). Навчання мережі має два основні процеси - зворотне проходження і пряме проходження. [30]

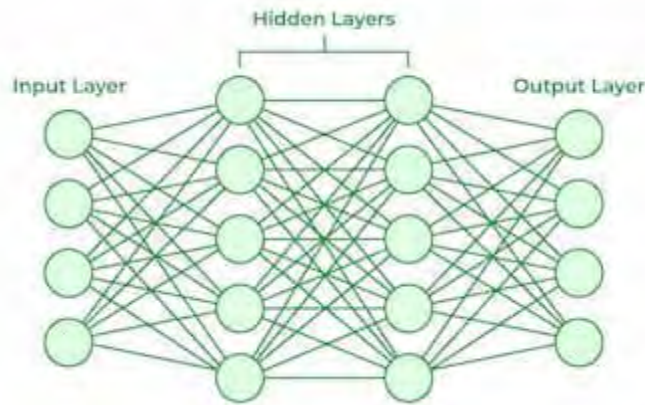


Рисунок 2.1 – Структура нейронної мережі [30]

Пряме проходження [29, 30]

Процес прямого поширення відбувається при подачі вхідних даних до нейронної мережі та обчисленні виходів мережі для заданих вхідних даних. Основні кроки прямого поширення:

- Подача вхідних даних: Вхідні дані подаються до вхідного шару мережі.
- Обчислення зваженої суми для кожного нейрона: Для кожного нейрона в кожному шарі обчислюється зважена сума за формулою 2.1 - вхідні значення, помножені на ваги нейрона, додаються, а потім додається зміщення.:

$$z = (x_1 * w_1) + (x_2 * w_2) + \dots + (x_n * w_n) + b \quad (2.1)$$

Тут:

x_1, x_2, \dots, x_n — вхідні значення.

w_1, w_2, \dots, w_n — ваги, пов'язані з цими вхідними значеннями.

b — зміщення (bias).

- Застосування активаційної функції: Результат зваженої суми передається через активаційну функцію, яка вводить нелінійність у вихідні дані нейрона.
- Передача вихідних значень до наступного шару: Отримані вихідні значення передаються в якості вхідних даних до наступного шару мережі.

- Повторення для кожного шару: Ці кроки повторюються для кожного шару мережі до досягнення вихідного шару.

Зворотне проходження

Зворотне поширення - це процес, який відбувається після прямого поширення, коли обчислені вихідні значення порівнюються зі справжніми мітками даних та обчислюється помилка прогнозування. На основі цієї помилки мережа коригує свої ваги та зміщення, щоб покращити точність прогнозування.

- Розрахунок втрат: Під час зворотного поширення витрати оцінюються за допомогою функції витрат (або функції втрат), яка вимірює різницю між прогнозованими вихідними значеннями мережі та справжніми мітками даних. Основна мета полягає в тому, щоб ця функція була мінімальною, щоб мережа здатна була здійснювати точні прогнози. Для задач регресії часто використовується середньо-квадратична помилка (MSE) [29, 30]:

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.2)$$

де y_i - справжні значення, \hat{y}_i - прогнозовані значення, а n - кількість прикладів у навчальному наборі.

- Градієнтний спуск: Це метод оптимізації, який використовується для оновлення параметрів мережі (ваги та зміщення) з метою мінімізації функції втрат (наприклад, середньоквадратичної помилки) шляхом зміни їх значень в напрямку, протилежному градієнту функції втрат. [29, 30]

- Налаштування ваг: Налаштування ваг відбувається через алгоритм оптимізації під час навчання мережі. Основна мета - мінімізувати функцію витрат, яка вимірює різницю між прогнозованими та справжніми значеннями. Процес включає пряме поширення для обчислення вихідних значень, обчислення втрат, зворотне поширення для розповсюдження помилки та оновлення ваг для зменшення втрат. Цикл повторюється для кожного навчального прикладу до досягнення певної точності або мінімізації витрат. [29, 30]

- Навчання: нейромережа навчається не тільки через повторне визначення ваг і використання активаційної функції, але й через цілий набір методів і кроків, що включають обчислення похибки, зворотне поширення похибки, регуляризацію та перевірку на валідаційних даних. Це комплексний процес, спрямований на оптимізацію ваг мережі для досягнення найкращих результатів на невидимих даних. [29, 30]

- Функції дії: Активаційна функція в нейронній мережі визначає, чи активується нейрон на основі зваженої суми його входів. Вона вводить нелінійність у модель, що дозволяє мережі навчатися і моделювати складніші залежності в даних. Серед функцій можна виокремити ReLU (Rectified Linear Unit), Tanh (гіперболічний тангенс), Softmax і Sigmoid. [29, 30]

2.2.4 Навчання нейронної мережі

1. Навчання з контрольованим навчанням

Навчання нейромереж з контрольованим навчанням - це процес, коли модель навчається на основі вхідних даних та відповідних міток. Модель коригується на основі різниці між її прогнозами та правильними відповідями. Це дозволяє точно контролювати процес навчання, але може вимагати великої кількості правильних міток та стикається з можливістю перенавчання.[29, 30]

2. Навчання за допомогою навчання без учителя

Навчання нейромереж без учителя - це процес навчання моделі без наявності супервізованих міток у вихідних даних. В цьому випадку мережа намагається виявити приховані структури або закономірності у наборі даних самостійно. Основна мета полягає в тому, щоб модель самостійно визначила корисні ознаки або групи подій, які можуть бути корисні для подальшого аналізу чи вирішення задачі. Навчання без учителя дозволяє отримувати нові знання з ненавчальних даних та використовувати їх у різних сферах, включаючи аналіз даних, кластеризацію, виявлення аномалій та побудову рекомендаційних систем. [29, 30]

3. Навчання з підкріпленням

Навчання нейромереж з підкріпленням - це тип навчання машинного навчання, в якому модель навчається шляхом взаємодії з навколишнім середовищем через проби й помилки. Основна ідея полягає в тому, щоб модель могла навчитися діяти в середовищі, вчачись на основі нагороди або покарання за свої дії.[29, 30]

2.2.5 Види нейронних мереж

Існує багато типів нейронних мереж, які можна використовувати.

Feedforward Networks (Прямопрямі мережі):

Це найпростіша форма нейронних мереж, де інформація рухається лише в одному напрямку - від вхідного до вихідного шару. Кожен нейрон в кожному шарі пов'язаний з кожним нейроном наступного шару, і він обчислює зважену суму вхідних сигналів та передає її через активаційну функцію. Feedforward мережі часто використовуються для задач розпізнавання образів, класифікації та прогнозування.

Багатошаровий перцептрон (MLP):

Це найпоширеніший тип нейронної мережі, який складається з одного або декількох прихованих шарів між вхідним та вихідним шарами. Кожен нейрон у прихованому шарі отримує вхід від усіх нейронів попереднього шару, обчислює зважену суму вхідних сигналів та передає її через активаційну функцію. MLP використовується для широкого спектру задач, включаючи класифікацію, регресію та апроксимацію функцій.

Згорткова нейронна мережа (CNN):

Це тип нейронної мережі, який спеціалізується на роботі з двовимірними структурованими даними, такими як зображення. CNN використовує згортки та пулінг для екстракції ознак з вхідних даних, забезпечуючи розпізнавання шаблонів та інваріантність до пересувань та масштабувань. Цей тип мережі широко використовується в задачах комп'ютерного зору, включаючи класифікацію зображень, виявлення об'єктів та сегментацію зображень.

Рекурентна нейронна мережа (RNN):

RNN призначена для роботи з послідовними даними, де важливий контекст залежить від попередніх елементів послідовності. Вона має зворотний зв'язок, що дозволяє їй зберігати інформацію про попередні стани та використовувати її для прийняття рішень у поточному моменті часу. RNN використовуються для різноманітних завдань, таких як машинний переклад, генерація тексту, аналіз відгуків тощо.

Довга короткочасна пам'ять (LSTM):

LSTM - це спеціальна форма RNN, призначена для управління проблемою зниклої (або вибухаючої) градієнта під час тренування. Вона має додаткові блоки пам'яті, які дозволяють зберігати, оновлювати та використовувати інформацію на протязі тривалого часу. LSTM широко використовується в задачах, де важлива довготривала залежність між елементами послідовності, таких як машинний переклад, генерація тексту та аналіз часових рядів.

Нейромережі, або нейронні мережі, - це моделі машинного навчання, що імітують роботу біологічних нейронів. Вони складаються з вхідного шару, одного або кількох прихованих шарів і вихідного шару. Нейрони у шарах з'єднані між собою, і кожен з них має свої ваги та пороги.

Нейромережі навчаються на даних, коригуючи свої параметри, щоб з часом підвищувати точність. Вони використовуються для розпізнавання зображень, обробки мови, прогнозування та класифікації даних. Нейромережі лежать в основі глибокого навчання, що дозволяє вирішувати складні завдання.

2.3 Бібліотеки Python

У цьому огляді розглядаються основні бібліотеки Python, які використовуються для розробки нейромережевого програмного модуля для сервісного робота, основна плата якого це Raspberry Pi. Raspberry Pi, популярний одноплатний комп'ютер, забезпечує достатню потужність і гнучкість для реалізації складних обчислювальних задач, включаючи проекти з машинного навчання та штучного інтелекту.

Використання Python, завдяки його простоті та багатому набору бібліотек, робить цей процес ефективним і зручним. [42]

Нейромеревеві програмні модулі дозволяють реалізовувати різні функції, такі як розпізнавання образів, обробка природної мови та інші завдання, пов'язані з інтелектуальним аналізом даних. Модульна структура проекту дозволяє легко додавати, змінювати або видаляти компоненти, що забезпечує масштабованість і спрощує подальший розвиток системи. [42]

2.3.1 Бібліотека OpenCV

OpenCV (Open Source Computer Vision Library) є потужним інструментом для комп'ютерного зору та обробки зображень. Вона має відкритий вихідний код і використовується для різноманітних завдань, включаючи розпізнавання облич, аналіз відео, виявлення об'єктів і багато іншого. OpenCV підтримує кілька мов програмування, таких як C++, Python, Java та MATLAB, і працює на різних операційних системах, включаючи Windows, Linux, macOS і Android. OpenCV включає функції для: [31,33]

- Обробки зображень (фільтрація, перетворення, морфологічні операції).
- Аналізу зображень (контурне виявлення, сегментація).
- Роботи з відео (обробка кадрів, відстеження об'єктів).
- Розпізнавання об'єктів (виявлення облич, очей, посмішок).
- Класифікації та кластеризації (алгоритми машинного навчання).

OpenCV оптимізована для роботи на різних апаратних платформах, включаючи процесори з підтримкою SIMD, графічні процесори (GPU) та інші прискорювачі. Можливість інтеграції з іншими бібліотеками і фреймворками, такими як TensorFlow, PyTorch, для побудови складних систем машинного навчання. [31,33]

Приклади використання OpenCV[31,33]:

- Розпізнавання обличчя: Використовується для ідентифікації осіб в зображеннях та відео.

- Аналіз відео: Застосовується для відстеження рухомих об'єктів, аналізу поведінки і виявлення подій.
- Автоматизоване управління транспортом: Використовується для розпізнавання дорожніх знаків, відстеження смуг руху, виявлення пішоходів.
- Медична діагностика: Застосовується для аналізу медичних зображень, виявлення патологій.

2.3.2 Бібліотека DeepFace

DeepFace - це бібліотека Python для роботи з обробкою облич у фотографіях. Вона базується на нейронних мережах і надає можливості для виконання різних завдань, пов'язаних з розпізнаванням облич, аналізом емоцій та визначенням статеві та вікової категорій осіб на фотографіях. DeepFace дозволяє виконувати ці завдання швидко і ефективно, забезпечуючи доступ до потужних моделей глибокого навчання, які були навчені на великих обсягах даних. Ця бібліотека знаходить широке застосування в області комп'ютерного зору, аналізу зображень та розпізнавання облич у різних сферах, включаючи соціальні мережі, безпеку, медицину та багато інших. [32]

Поміж інших цікавих особливостей DeepFace є:

1. DeepFace надає зручний та простий інтерфейс для використання своїх функцій, що робить його доступним для широкого кола користувачів, від початківців до досвідчених розробників.
2. Бібліотека дозволяє виконувати не лише розпізнавання облич, але й аналіз емоцій, визначення статі та віку, а також вирішення інших завдань, пов'язаних з обробкою зображень облич.
3. DeepFace базується на передових методах глибокого навчання, таких як згорткові нейронні мережі (CNN), які забезпечують високу точність і ефективність обробки зображень.

4. Ця бібліотека знаходить застосування у багатьох сферах, включаючи соціальні мережі (наприклад, автоматичне розміщення фотографій), безпеку (відслідковування обличчя на відео), медицину (діагностика за виразом обличчя) та інші.

5. DeepFace продовжує розвиватися і підтримується активними розробниками, що дозволяє вносити в неї нові функції та вдосконалення.

2.3.3 Бібліотека gTTS

GTTS (Google Text-to-Speech) - це простий і зручний інструмент для перетворення тексту на мовлення з використанням штучного інтелекту та голосових технологій Google. За допомогою GTTS можна легко генерувати аудіофайли з будь-якого тексту, а потім використовувати їх для програвання на пристроях або в проектах, де потрібно виводити мовлення на основі тексту. Цей інструмент дозволяє вибирати різні мови та голоси, а також налаштовувати параметри швидкості та інтонації мовлення. [34]

Окрім базового функціоналу перетворення тексту на мовлення, GTTS має кілька цікавих особливостей:[34]

1. GTTS підтримує багато мов, що дозволяє генерувати мовлення на різних мовах з однаковою легкістю.
2. Користувач може вибрати різні голоси для генерації мовлення, що дозволяє створювати різноманітні аудіофайли з різними акцентами та інтонацією.
3. GTTS дозволяє налаштовувати параметри мовлення, такі як швидкість вимови та паузи між словами, що дозволяє досягти більшої натуральності згенерованого мовлення.
4. Інтерфейс GTTS простий та інтуїтивно зрозумілий, що робить його дуже зручним для швидкої генерації мовлення з будь-якого тексту.

Загалом, GTTS є потужним інструментом для генерації мовлення з тексту з широким спектром можливостей та простотою використання.

2.3.4 Бібліотека `SpeechRecognition`

`SpeechRecognition` - це бібліотека для розпізнавання мовлення в програмах Python. Вона надає зручний інтерфейс для роботи з різними движками розпізнавання мовлення, такими як Google Speech Recognition, Sphinx, а також іншими сервісами розпізнавання мовлення через Інтернет. За допомогою `SpeechRecognition`, ви можете легко записувати аудіо з мікрофону, а потім перетворювати його на текст за допомогою різних движків розпізнавання мовлення. Це дозволяє створювати програми, які взаємодіють з користувачем за допомогою голосових команд або розпізнають мовлення з аудіофайлів. [35]

Цікавим фактом є те, що бібліотека `SpeechRecognition` підтримує різні мови і може використовуватись для розпізнавання мовлення на багатьох мовах світу. Крім того, вона дозволяє працювати з різними джерелами аудіо, такими як мікрофон, аудіофайли та аудіопотоки з Інтернету. Це робить її досить універсальним інструментом для створення програм з розпізнаванням мовлення.

2.3.5 Бібліотека `pyttsx3`

`Pyttsx3` — це Python-бібліотека для перетворення тексту в мову. Відмінністю цієї бібліотеки від інших є її можливість працювати в автономному режимі та підтримка як Python 2, так і Python 3. Щоб використовувати `pyttsx3`, програма викликає фабричну функцію `pyttsx3.init()`, яка повертає посилання на екземпляр двигуна `pyttsx3`. Це дуже зручний інструмент для перетворення введеного тексту на мову. [36]

Бібліотека `pyttsx3` підтримує два голоси: один жіночий і один чоловічий, які використовують механізм "sapi5" для Windows. Вона також підтримує три різні двигуни TTS (text-to-speech): [36]

- `sapi5` — SAPI5 на Windows
- `nsss` — NSSpeechSynthesizer на Mac OS X
- `espeak` — eSpeak на інших платформах

Pytsx3 дозволяє розробникам легко інтегрувати функції перетворення тексту в мову у свої додатки. Бібліотека працює без необхідності в інтернет-з'єднанні, що робить її ідеальним вибором для офлайн-додатків. Завдяки підтримці різних версій Python, pytsx3 є універсальним рішенням для різноманітних проектів.[36]

2.3.6 Бібліотека API ChatGPT

Бібліотека API ChatGPT[37] від OpenAI надає розробникам потужні інструменти для інтеграції можливостей генеративного штучного інтелекту у свої додатки. Вона дозволяє доступ до моделей GPT, які можуть генерувати текст на основі заданих підказок, що відкриває широкий спектр можливостей. Основні функції :

1. **Текстове завершення:** Моделі можуть створювати відповіді або продовження тексту, що корисно для автоматичного написання статей, листів або створення контенту.
2. **Чатботи:** API ідеально підходить для створення інтерактивних чатботів, які можуть вести природні діалоги з користувачами, забезпечуючи високий рівень взаємодії та підтримки.
3. **Аналіз тексту:** Можливості аналізу тексту дозволяють витягати ключові поняття, резюмувати довгі тексти, проводити класифікацію та здійснювати інші види аналізу даних.
4. **Персоналізація:** Розробники можуть налаштовувати поведінку моделей, щоб вони відповідали конкретним вимогам їхніх додатків, наприклад, для специфічних галузей або типів взаємодії.
5. **Підтримка багатьох мов:** Бібліотека підтримує різні мови, що дозволяє створювати багатомовні рішення та обслуговувати глобальну аудиторію.

Додаткові переваги включають:

- **Інтеграція з існуючими системами:** API легко інтегрується з існуючими програмами та інфраструктурою, що робить процес впровадження швидким та ефективним.

- Постійне оновлення та підтримка: OpenAI регулярно оновлює моделі та забезпечує підтримку для розробників, що гарантує актуальність та високий рівень продуктивності.

- Розширені можливості налаштування: Розробники можуть тонко налаштовувати параметри генерації тексту, такі як тон, стиль, контекст, щоб досягти бажаного результату.

API ChatGPT від OpenAI є незамінним інструментом для розробників, які прагнуть інтегрувати передові можливості штучного інтелекту у свої продукти, підвищуючи їх функціональність, зручність використання та загальну якість [37]

2.3.7 Бібліотека Google Cloud Dialogflow

Google Cloud Dialogflow [38] — це платформа для розробки чат-ботів і віртуальних асистентів, яка дозволяє створювати природні та інтуїтивно зрозумілі діалоги між користувачами та додатками, пристроями або іншими системами. Dialogflow використовує алгоритми NLP для розуміння та обробки природної мови користувачів, що дозволяє створювати більш реалістичні та ефективні взаємодії. Бібліотека легко інтегрується з іншими сервісами Google Cloud, що дозволяє використовувати потужні хмарні можливості для масштабування та управління ботами. Підтримка багатьох платформ, таких як Google Assistant, Facebook Messenger, Slack, і багато інших, що дозволяє створювати чат-ботів для різних каналів одночасно.

Dialogflow надає інтуїтивний інтерфейс для створення і налаштування діалогів, що робить процес розробки простим і доступним навіть для користувачів без глибоких технічних знань. Можливість інтеграції з сторонніми API та сервісами для розширення функціональності чат-ботів, наприклад, для доступу до баз даних або інших систем. Завдяки використанню машинного навчання, боти, створені на основі Dialogflow, можуть адаптуватися і вдосконалювати свої відповіді на основі попередніх взаємодій. Вбудовані засоби для забезпечення безпеки даних користувачів і надійної роботи ботів. [38]

Таким чином, Google Cloud Dialogflow є потужним інструментом для створення розумних та ефективних чат-ботів і віртуальних асистентів, що забезпечує багатофункціональність і зручність у використанні.

2.3.8 Бібліотека ChatterBot

ChatterBot — це Python-бібліотека для створення чат-ботів, які можуть навчатися та адаптуватися до нових ситуацій. Вона використовує машинне навчання для автоматичної генерації відповідей на основі даних, які вона отримує під час спілкування. [39]

Підтримує різні алгоритми машинного навчання для навчання моделей спілкування. Може збирати дані з різних джерел, таких як діалоги, чати та тексти, для вдосконалення відповідей. Простота інтеграції з існуючими проектами завдяки зрозумілому API. Може навчатися новим фразам та відповідям в реальному часі, покращуючи свої відповіді з часом. Підтримує кілька мов для створення багатомовних чат-ботів. Надає можливість налаштування та розширення для задоволення специфічних потреб. [39]

ChatterBot підходить для створення простих та середньо складних чат-ботів для веб-додатків, клієнтської підтримки, розважальних ботів та інших застосувань, де потрібна автоматизована взаємодія з користувачами. [39]

2.4 Висновок

Програмне забезпечення – одна з найважливіших частин для будь-якого багатофункціонального робота. Класифікація програмного забезпечення (ПЗ) для роботів може бути здійснена за функціональністю, за рівнем абстракції, за типом робототехнічної системи, за сферою застосування та за типом взаємодії з користувачем.

Дуже часто для збільшення функціоналу робототехнічної системи використовують інтегрування нейронних мереж у програмний модуль. Нейронна мережа - це алгоритм або модель машинного навчання, яка функціонує на зразок людського мозку. Нейронні мережі використовують навчальні дані для самонавчання

та підвищення точності. Після налаштування вони стають потужними інструментами в інформатиці та штучному інтелекті, здатними швидко класифікувати та кластеризувати дані. Навчання нейронних мереж може відбуватись різними методами: з контрольованим навчання, за допомогою навчання без учителя, з підкріпленням. Також існують різні види нейронних мереж: Feedforward Networks, багатошаровий перцептрон, згорткова нейронна мережа, рекурентна нейронна мережа, довга короткочасна пам'ять.

Розглянуто основні бібліотеки Python, що допомогли б розробити власний нейромережевий модуль, серед них OpenCV, DeepFace, SpeechRecognition pytsx3, gTTS, API ChatGPT, Google Cloud Dialogflow та ChatterBot. Оскільки ці бібліотеки використовують нейронні мережі для аналізу зображень (OpenCV та DeepFace), генерації відповідей (API ChatGPT, Google Cloud Dialogflow та ChatterBot) та синтезу мовлення (gTTS, SpeechRecognition, pytsx3), програмне забезпечення, яке їх застосовує, може бути класифіковане як нейромережеве. Використання таких інструментів свідчить про інтеграцію методів машинного навчання та глибокого навчання у вашому програмному забезпеченні.

3 РОЗДІЛ 3: НЕЙРОМЕРЕЖЕВИЙ МОДУЛЬ ДЛЯ СЕРВІСНОГО РОБОТА

3.1 Основні модулі

При створення програмного модулю було прийнято рішення розробити забезпечення за модульним принципом, аби в процесі подальшого вдосконалення програми та наступних розробок, модулі можна було б легко інтегрувати, несильно змінюючи попередньо написаний код. У попередньому розділі було описані одні з основних бібліотек, що підходили б для реалізації необхідного функціоналу робота.

3.1.1 Визначення емоцій та обробка зображень

Коли йде мова про обробку зображень, то без сумніву потрібно використовувати функціонал бібліотеки OpenCV. Вона пропонує широкий спектр функцій, доступна безкоштовно та з відкритим кодом, працює на різних платформах, має велику спільноту та легко інтегрується з іншими бібліотеками. Також Вона використовується багатьма провідними компаніями та дослідницькими лабораторіями. Існує багато навчальних посібників та ресурсів, які допоможуть вам розпочати роботу з OpenCV. OpenCV постійно оновлюється та вдосконалюється. Завдяки цим перевагам OpenCV є одним з найкращих варіантів для обробки зображень.

Під час роботи над визначенням емоцій, постало питання вибору між функціоналом бібліотек MediaPipe та DeepFace, оскільки це дві різні бібліотеки, які використовуються для обробки та аналізу відео і зображень, але вони мають різні цілі та функціональні можливості. Основні відмінності описані у таблиці 3.1

Після аналізу даних таблиці 3.1, можна прийти до висновку, що DeepFace є кращою для аналізу емоцій у реальному часі завдяки її спеціалізації на розпізнаванні обличч і аналізі характеристик обличчя, високій точності завдяки навчанню на великих наборах даних та здатності розпізнавати навіть мікро-вирази. MediaPipe, хоч і потужна для загальних задач обробки мультимедіа, не має такої ж глибокої спеціалізації на аналізі емоцій, як DeepFace.

Таблиця 3.1

Основні відмінності бібліотек MediaPipe та DeepFace

MediaPipe	DeepFace
Зосереджена на загальних задачах обробки зображень та відео в реальному часі.	Спеціалізується на розпізнаванні облич та аналізі їх характеристик.
Більш універсальна з функціями для відстеження різних частин тіла, жестів, об'єктів та обробки відео.	Головним чином використовується для ідентифікації облич та їх аналізу.
Використовує оптимізовані графи обробки даних для швидкої та ефективної роботи в реальному часі.	Включає кілька моделей глибокого навчання для досягнення високої точності в розпізнаванні облич.
Легка та ефективна.	Більш ресурсоемна.
Менш точна.	Більш точна та надійна.

Після вибору основних бібліотек, створено код, що в режимі реального часу визначає емоції людини (рис. 3.1). Основні етапи роботи коду:

1. Захоплення відеопотоку з веб-камери.
2. Перетворення кожного кадру у відтінки сірого для детекції облич.
3. Використання Haar Cascade для виявлення облич.
4. Аналіз емоцій на виявлених обличчях за допомогою DeepFace.
5. Відображення результатів аналізу на зображенні.
6. Повторення процесу для кожного кадру до натискання клавіші 'Enter'.

Результати роботи представлені на рисунку 3.2, в даному випадку представлено як визначено нейтральну емоцію, радість та подив.

для отримання доступу необхідно звернутись до авторів проекту

Рисунок 3.1 – Програмний модуль для визначення емоцій у реальному часі

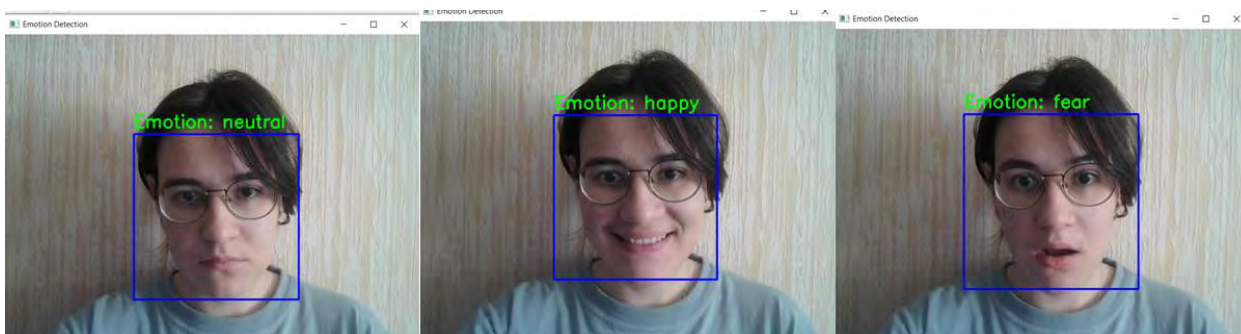


Рисунок 3.2 – Результат роботи коду

3.1.2 Розпізнавання голосу

Бібліотека `speech_recognition` є найкращим варіантом для розпізнавання голосу завдяки своїм численним перевагам, наприклад, вона підтримує декілька сервісів розпізнавання голосу, таких як Google Web Speech API, Microsoft Bing Voice Recognition, IBM Speech to Text, і інші. Простий та інтуїтивно зрозумілий API дозволяє швидко налаштувати розпізнавання голосу без потреби в глибоких знаннях. Підтримує роботу з аудіофайлами та потоковим аудіо, дозволяючи розпізнавати голос в реальному часі. Добре задокументована бібліотека з великою спільнотою користувачів, що полегшує вирішення проблем та впровадження нових функцій. Відкритий вихідний код, що дозволяє користувачам модифікувати та адаптувати бібліотеку під свої потреби.

Ці фактори роблять `speech_recognition` ідеальним вибором для даного проекту, де потрібне надійне та гнучке розпізнавання голосу. Загалом, `speech_recognition` є потужним та зручним інструментом для розпізнавання голосу, який може бути успішно використаний у широкому спектрі застосувань.

Було створено програмний модуль для розпізнавання мови (рис. 3.3), що в подальшому може бути інтегрований до основного коду. Принцип роботи дуже простий:

1. Коли ти запускаєш код, спершу виконується запис аудіо з мікрофону протягом 5 секунд.
2. Записане аудіо зберігається у файл "recorded_audio.wav".
3. Далі цей файл передається функції для розпізнавання мови, яка зчитує аудіофайл і намагається розпізнати текст у ньому за допомогою сервісу Google.
4. Якщо розпізнавання успішне, розпізнаний текст виводиться на екран, інакше виводиться повідомлення про помилку.

для отримання доступу необхідно звернутись до авторів проекту

Рисунок 3.3 – Програмний модуль Python для розпізнавання голосу

3.1.3 Озвучування тексту

Існує багато бібліотек, що допомагають у озвученні тексту, найпопулярніші з них це gtts, speech_recognition та pyttsx3. Розглянемо кожен з них. Основні відмінності представлені у таблиці 3.2

Хоча бібліотека pyttsx3 працює офлайн і не потребує інтернет-з'єднання, якість синтезу мови може бути нижчою, особливо для української мови. Це пов'язано з тим, що підтримка мов залежить від встановлених на системі мовних пакетів, які можуть не забезпечувати належну якість для української мови.

Бібліотека SpeechRecognition призначена для розпізнавання мови, а не для синтезу, як багато хто міг подумати. Вона не підходить для завдання озвучування тексту, оскільки її основна функція - перетворення аудіо в текст, а не навпаки.

Таблиця 3.2

Відмінності між бібліотеками gtts, speech_recognition та pyttsx3

Критерій	gTTS	SpeechRecognition	pyttsx3
Призначення	Перетворення тексту в аудіо (TTS - Text-to-Speech).	Розпізнавання мови (STT - Speech-to-Text).	Перетворення тексту в аудіо (TTS - Text-to-Speech).
Функціонал	<ul style="list-style-type: none"> •Перетворює введений текст на аудіо файл з використанням Google Translate Text-to-Speech API. •Підтримує багато мов, включаючи українську. •Може зберігати аудіо у файл або відтворювати його безпосередньо. 	<ul style="list-style-type: none"> •Розпізнає мову з аудіо файлів або мікрофону. •Підтримує різні сервіси розпізнавання мови (Google Web Speech API, Sphinx, Microsoft Bing Voice Recognition, тощо). •Підтримує багато мов, включаючи українську. 	<ul style="list-style-type: none"> •Офлайн текст-в-мову движок, який працює без інтернет-з'єднання. •Підтримує кілька мов, але якість та підтримка залежать від встановлених на системі мовних пакетів. •Може зберігати аудіо у файл або відтворювати його безпосередньо.
Переваги	<ul style="list-style-type: none"> •Висока якість синтезу мови завдяки використанню Google API. •Простота використання. •Підтримка різних мов і акцентів. 	<ul style="list-style-type: none"> •Гнучкість в використанні різних сервісів розпізнавання. •Можливість обробки аудіо файлів та потокового аудіо з мікрофону. 	<ul style="list-style-type: none"> •Не потребує інтернет-з'єднання, працює офлайн. •Підтримує різні синтезатори мови, залежно від операційної системи (наприклад, SAPI5 на Windows).
Недоліки	Потребує інтернет-з'єднання для роботи, оскільки використовує онлайн сервіс.	Потребує інтернет-з'єднання для деяких сервісів розпізнавання мови (наприклад, Google Web Speech API).	Обмежена підтримка української мови (залежить від системних налаштувань та встановлених мовних пакетів).

Враховуючи вище зазначене, gTTS є найкращим вибором для озвучування введеного тексту з клавіатури українською мовою, бо надає високу якість синтезу мови, простоту використання та підтримку української мови,. Приклад коду з реалізацією озвучення введеного тексту на рис. 3.4

для отримання доступу необхідно звернутись до авторів проекту

Рисунок 3.4 – Програмний модуль Python для озвучення введеного тексту

3.1.4 Інтеграція чат-ботів зі штучним інтелектом

Для генерації відповідей на питання користувачів, планувалось підключити до коду чат-бот. Вибір був серед API ChatGPT, ChatterBot та GoogleCloudDialogflow, оскільки ці чат-боти мають вбудовані засоби для забезпечення безпеки даних користувачів і надійної роботи ботів. Проте зараз використання API ChatGPT від OpenAI не є повністю безкоштовним і необмеженим. OpenAI надає певні безкоштовні квоти для користування API, але після досягнення цих лімітів необхідно сплачувати за подальше використання.

Google Cloud Dialogflow пропонує безкоштовний рівень використання, але він теж має певні обмеження. Користувачі можуть безкоштовно створювати та тестувати свої боти, але для більш масштабного або комерційного використання можуть знадобитися платні плани.

З бібліотекою ChatterBot були певні проблеми при встановленні, оскільки для цього потрібна вища версія Python, проте після оновлення, деякі інші модулі, що були створенні раніше, працюють некоректно.

Зважаючи все вищезазначене, було вирішено спростити функціонал програмного забезпечення до поки не буде знайдено можливість фінансування або подовження термінів безкоштовного використання одного з ботів. У майбутньому планується знайти кошти для можливості підключення даного модулю.

3.2 Реалізація програмного забезпечення та його алгоритм роботи

Проект був спрямований на реалізацію функцій розпізнавання облич, перетворення мовлення у текст, а також формулювання відповідей на запитання користувачів. Оскільки модуль з чат-ботом не вдалось підключити, модуль з розпізнаванням голосу теж став зайвим. Тепер же зібраний програмний код виявляє емоції на обличчях у режимі реального часу за допомогою камери. В коді використовується бібліотеку OpenCV для захоплення зображень і виявлення облич, DeepFace для аналізу емоцій, та gTTS для озвучення тексту на основі виявлених емоцій. Основна мета — відображення аудіо реакції на виявлені емоції. Загальний код представлений у додатку А.

Однак модулі, що не були інтегровані, лишаються робочими та повністю готовими до роботи. В подальшому, після вирішення проблеми з фінансуванням або після знаходження безкоштовного аналогу бота, планується доробити ПЗ та створити бажаний функціонал для робототехнічної системи.

Якщо описувати узагальнений алгоритм роботи реалізованого коду, то можна виділити такі етапи:

1. Ініціалізується камера та змінні.
2. Запускається безкінечний цикл для захоплення кадрів з камери.
3. Кожен кадр перетворюється у відтінки сірого та обробляється для виявлення облич.
4. Кожні 10 секунд проводиться аналіз емоцій на виявлених обличчях.
5. В залежності від виявлених емоцій, на зображенні відображається відповідний текст та озвучується аудіо.
6. Відображається оброблений кадр.
7. Якщо натиснута клавіша 'Enter', цикл завершується.
8. Закривається відеопотік та всі вікна.

Блок-схема алгоритму представлена у додатку Б (рис. Б.1- рис. Б.2).

Дана програма має реакції на 4 основних емоцій: радість, сум, злість та здивування. Якщо емоція не розпізнана, то буде озвучено фразу «Ця емоція ще потребує аналізу!». Реакція на емоцію – озвучення попередньо прописаних фраз – цікавих фактів.

Робота коду не зовсім досконала, бо виявлення тої чи іншої емоції залежить від якості камери робота, освітлення, кута сканування обличчя та контрастності на фоні, тобто якщо фон має різні малюнки з обличчями, людьми, тваринами тощо, то програма може ідентифікувати емоцію неправильно.

ВИСНОВКИ

У дипломній роботі на тему "Нейромережевий модуль для сервісного робота", було нейромережевий модуль для сервісного робота, реалізований на мові програмування Python. Основними бібліотеками, що використовуються у проекті, є OpenCV та DeepFace для обробки зображення та розпізнавання емоцій, а також gTTS (Google Text-to-Speech) для озвучування реакції на ті чи інші емоції.

На початку роботи основна ідея проекту полягала в тому, щоб створити систему, яка здатна розпізнавати обличчя людей, зчитувати їх голос, перетворювати його у текст, надсилати запити до чат-боту, наприклад, API ChatGPT для формулювання відповідей на запитання та озвучувати ці відповіді. Однак, під час реалізації проекту було вирішено зосередитися на створенні системи, що може відображати аудіо реакцію на виявлені емоції у режимі реального часу, оскільки модуль для підключення чат-ботів зі штучним інтелектом, на жаль, не вдалось реалізувати через обмеження у використанні безкоштовних квот, але після досягнення цих лімітів необхідно сплачувати за подальше використання. Проте навіть з цим функціоналом це дозволяє роботу не лише розпізнавати обличчя та емоції людей, але й відповідати на ці емоції в реальному часі, створюючи більш природну і динамічну взаємодію з користувачем.

Програма розпізнає та реагує на чотири основні емоції: радість, сум, злість та здивування. Реакція на емоцію полягає в озвученні заздалегідь прописаних фраз з цікавими фактами. Програма не є досконалою, оскільки точність розпізнавання емоцій залежить від деяких факторів. Це обмеження слід враховувати при використанні системи в різних умовах.

Дане програмне забезпечення побудоване на модульному принципі, що забезпечує багато переваг, серед яких легкість управління, зрозумілість коду, масштабованість, повторюваність, паралельна розробка, спрощене тестування, зменшення ризиків, легкість обслуговування, гнучкість та інтеграція. Цей підхід дозволяє створювати більш надійні, гнучкі та ефективні програмні рішення, які легше підтримувати та розвивати з часом.

Подальше вдосконалення модулю, а саме додавання можливості формулювання відповідей на запитання людини за допомогою чат-боту, може значно розширити функціональність робота, перетворюючи його на багатофункціональну інтерактивну систему. Такий робот може знайти застосування у різних галузях, від побутового обслуговування до надання інформаційних послуг.

Висока точність розпізнавання облич та емоцій, а також можливість реального часу реагувати на емоції користувача, роблять наш проект унікальним інструментом для створення інтелектуальних сервісних роботів. Це може значно покращити взаємодію між людиною і машиною, роблячи її більш природною та ефективною.

СПИСОК ЛІТЕРАТУРИ

- [1] Топ 10 світових лідерів у роботехніці у 2022 році [Електронний ресурс] // NanitRobot — Режим доступу: https://nanitrobot.com/top_10_leaders/ (дата звернення: 18.04.2024). — Назва з екрана.
- [2] Робототехніка. Штучний інтелект [Електронний ресурс] // Чернігівський обласний центр зайнятості. — Режим доступу: <https://chg.dcz.gov.ua/publikaciya/robototekhnika-shtuchnyy-intelekt> (дата звернення: 18.04.2024). — Назва з екрана.
- [3] Робот [Електронний ресурс] // Вікіпедія (wikipedia.org). — Режим доступу: <https://uk.wikipedia.org/wiki/%D0%A0%D0%BE%D0%B1%D0%BE%D1%82> (дата звернення: 18.04.2024). — Назва з екрана.
- [4] Єдиний закон робототехніки: поліпшити добробут людини [Електронний ресурс] // BBC NEWS Україна - Режим доступу: <https://www.bbc.com/ukrainian/features-40450278> (дата звернення: 18.04.2024). — Назва з екрана.
- [5] Види роботів і класифікація в робототехніці: всі області застосування і провідні виробники розумних машин [Електронний ресурс] // worldbank.org.ua - Режим доступу: <https://worldbank.org.ua/3720-vidi-robotiv-i-klasifikatsiya-v-robototekhnitsi.html> (дата звернення: 25.04.2024).
- [6] ISO 8373:2021 - Robotics — Vocabulary (вступив у дію з 2021 року на заміну ISO 8373:2012) [Текст]. — Чинний від 2021-11. — К.: ISO/TC 299, 2015. — 22 с.
- [7] Executive Summary WR 2021 Service Robots // [Електронний ресурс]. - Режим доступу: <https://ifr.org/>. (дата звернення: 25.04.2024).
- [8] Покоління роботів [Електронний ресурс]. //studfile.net - Режим доступу: <https://studfile.net/preview/7418766/page:19/> (дата звернення: 25.04.2024).
- [9] Промислові роботи [Електронний ресурс]// Кафедра автоматизації технологічних процесів та виробництв (tntu.edu.ua) - Режим доступу: <https://kaf-av.tntu.edu.ua/index.php/mn-abiturient/mn-articles/676-art-industrial-robots> (дата звернення: 25.04.2024).

[10] What Are Service Robots And How They Benefit Mankind? [Електронний ресурс]// Robots.net - Режим доступу: <https://robots.net/tech-reviews/what-are-service-robots/#:~:text=Types%20Of%20Service%20Robots%201%20Retail%20Service%20Robots,Service%20Robots%20...%206%20Event%20Service%20Robots%20> (дата

звернення: 09.05.2024).

[11] А.А. Сакута, доц. Галаган Р. М. Сервісні роботи та їх розвиток у світі: збірник праць XIX Всеукраїнської науково-практичної конференції студентів, аспірантів та молодих вчених “Ефективність та автоматизація інженерних рішень у приладобудуванні” 20-21 грудня 2023 р. — К.: ПБФ, КПІ ім. Ігоря Сікорського. — 2023. — 193-196 с.

[12] Service Robots 2022-2032: Technologies, Players & Markets [Електронний ресурс]// IDTechEx - Режим доступу: <https://www.idtechex.com/en/research-report/service-robots-2022-2032-technologies-players-and-markets/864> (дата звернення: 09.05.2024).

[13] What's the Difference Between Industrial Robots and Service Robots [Електронний ресурс]// robotlab.com - Режим доступу: <https://www.robotlab.com/group/blog/whats-the-difference-between-industrial-robots-and-service-robots#:~:text=In%20general%2C%20industrial%20robots%20are%20intended%20for%20manufacturing,uses%20in%20healthcare%2C%20hospitality%2C%20cleaning%2C%20and%20similar%20areas> (дата звернення: 09.05.2024).

[14] Долиненко В.В., Шаповалов Є.В., Скуба Т.Г., Коляда В.О., Куц Ю.В., Галаган Р.М., Карпінський В.В. Роботизована система неруйнівного вихрострумовеого контролю виробів зі складною геометрією. *Автоматическая сварка*. Київ. № 5-6 (764). — 2017 — ст. 60-67.

[15] International Federation of Robotics // [Електронний ресурс]. - Режим доступу: <https://ifr.org/wr-service-robots> (дата звернення: 13.05.2024).

[16] Service Robots [Електронний ресурс] // GeeksforGeeks- Режим доступу: <https://www.geeksforgeeks.org/service-robots/> (дата звернення: 13.05.2024).

[17] Honda Debuts New Humanoid Robot "ASIMO" [Электронный ресурс] // Honda Global | November 20, 2000 - Режим доступа: <https://global.honda/en/newsroom/news/2000/c001120b-eng.html> (дата звернения: 13.05.2024).

[18] Spot [Электронный ресурс] // Boston Dynamics - Режим доступа: <https://bostondynamics.com/products/spot/> (дата звернения: 13.05.2024).

[19] Meet Pepper: The Robot Built for People [Электронный ресурс] // SoftBank Robotics America - Режим доступа: <https://us.softbankrobotics.com/pepper> (дата звернения: 13.05.2024).

[20] Our Robots [Электронный ресурс] // Aethon - Режим доступа: <https://aethon.com/our-robots/> (дата звернения: 13.05.2024).

[21] ROS [Электронный ресурс] // Home - Режим доступа: <https://www.ros.org/> (дата звернения: 15.05.2024).

[22] ROS 2 Documentation [Электронный ресурс] // ROS 2 Documentation: Foxy documentation - Режим доступа: <https://docs.ros.org/en/foxy/index.html> (дата звернения: 15.05.2024).

[23] VxWorks [Электронный ресурс] // Industry Leading RTOS for Embedded Systems (windriver.com) - Режим доступа: <https://www.windriver.com/products/vxworks> (дата звернения: 15.05.2024).

[24] What is QNX? [Электронный ресурс] // Sealevel - Режим доступа: <https://www.sealevel.com/2023/01/02/what-is-qnx/> (дата звернения: 15.05.2024).

[25] The Complete Guide to Linux Fundamentals for Robotics [Электронный ресурс] // Automatic Addison - Режим доступа: <https://automaticaddison.com/the-complete-guide-to-linux-fundamentals-for-robotics/> (дата звернения: 15.05.2024).

[26] Software [Электронный ресурс] // Wikipedia - Режим доступа: <https://en.wikipedia.org/wiki/Software> (дата звернения: 17.05.2024).

[27] Robot software [Электронный ресурс] // Wikipedia - Режим доступа: https://en.wikipedia.org/wiki/Robot_software (дата звернения: 17.05.2024).

[28] 9 Types of Robotics Software You Might Consider for Your Robot [Електронний ресурс] // RoboDK blog - Режим доступу: <https://robodk.com/blog/robotics-software-to-consider/> (дата звернення: 17.05.2024).

[29] What is a Neural Network? [Електронний ресурс] // IBM - Режим доступу: <https://www.ibm.com/topics/neural-networks> (дата звернення: 17.05.2024).

[30] What is a neural network? [Електронний ресурс] // GeeksforGeeks - Режим доступу: <https://www.geeksforgeeks.org/neural-networks-a-beginners-guide/> (дата звернення: 17.05.2024).

[31] OpenCV [Електронний ресурс] // Open Computer Vision Library – Режим доступу: <https://opencv.org/> (дата звернення: 17.05.2024).

[32] DeepFace. [Електронний ресурс] – Режим доступу: [www.URL: https://github.com/serengil/deepface/blob/master/README.md](https://github.com/serengil/deepface/blob/master/README.md) (дата звернення: 17.05.2024).

[33] Розробка програм для цифрової обробки зображень з застосуванням OPENCV: навч. посіб. для студ. спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології» / М.В. Боровицький; КІП ім. Ігоря Сікорського, 2022. - 83с.

[34] gTTS [Електронний ресурс] // gTTS documentation – Режим доступу: <https://gtts.readthedocs.io/en/latest/> (дата звернення: 17.05.2024).

[35] SpeechRecognition [Електронний ресурс] // PyPI – Режим доступу: <https://pypi.org/project/SpeechRecognition/> (дата звернення: 17.05.2024).

[36] Python | Text to Speech by using pyttsx3 [Електронний ресурс] // GeeksforGeeks – Режим доступу: <https://www.geeksforgeeks.org/python-text-to-speech-by-using-pyttsx3/> (дата звернення: 17.05.2024).

[37] Using GPT-3.5 and GPT-4 via the OpenAI API in Python [Електронний ресурс] // DataCamp – Режим доступу: <https://www.datacamp.com/tutorial/using-gpt-models-via-the-openai-api-in-python> (дата звернення: 17.05.2024).

[38] dialogflow [Електронний ресурс] // PyPI – Режим доступу: <https://pypi.org/project/dialogflow/#:~:text=Project%20description%201%20Before%20you%20begin%20Select%20or,Contributing%20Contributions%20welcome%21%206%20License%20Apache%20Version%202.0> (дата звернення: 17.05.2024).

[39] About ChatterBot [Електронний ресурс] // ChatterBot 1.0.8 documentation – Режим доступу: <https://chatterbot.readthedocs.io/en/stable/> (дата звернення: 17.05.2024).

[40] Долиненко В.В. Роботизована система неруйнівного вихрострумовеого контролю виробів зі складною геометрією / В.В. Долиненко, Є.В. Шаповалов, Т.Г. Скуба, В.О. Коляда, Ю.В. Куц, Р.М. Галаган, В.В. Карпінський // Автоматическая сварка. – Київ. – 2017. - № 5-6 (764). – С. 60-67

[41] Skladchikov I. O., Momot A. S., Galagan R. M., Bohdan H. A., Trotsiuk K. M. Application of YOLOX deep learning model for automated object detection on thermograms. Information Extraction and Processing. 2022, 50(126), 69-77

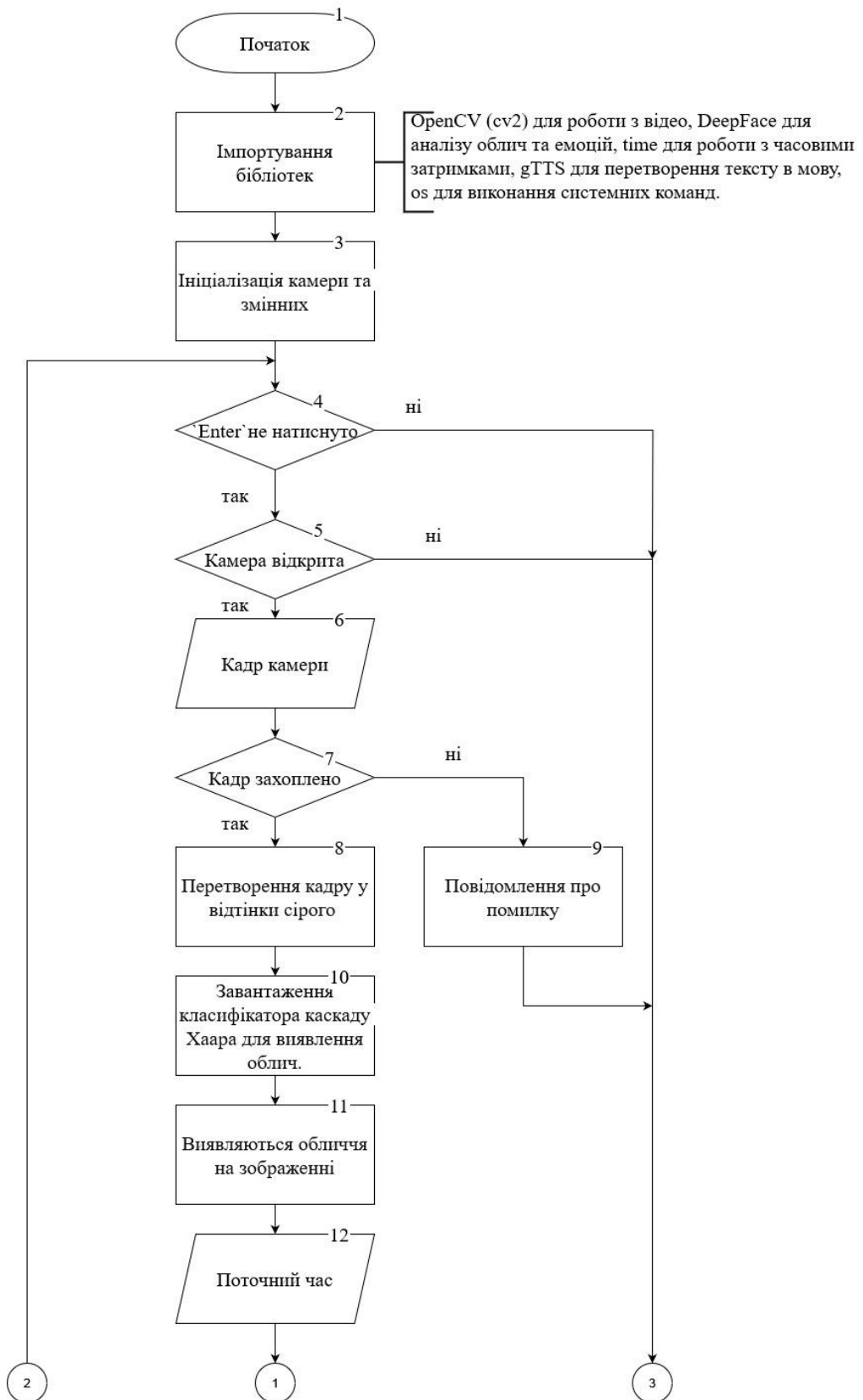
[42] Галаган Р.М., Андреев С.М., Петрик В.Ф., Баженов В.Г., Лисенко Ю.Ю. Виявлення дефектів бетонних конструкцій на основі аналізу зображень за допомогою згорткових нейронних мереж. Вчені записки ТНУ імені В.І. Вернадського. Серія: Технічні науки. 2023. Том 34 (73), № 2. с. 138-144.

[43] Куц, Ю. В. Новітні системи та технології. Частина І. Загальні питання побудови та опрацювання даних в комп'ютерно-інтегрованих системах НКТД [Електронний ресурс] / Ю. В. Куц, Ю. Ю. Лисенко, А. С. Момот ; КПІ ім. Ігоря Сікорського. – Київ : КПІ ім. Ігоря Сікорського, 2022. – 123 с

ДОДАТОК А: ЗАГАЛЬНИЙ КОД ПРОГРАМНОГО МОДУЛЯ

для отримання доступу необхідно звернутись до авторів проекту

ДОДАТОК Б: БЛОК-СХЕМА АЛГОРИТМУ КОДУ



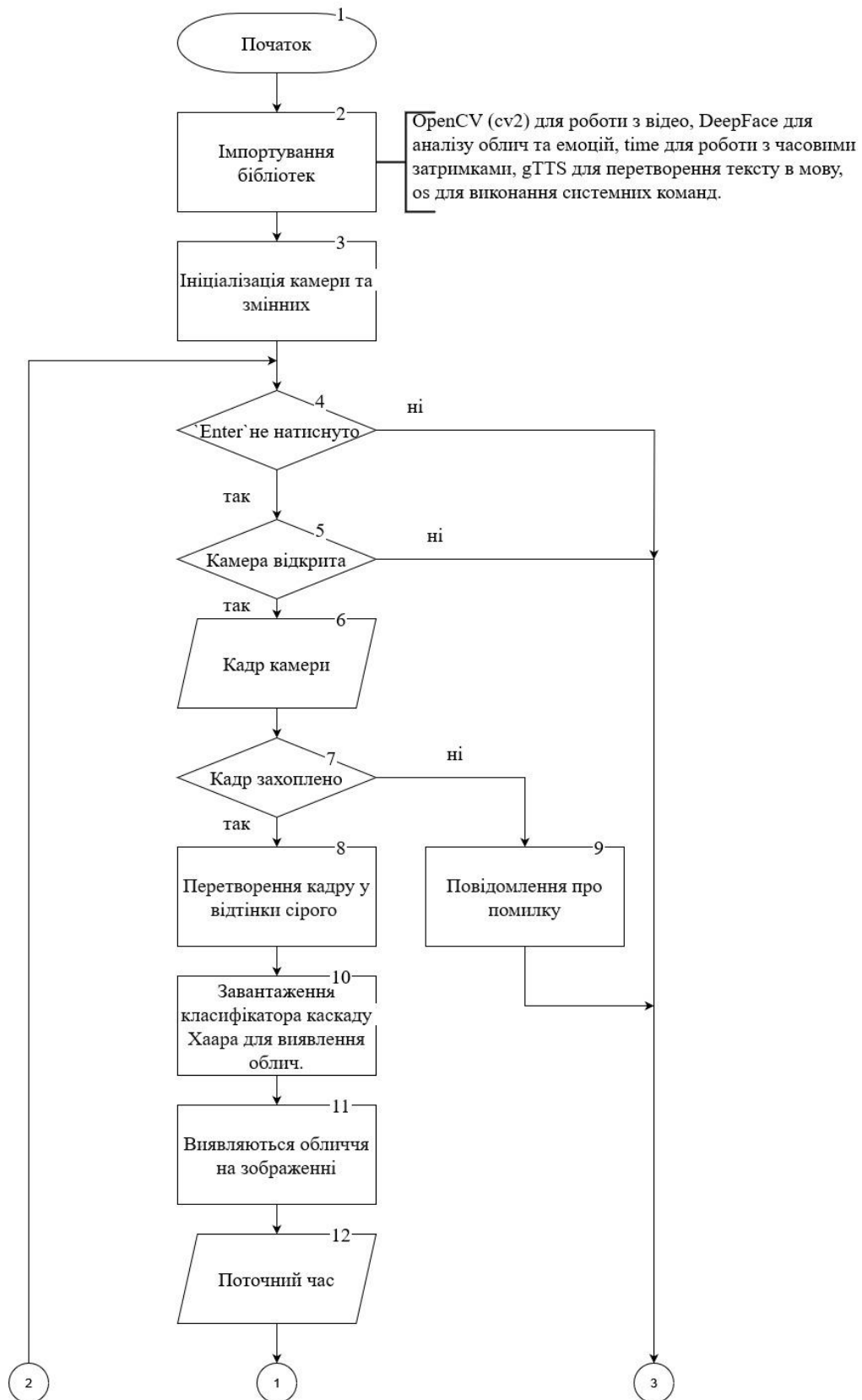


Рисунок Б.1

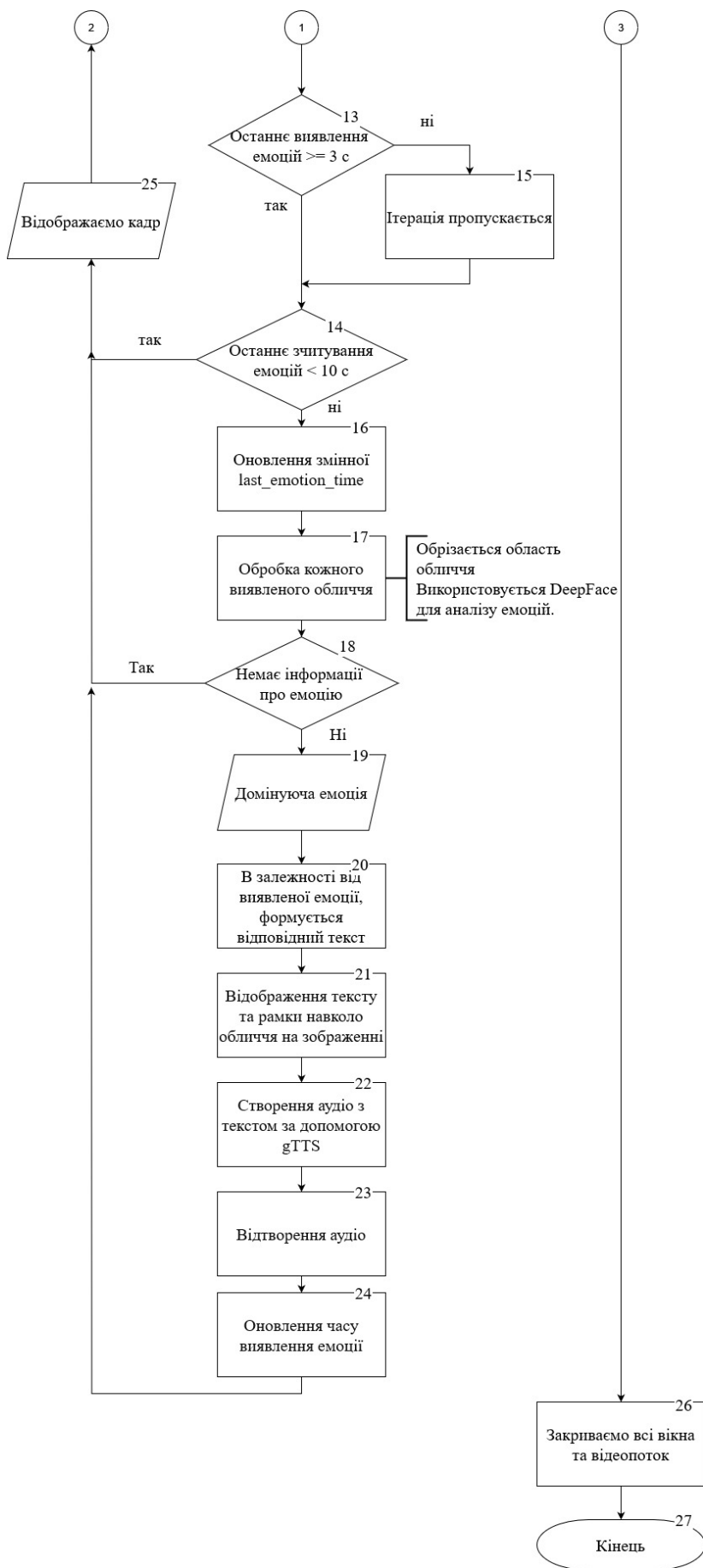


Рисунок Б.2