

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту  
(повна назва)

Кафедра Інформатики  
(повна назва)

**КВАЛІФІКАЦІЙНА РОБОТА**  
**Пояснювальна записка**

рівень вищої освіти другий (магістерський)

**ДОСЛІДЖЕННЯ МЕТОДУ РОЗПІЗНАВАННЯ**

**ЖЕСТОВОЇ МОВИ**

(тема)

Виконав:

студент 2 курсу, групи ІНФМ-20-1

Харченко А.В.

(прізвище, ініціали)

Спеціальності 122 Комп'ютерні науки

(код і повна назва спеціальності)

Тип програми освітньо-професійна

Освітня програма Інформатика

(повна назва освітньої програми)

Керівник доц. Творошенко І.С.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри

\_\_\_\_\_

(підпис)

Кобилін О.А.

(прізвище, ініціали)

2021 р.

## Харківський національний університет радіоелектроніки

Факультет Інформаційно-аналітичних технологій та менеджменту  
(повна назва)Кафедра Інформатики  
(повна назва)Рівень вищої освіти другий (магістерський)Спеціальність 122 Комп'ютерні науки  
(код і повна назва)Тип програми освітньо-професійнаОсвітня програма Інформатика  
(повна назва освітньої програми)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

«\_\_\_\_\_» \_\_\_\_\_ 2021 р.

**ЗАВДАННЯ**  
НА КВАЛІФІКАЦІЙНУ РОБОТУстудентові Харченко Анастасії Віталіївні  
(прізвище, ім'я, по батькові)1. Тема роботи Дослідження методу розпізнавання жестової мови

затверджена наказом по університету від « 22 » жовтня 2021 року № 1574Ст.

2. Термін подання студентом роботи до екзаменаційної комісії 22 листопада 2021 р.3. Вихідні дані до роботи науково-методична та науково-технічна література, матеріали конференцій, дані інтернет-мережі, бібліотека OpenCV, мова програмування Python, середовище розроблення Jupyter Notebook.

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

1. Проаналізувати сучасні системи розпізнавання жестової мови.2. Здійснити класифікацію і проаналізувати існуючі методи розпізнавання жестової мови.3. Виявити особливості азбуки та мови жестів.4. Виявити особливості розроблення методів розпізнавання жестової мови.5. Здійснити вибір інструментального засобу для програмної реалізації.6. Програмно реалізувати метод розпізнавання жестової мови.7. Протестувати розроблений застосунок та провести аналіз результатів.8. Виявити перспективи подальшої роботи.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Актуальність дослідження, об'єкт та мета дослідження, постановка задачі дослідження, вихідні дані дослідження, етапи розроблення, результат тестування, висновки, перспективи подальших досліджень, апробація результатів роботи.

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

| Найменування розділу                               | Консультант<br>(посада, прізвище, ім'я, по батькові) | Позначка консультанта про виконання розділу |      |
|--|--|---|------|
|  |  | підпис                                      | дата |
| Консультант з дотримання діючих стандартів та норм | Доцент Белова Н.В.                                   |   |      |

### КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів роботи                         | Терміни виконання етапів роботи | Примітка |
|-------|---|---------------------------------|----------|
| 1     | Отримання завдання на кваліфікаційну роботу | 22.10.2021                      |          |
| 2     | Аналіз завдання, підбір літератури          | 22.10.21-04.11.21               |          |
| 3     | Аналіз літератури з досліджуваної проблеми  | 05.11.21-15.11.21               |          |
| 4     | Аналіз методів розпізнавання жестової мови  | 17.11.21-26.11.21               |          |
| 5     | Розробка методу                             | 27.11.21-31.11.21               |          |
| 6     | Програмна реалізація                        | 01.11.21-07.11.21               |          |
| 7     | Оформлення пояснювальної записки            | 08.11.21-15.11.21               |          |
| 8     | Перевірка на плагіат                        | 20.11.2021                      |          |
| 9     | Рецензування                                | 28.11.2021                      |          |
| 10    | Підготовка презентації та доповіді          | 29.11.2021                      |          |
| 11    | Занесення роботи в електронний архів        | 01.12.2021                      |          |
| 12    | Попередній захист кваліфікаційної роботи    | 01.12.2021                      |          |
|       |   |                                 |          |
|       |   |                                 |          |

Дата видачі завдання 22 жовтня 2021 р.

Студент \_\_\_\_\_  
(підпис)

Керівник роботи \_\_\_\_\_ доц. Творошенко І.С.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 74 с., 40 рис., 73 джерел.

АЗБУКА МОВИ ЖЕСТІВ, ЖЕСТ, ЗРАЗОК ЖЕСТУ, МОВА ЖЕСТІВ, КОЛЕКЦІЯ ТЕСТІВ, КОНТУР ЗОБРАЖЕННЯ РУКИ, ЛЮДИ З ВАДАМИ СЛУХУ, РОЗПІЗНАВАННЯ КИСТІ.

Об'єктом дослідження є набір зображень жестів рук.

Метою дослідження є реалізація та вивчення методу розпізнавання жестової мови для безбар'єрного розуміння людей з вадами слуху або автоматичного фіксування спеціальних жестів.

У ході виконання кваліфікаційної роботи продемонстровано результати експерименту щодо 7 жестів, загалом у базі є 30 зразків жестів рук.

Дана розробка зумовлена не тільки, щоб зробити комфортним спілкування між глухонімими людьми та з глухонімими людьми, але й як інструмент автоматичного фіксування прохання про допомогу будь-якої людини, яка потрапила у кадр відеореєстратора. У ході тестування найвищий відсоток розпізнавання зафіксовано 95%, найменший – 43%.

Програма може використовуватись для безбар'єрного розуміння глухонімих людей або автоматичного фіксування жестів, які «повідомляють» про небезпечну життєву ситуацію чи загрозу життю людини.

ALPHABET OF GESTURAL LANGUAGE, GESTURE, SAMPLE GESTURE, GESTURE LANGUAGE, COLLECTION OF TESTS, CONTOURS OF IMAGE OF HANDS, PEOPLE WITH HEARING IMPAIRMENTS, RECOGNITION.

The object of research is a set of images of hand gestures.

The aim of the study is to implement and study the method of sign language recognition for barrier-free understanding of people with hearing impairments or automatic recording of special gestures.

During the qualification work the results of the experiment on 7 gestures were demonstrated, in total there are 30 samples of hand gestures in the database.

This development is designed not only to make comfortable communication between deaf people and with deaf people, but also as a tool to automatically record the request for help of anyone who was in the frame of the DVR. During testing, the highest percentage of recognition was recorded at 95%, the lowest 43%.

The program can be used for barrier-free understanding of deaf people or automatic recording of gestures that "report" a dangerous life situation or threat to human life.

## ЗМІСТ

|  |    |
|--|----|
| Перелік умовних позначень, символів, одиниць, скорочень і термінів .....                   | 7  |
| Вступ .....  | 8  |
| 1 Аналіз існуючих методів розпізнавання жестової мови .....                                | 10 |
| 1.1 Аналіз сучасних систем розпізнавання жестової мови .....                               | 10 |
| 1.2 Класифікація та аналіз існуючих методів розпізнавання жестової мови .....              | 11 |
| 1.2.1 Розпізнавання позиції та орієнтації за допомогою моментів зображення.....            | 12 |
| 1.2.2 Розпізнавання рухів на основі аналізу різниць зображень .....                        | 14 |
| 1.2.3 Розпізнавання конфігурації на основі аналізу гістограм.....                          | 15 |
| 1.2.4 Розпізнавання конфігурації і позиції із застосуванням кольорових рукавичок.....      | 17 |
| 1.2.5 Розпізнавання конфігурації і позиції на основі аналізу контуру зображення руки ..... | 18 |
| 1.3 Особливості азбуки та мови жестів.....   | 19 |
| 1.4 Аналіз літературних джерел щодо апробації результатів розпізнавання жестової мови..... | 21 |
| 1.5 Постановка задачі дослідження .....  | 28 |
| 2 Особливості розроблення методів розпізнавання жестової мови .....                        | 29 |
| 2.1 Механізм оброблення зображення руки.....   | 29 |
| 2.2 Принципи розпізнавання складових кисті .....   | 31 |
| 2.3 Методика побудови тривимірного подання верхньої кінцівки .....                         | 43 |
| 2.4 Методика розпізнавання азбуки жестів руки.....   | 46 |
| 3 Реалізація та дослідження методу розпізнавання жестової мови .....                       | 50 |
| 3.1 Вибір інструментального засобу для реалізації методу розпізнавання жестової мови.....  | 50 |
| 3.1.1 Огляд мови Python та бібліотеки OpenCV.....  | 50 |

|   |    |
|---|----|
|   | 6  |
| 3.1.2 Обґрунтування вибору середовища програмної реалізації.....        | 52 |
| 3.2 Етапи програмної реалізації методу розпізнавання жестової мови..... | 53 |
| 3.3 Тестування розробленого застосунку та аналіз результатів .....      | 57 |
| 3.4 Перспективи подальшої роботи.....                                   | 64 |
| Висновки.....   | 66 |
| Перелік джерел посилання .....  | 67 |

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

США – Сполучені Штати Америки

ASL – American Sign Language (американська мова жестів)

DPCNN – Deep Pyramid Convolutional Neural Networks (глибока піраміда згорткової нейронної мережі)

HCI – Human-Computer Interaction (взаємодія людини з комп'ютером)

MEI – Motion Energy Image (зображення енергії руху)

MHI – Motion History Image (зображення історії руху)

NLP – Natural Language Processing (опрацювання природної мови)

OM – Orientation Maps (орієнтаційні карти)

OH – Orientation Histograms (орієнтаційні гістограми)

PCNN – Pulse-Coupled Networks (мережі з імпульсним зв'язком)

## ВСТУП

Безумовно сьогодні створення природних, легких в управлінні людино-машинних інтерфейсів для різних застосунків є актуальною задачею.

У даний час проводиться досить багато досліджень по створенню методів розпізнавання образів [1-5], що дозволяють безконтактно взаємодіяти з комп'ютером за допомогою жестів рук, в тому числі для розпізнавання жестової мови.

Постає проблема щодо забезпечення комунікації людей з вадами слуху з іншими представниками суспільства, що потребує розроблення систем моделювання та розпізнавання жестової мови. Зокрема, приділяється увага розпізнаванню елементів дактильної жестової мови, яка, в свою чергу, зображує букви алфавіту. Адже люди з вадами слуху використовують дактильну мову для відображення власних назв, імен, іншомовних наукових термінів, речовин і т.д.

Розробці систем для розпізнавання елементів жестової мови присвячено значну кількість робіт, зокрема, у працях [6, 7] описано системи розпізнавання жестової мови при використанні спеціального апаратного забезпечення. Ряд дослідників запропонували систему, яка ідентифікує жести конкретної особи [8]. Крім того, більшість підходів спрямовані на розпізнавання окремих ознак жестів або розпізнавання жестів, які характеризуються лише напрямком руху руки [9, 10].

Разом з тим, у процесі відтворення елементів жестової мови виникає ситуація, коли в жесті присутнє перекривання між пальцями або фронтальне направлення пальців до співрозмовника тощо. У силу цього при розпізнаванні жесту виникають труднощі для ідентифікації фронтально направлених пальців або областей перекривання пальців.

У роботі [11] пропонується використовувати стереозір для вирішення даної проблеми. Стереозір включає в себе два процеси: співставлення деталей,

які спостерігаються двома чи більше камерами, і відтворення їх тривимірного прообразу [12]. Співставлення деталей, що фіксуються очима людини, та використання їх відмінностей надає можливість покращити уявлення про глибину об'єкта. У свою чергу, відтворення тривимірного прообразу точок можна знайти як точку перетину променів, що проходять через зазначені точки та центри відповідних діафрагм камер.

Основним завданням розпізнавання жестової мови є одержання перекладу жестів у вигляді тексту для комунікації звичайної людини та людини з вадами слуху. Області застосування – системи безконтактної людино-машинної взаємодії.

У даній роботі розглядається задача розпізнавання жестів долоні на зображеннях [13-18], отриманих з однієї або двох вебкамер. Для вирішення поставленого завдання пропонується метод, який працює на основі аналізу форми долоні. Використання тільки форми, без текстурних ознак, дозволяє застосовувати методику навіть на зображеннях низької якості, що отримані, як кадри, з вебкамери.

Під час дослідження запропоновано метод виділення саме кінчиків пальців, вводиться поняття центру долоні та спосіб його обчислення. Зазначені дані використовуються як ознаки для класифікації жестів.

Актуальність дослідження полягає у створенні системи розпізнавання жестової мови, що надасть можливість безбар'єрного розуміння глухонімих людей або автоматичного фіксування жестів, які «повідомляють» про небезпечну життєву ситуацію чи загрозу життю людини.

# 1 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ РОЗПІЗНАВАННЯ ЖЕСТОВОЇ МОВИ

## 1.1 Аналіз сучасних систем розпізнавання жестової мови

Жести – це різні рухи тіла, які є невербальним способом передачі інформації. Рух пальців, рук, голови, плечей, міміка обличчя: все перераховане є жестами. За допомогою жестів людина може передати самостійну інформаційну одиницю, доповнити словесний ряд, передати почуття і т.д. Жести поділяються на статичні (сприймаються одномоментно) і динамічні (що сприймаються в деякий період часу) жести, які мають певну інтерпретацію в ручних абетках і безконтактних людино-машинних інтерфейсах.

Система розпізнавання жестів руки – це сукупність комп'ютерних технологій і математичних алгоритмів, яка дозволяє вирішувати задачу розпізнавання певної групи жестів рук.

Розпізнавання жестів можна застосовувати, наприклад, у таких областях діяльності людини:

– управління комп'ютером і побутовими приладами:

1) у комп'ютерному застосунку кожної конфігурації руки зіставляється певна команда. Людина показує жест, система розпізнає конфігурацію руки і відправляє відповідну команду в комп'ютер;

2) позиція долоні зіставляється з позицією курсора миші на екрані. Рухи руки призводять до рухів курсору. Команди натискання кнопок мишки зіставляються з різними конфігураціями руки;

3) розпізнані позиції кінчиків пальців руки можуть бути використані для управління комп'ютером або побутовими приладами одночасно кількома пальцями руки;

– створення природних людино-машинних інтерфейсів для глухонімих:

1) систему розпізнавання жестів можна застосувати для введення тексту в комп'ютер за допомогою жестів руки, що для глухонімих людей простіше і природніше, ніж введення тексту за допомогою клавіатури комп'ютера;

2) систему розпізнавання жестів можна застосувати для створення програмних засобів онлайн-зв'язку, коли пропускна здатність мережі мала і відеозв'язок неможливий. Маючи розпізнані жести руки на одному кінці мережі, можна показати анімацію цих же жестів, на іншому кінці, передаючи по мережі тільки характеристики показаних жестів;

– маніпуляція тривимірними моделями об'єктів. На сьогоднішній день для роботи з тривимірними моделями, зазвичай, використовують комп'ютерну мишу, яка не дуже зручна для цього завдання. Маючи тривимірні координати руки і кінчиків пальців руки, можна створити систему HCI (Human-Computer Interaction), яка дозволить керувати моделями в усіх напрямках тривимірного простору;

– застосунки віртуальної реальності. Доповнивши систему розпізнавання жестів пристроями, наприклад, стереоскопічними окулярами, можна створити інструментальні засоби віртуальної реальності, де користувач зможе «доторкатися» до віртуальних об'єктів.

Наведений список можна доповнити системами, що використовуються для реабілітації пацієнтів [11], розпізнавання жестової мови і т.д. Декілька з наведених програмних засобів вже використовуються в різних пристроях і програмах, а деякі з них ще знаходяться на етапі досліджень.

## 1.2 Класифікація та аналіз існуючих методів розпізнавання жестової мови

Особливістю методів, заснованих на аналізі зовнішніх ознак жесту, є інтепретація тільки зовнішнього вигляду (форми, позиції і т.д.) цільового

об'єкта. Під час розпізнавання необхідного предмету інформація про його фізичні властивості не зберігається.

Розглянемо відомі роботи і методи, присвячені розпізнаванню жестів руки людини на основі аналізу зовнішніх ознак.

### 1.2.1 Розпізнавання позиції та орієнтації за допомогою моментів зображення

Позначимо через  $M_{0,0}$ ,  $M_{1,0}$ ,  $M_{0,1}$ ,  $M_{2,0}$ ,  $M_{1,1}$ ,  $M_{0,2}$  моменти нульового, першого і другого порядку зображення з функцією інтенсивності  $f(x,y)$ :

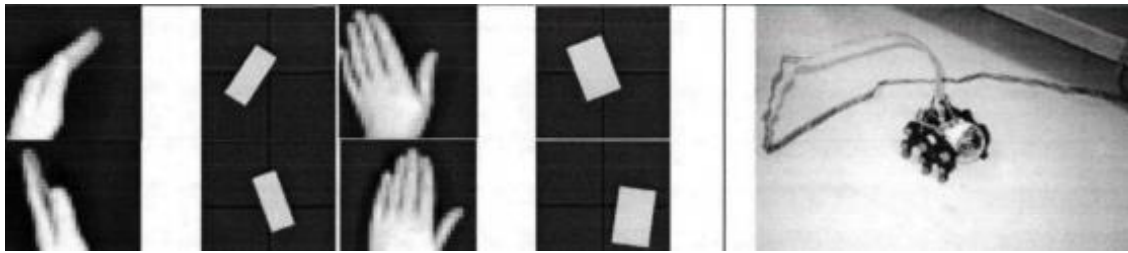
$$M_{0,0} = \sum_x \sum_y f(x, y), \quad (1.1)$$

$$M_{0,1} = \sum_x \sum_y y \times f(x, y), M_{1,0} = \sum_x \sum_y x \times f(x, y), \quad (1.2)$$

$$\begin{aligned} M_{1,1} &= \sum_x \sum_y x \times y \times f(x, y), M_{0,2} = \sum_x \sum_y y^2 \times f(x, y), M_{2,0} = \\ &= \sum_x \sum_y x^2 \times f(x, y). \end{aligned} \quad (1.3)$$

При деяких обмеженнях моменти зображення можуть бути використані для розпізнавання простих жестів рук і створення на їх основі інструментальних засобів НСІ.

Наприклад, в роботі [13] розглядається застосунок, що дозволяє управляти іграшковим роботом за допомогою рухів руки, де орієнтація руки визначає напрямки руху робота (рис. 1.1).



(a)

(б)

Рисунок 1.1 – Управління іграшковим роботом простими жестами руки:

(а) – прямокутник показує розпізнаний напрямок, позицію і розмір зображення руки; (б) – іграшковий робот

Розпізнавання позиції  $(x_c, y_c)$  і орієнтації руки  $\Theta$  в бінарному зображенні здійснюється за допомогою обчислення моментів зображення, за умови, що фон зображення однорідний і рука є домінуючим об'єктом в зображенні:

$$x_c = \frac{M_{1,0}}{M_{0,0}}, \quad (1.4)$$

$$y_c = \frac{M_{0,1}}{M_{0,0}}, \quad (1.5)$$

$$\Theta = \frac{\arctan^*(b, (a-c))}{2}, \quad (1.6)$$

де

$$a = \frac{M_{2,0}}{M_{0,0}} - x_c^2; \quad (1.7)$$

$$c = \frac{M_{0,2}}{M_{0,0}} - y_c^2; \quad (1.8)$$

$$b = 2 \times \left( \frac{M_{1,1}}{M_{0,0}} - x_c \times y_c \right). \quad (1.9)$$

Функція  $\arctan^*$  задовольняє умові  $\arctan(i, j) \in (-\pi, \pi]$  і обчислюється за формулою:

$$\arctan^*(i, j) = \begin{cases} \arctan\left(\frac{i}{j}\right) & j > 0, \\ \arctan\left(\frac{i}{j}\right) + \pi & i \geq 0, j < 0, \\ \arctan\left(\frac{i}{j}\right) - \pi & i < 0, j < 0, \\ +\frac{\pi}{2} & i > 0, j = 0, \\ -\frac{\pi}{2} & i < 0, j = 0, \\ \text{невизначено} & i = 0, j = 0. \end{cases} \quad (1.10)$$

Розмір зображення руки оцінюється значеннями  $l_1$  та  $l_2$  :

$$l_1 = \sqrt{\frac{(a+c) + \sqrt{b^2 + (a-c)^2}}{2}}, \quad (1.11)$$

$$l_2 = \sqrt{\frac{(a+c) - \sqrt{b^2 + (a-c)^2}}{2}}. \quad (1.12)$$

На рисунку 1.1 (а) для кожного зображення руки наведені відповідні прямокутники з центром  $(x_c, y_c)$  та з розміром  $l_1 \times l_2$ .

### 1.2.2 Розпізнавання рухів на основі аналізу різниць зображень

На практиці забезпечити умови однорідного фону вдається не завжди. Для таких випадків, зазвичай, використовується метод, заснований на аналізі центру маси різниць зображень (Motion Energy Image – MEI) руки в кадрах відеоряду (рис. 1.2) [14].

Використання різниць кадрів відеоряду (MEI) дозволяє в реальному часі аналізувати руху об'єкта в відеоряді при стабільному, але необов'язково однорідному фоні зображення.

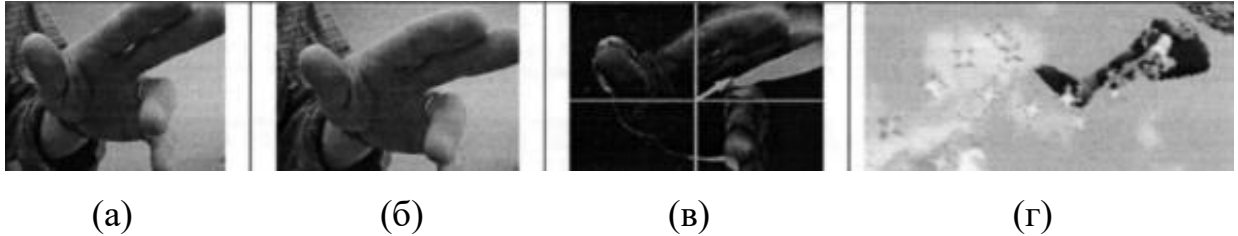


Рисунок 1.2 – Управління аватаром в комп'ютерній грі:

- (а) і (б) – два кадри з відеоряду; (в) – різниця зображень (а-б);  
 (г) – центр маси (стріла (в)) управляє напрямком польоту аватара в грі

На практиці дана технологія і її вдосконалені види (наприклад, Motion History Image – МНІ) знайшли застосування в таких застосунках, як інтерактивний віртуальний тренер з аеробіки [15-16] і інтерактивна кімната для розповіді історій [17].

### 1.2.3 Розпізнавання конфігурації на основі аналізу гістограм

У більшості програмних засобів комп'ютерного зору, крім позиції і орієнтації руки людини, потрібна додаткова інформація про її конфігурації. Дане завдання в роботах [17-19] вирішується аналізом, так званих гістограм напрямків (Orientation Histograms) і карт напрямків (Orientation Maps) зображення, які менш чутливі до змін освітлення середовища, яке спостерігається.

Припустимо  $f(x,y)$  є інтенсивністю бінарного або напівтонового зображення руки. Локальний напрямок  $\theta(x,y)$  у точці  $(x,y)$  визначається таким чином [20]:

$$\theta(x, y) = \arctan^*(f(x, y) - f(x - 1, y), f(x, y) - f(x, y - 1)). \quad (1.13)$$

Для заданого цілого значення  $N$ , що визначає кількість секторів напрямків, будується вектор  $\Phi$ , який показує кількість локальних напрямків  $\theta(x, y)$ , які знаходяться між кутами  $\frac{360^\circ}{N}(i - \frac{1}{2})$  та  $\frac{360^\circ}{N}(i + \frac{1}{2})$ :

$$\Phi(i) = \sum_{x,y} \begin{cases} 1, \text{ якщо } \left| \theta(x, y - \frac{360^\circ}{N}i) \right| < \frac{360^\circ}{2N}, i = 1, 2, \dots, N, \\ 0, \text{ в іншому випадку.} \end{cases} \quad (1.14)$$

Відстань між двома зображеннями руки визначається як евклідова відстань між векторами  $\Phi$  цих зображень. На рисунку 1.3 наведені зображення жестів руки і відповідних гістограм напрямків при  $N = 36$ .

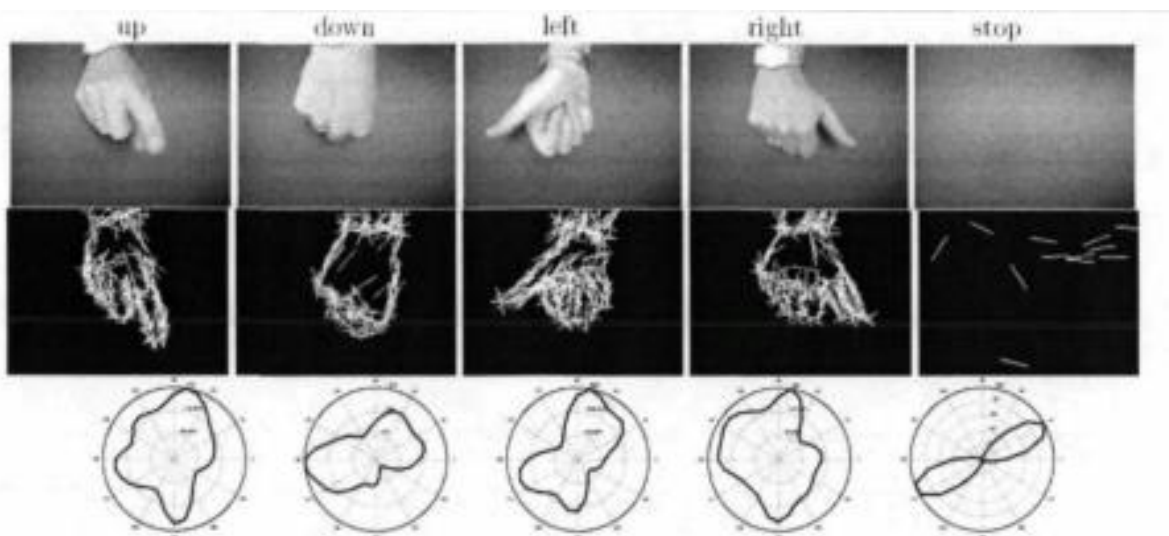


Рисунок 1.3 – Жести руки, карти та гістограми напрямків

Наведені жести і технологія розпізнавання конфігурацій руки на основі гістограм напрямків використано в програмному засобі управління анімаційним краном [20]. У роботі зазначається, що застосунок працює в реальному часі, він не відчуває незначні зміни розміру руки, але фіксує зміну напрямку руки. Програмне забезпечення вимагає навчання для кожного оператора.

Дана технологія дозволяє в режимі реального часу розпізнавати конфігурацію руки, якщо виконуються такі умови:

- рука є домінуючим об'єктом на зображенні;
- фон зображення однорідний;
- жести обрані таким чином, що їх гістограми напрямків мають значимі відмінності один від одного.

#### 1.2.4 Розпізнавання конфігурації і позиції із застосуванням кольорових рукавичок

Для розпізнавання жестів руки часто застосовують кольорові рукавички [17-21]. Запропонований в роботі [17] метод дозволяє за допомогою однієї відеокамери в реальному часі розпізнавати конфігурацію руки і відстежувати рухи долоні в тривимірному просторі (рис. 1.4). Використана рукавичка сконструйована з двадцяти сегментів десяти різних кольорів. Використання невеликої кількості кольорів дозволяє розпізнати колір обраної точки зображення рукавички при різному освітленні, а спеціальне розміщення кольорових сегментів не дозволяє отримати ідентичні зображення при різних конфігураціях руки.

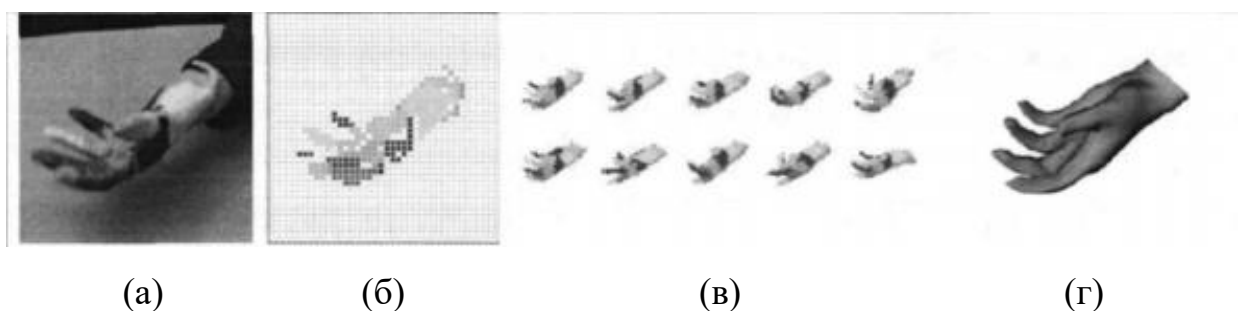


Рисунок 1.4 – Результат застосування кольорових рукавичок у завданні розпізнавання жестів руки:

(а) – вхідне зображення руки; (б) – растрове зображення руки розміром 40×40; (в) – схожі зображення руки в базі даних; (г) – знайдена конфігурація

Розпізнавання конфігурації руки здійснюється згладжуванням початкового зображення алгоритмом двостороннього розмиття (Bilateral Filter), видаленням фонових точок, де останні класифікуються на основі моделі гаусової суміші (Gaussian Mixture Models) і перетворенням отриманого зображення в растрове меншого розміру (рис. 1.4 (б)). Отримані результати порівнюється еталонними зображеннями (18000 екземплярів), що зберігаються в базі даних та відповідають різним формам руки (рис. 1.4 (в)).

Відстань  $d(r^i, r^j)$  між растровими зображеннями  $r^i$  та  $r^j$  оцінюється так:

$$d(r^i, r^j) = \tilde{d}(r^i, r^j) + \tilde{d}(r^j, r^i), \quad (1.15)$$

де

$$\tilde{d}(r^i, r^j) = \sqrt{\frac{1}{|C_i|} \sum_{(x,y) \in C_i} \min_{(u,v) \in S_{xy}} (u-x)^2 + (v-y)^2}; \quad (1.16)$$

$$S_{xy} = \{(u, v) | r^i(x, y) = r^j(u, v)\}; \quad (1.17)$$

$$C_i = \{(x, y) | r^i(x, y) \neq \text{фон}\}. \quad (1.18)$$

Описана вище система знайшла своє застосування в задачах управління анімаційним персонажем.

### 1.2.5 Розпізнавання конфігурації і позиції на основі аналізу контуру зображення руки

Як дескриптор конфігурації руки часто вибирають контур її зображення. Наприклад, в роботі [22] пропонується система розпізнавання жестів ручної азбуки глухонімих ASL на основі аналізу контуру зображення руки. Для

полегшення завдання два динамічних жести не розглядаються, а частина жестів замінюються новими. У кольоровому зображенні руки видаляються всі точки, які не відповідають кольору шкіри людини.

Отримане зображення перетворюється в бінарне зображення, після чого воно згладжується фільтром Гауса. Рука виділяється як пов'язаний компонент. Контури руки виділяються за допомогою оператора Собеля. Обчислюється описаний прямокутник контуру руки, центр якого відповідає початку координат.

Дескриптором жесту є вектор  $(x_c, y_c)$ , а площа контуру руки

$$(x_c, y_c) = \frac{1}{n} (\sum_{i=1}^n x_i, \sum_{i=1}^n y_i), \quad (1.19)$$

де  $(x_i, y_i)$  є координатами точок контуру,  $i=1, 2, \dots, n$ .

Схожість жестів оцінюється відстанню Евкліда між векторами дескрипторів жестів. У даній роботі наводиться оцінка якості розпізнавання – 90%. Зазначено, що якість розпізнавання при показі жестів різними людьми не вивчалася.

У роботі [21] розглядається задача розпізнавання позиції кінчиків пальців руки на основі аналізу контуру руки в кольоровому зображенні. Першими кроками алгоритму, як і в випадку [20], є видалення фонових точок, згладжування і виділення зображення руки. Маючи контур руки, пальці руки виділяються за допомогою аналізу локальних вигинів контуру. Даний метод використовувався для управління вебкамерою персонального комп'ютера.

### 1.3 Особливості азбуки та мови жестів

У наші дні для того, щоб вивчити якусь мову, достатньо мати доступ до мережі з будь-якого пристрою. Кожен день в інтернеті з'являється велика

кількість сайтів, за допомогою яких можливо знайти багато методів та уроків вивчення іноземних мов, у тому числі за допомогою відеоуроків.

Створюються мобільні застосунки, які спрямовані на вивчення мови в захоплюючій формі. Але існує одна унікальна мова, що відрізняється від інших саме невербальним способом передачі інформації – це мова жестів. Слова і пропозиції в мові жестів показуються за допомогою жестів рук, пальців та міміки обличчя. Один жест мови жестів може означати літеру, слово і навіть висловлювати почуття.

Ручна азбука або дактильний алфавіт – це азбука, яка використовується в дактилології, що відтворює за допомогою пальців рук орфографічну форму слова мови. Спілкування за допомогою дактиля вважається вербальним. Для позначення тієї чи іншої букви змінюється положення пальців руки.

Більшість букв дактильного алфавіту схожа на друковані, це полегшує його вивчення. Володіючи ручною азбукою, відбувається спілкування між глухою людиною та людиною, яка чує. Однак, можуть виникнути труднощі під час такого спілкування, коли людина не володіє мовою жестів. Для вирішення даних проблем проводяться дослідження зі створення систем автоматичного сурдоперекладу і систем, оснащених більш природним людино-машинним інтерфейсом для глухих людей.

Враховуючи те, що ручна азбука має лише частину жестів, що використовуються в мові глухонімих, автоматичне розпізнавання ручної азбуки відкриває шлях для створення більш природних людино-машинних інтерфейсів, прибирає обмеження спілкування, з якими стикаються глухі люди в повсякденному житті.

На відміну від людей, які стали глухими в результаті нещасного випадку або по причині захворювання, люди, які не чуять з народження, віддають перевагу мові жестів замість звичайного тексту. Їм легше прийняти і показати жести, ніж читати або набирати текст на клавіатурі комп'ютера або телефону.

#### 1.4 Аналіз літературних джерел щодо апробації результатів розпізнавання жестової мови

Згідно з даними Всесвітньої організації охорони здоров'я [23], у світі 360 млн. людей з вадами слуху і глухих людей. Враховуючи швидкий розвиток комунікаційних технологій, проблема комунікації людей з вадами слуху продовжує бути актуальною. З кожним днем з'являється все більше досліджень, що спрямовані на вирішення даної проблеми. Розглянемо існуючі на даний момент розробки в області синхронного перекладу з мови жестів.

Один з напрямків по перекладу з мови жестів пов'язаний з розвитком робототехніки.

Машинний переклад з мови жестів став можливим з 1977 року, коли була створена роботизована рука.

Адже використання кібер-рукавичок – це один із способів, який дозволяє комп'ютеру відстежувати рухи людини і перекладати їх на розмовну або письмову мову.

У 2002 році Райаном Паттерсоном [24], у США, була розроблена спрощена рукавичка. Пристрій дозволяв відстежувати жести азбуки глухих, а потім здійснювати бездротову передачу даних на портативний пристрій, який відображав текст на екрані. Рукавичку необхідно було навчати відчувати руку конкретної людини.

Пізніше, використання кібер-рукавичок з датчиками руху стало основним, з'явилися такі проєкти, як CyberGlove [25] (рис. 1.5) та VPL Data Glove [26] (рис. 1.6).

Більш корисними є рукавички, які можуть захоплювати цілі знаки. Наприклад, CyberGlove рукавичка, що має 18-22 датчиків і підключається до головного комп'ютера через послідовний кабель (CyberGlove, n.d.) [27].

У VPL Data Glove є датчики, які є оптоволоконними перетворювачами і вимірюють кути вигину пальця.



Рисунок 1.5 – Приклад застосування рукавички CyberGlove



Рисунок 1.6 – VPL Data Glove рукавичка

Практичним прикладом використання даних технологій є проєкт SignAloud виконаний групою студентів з університету Вашингтону. За допомогою двох рукавичок (рис. 1.7), оснащених датчиками, що відстежують рух і передають дані у комп'ютерну систему через Bluetooth, розробники змогли здійснити переклад на англійську мову американську мову жестів (ASL) [28].



Рисунок 1.7 – Приклад однієї з рукавичок проєкта SignAloud

Незважаючи на високу точність, дані технології накладають певні обмеження для користувачів: необхідність використання спеціальних технічних пристроїв, відсутність можливості враховувати артикуляцію.

Kinect Sign Language Translator – перекладач, розроблений в 2012 році китайською командою у співпраці з командою Microsoft Research Asian [29], приклад застосування наведено на рисунку 1.8.

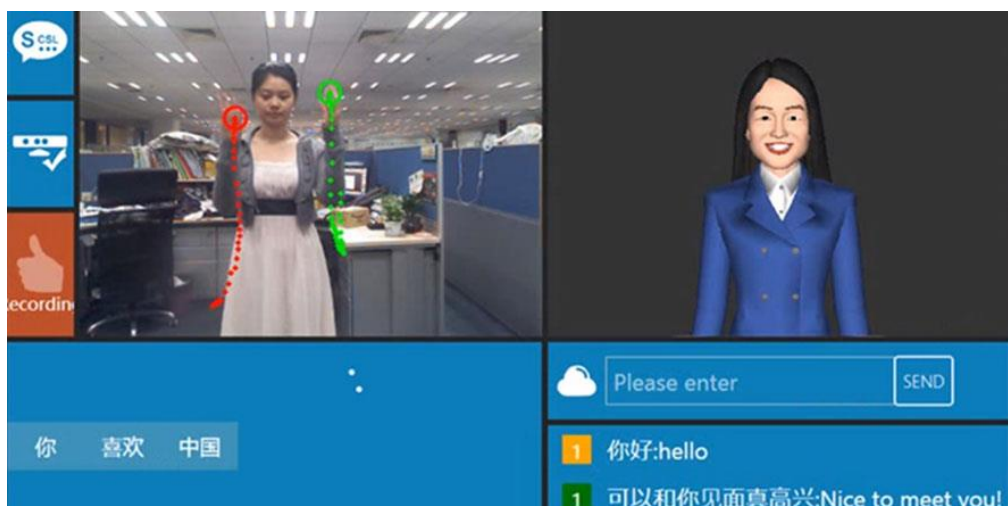


Рисунок 1.8 – Kinect Sign Language Translator у режимі спілкування

Перекладач складається з двох режимів: режиму перекладу і режим спілкування. Режим перекладу здатний перекладати окремі жести в письмовий текст (слова) і навпаки. Режим перекладача також може визначати пози і форми рук, а також траєкторію руху, використовуючи технології машинного навчання, розпізнавання образів і комп'ютерного зору. Режим спілкування може перекладати повні пропозиції, і розмова може бути автоматично перекладеною з використанням 3D-аватара. Для зчитування жестів використовується периферійний пристрій Kinect до ігрової приставки Xbox360 від Microsoft.

SignAll – це автоматична система угорської компанії Dolphio Technologies для перекладу американської мови жестів (ASL). Роботи ведуться з 2015 року. Дане дослідження засноване на комп'ютерному зорі і обробці природної мови (NLP) [30], використовує Kinect від Microsoft і вебкамери з датчиками глибини (датчик глибини і 3 вебкамери), що підключені до комп'ютера (рис. 1.9).

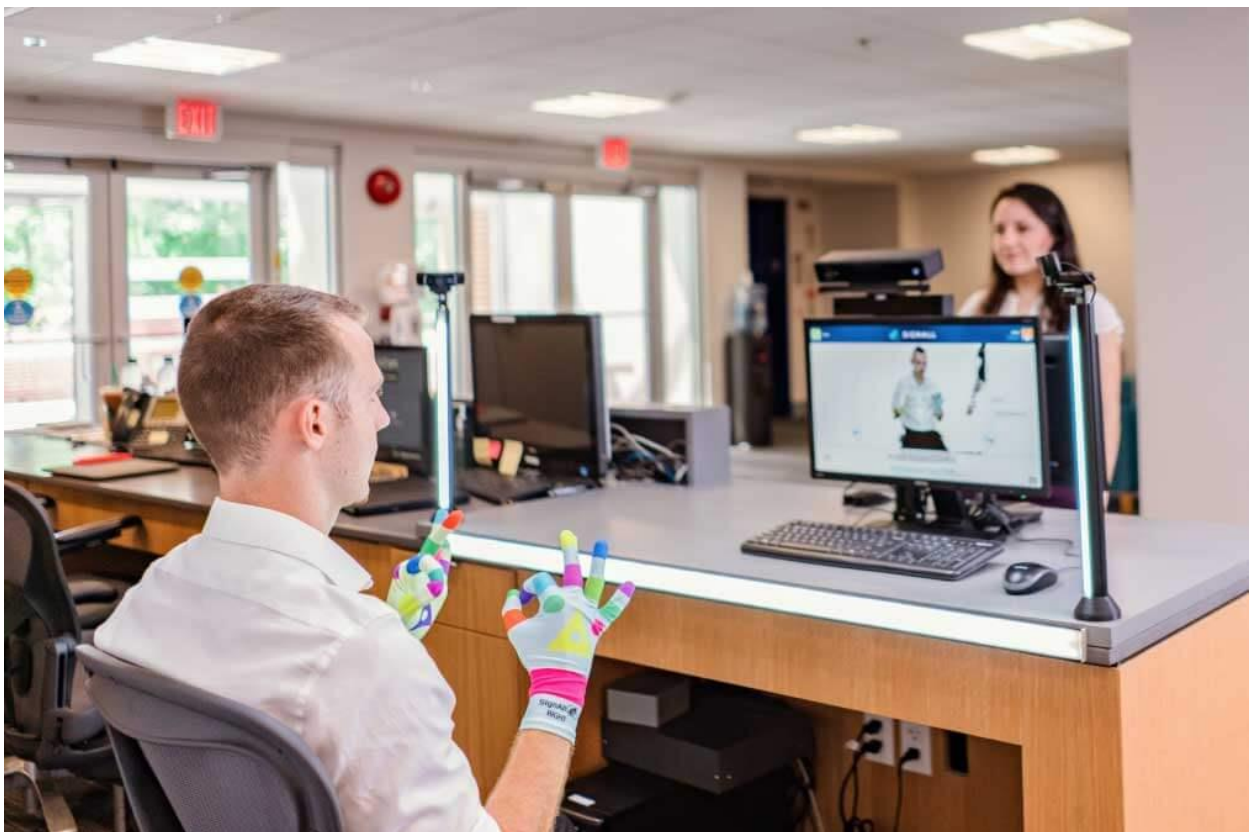


Рисунок 1.9 – Використання SignAll рукавичок

Технологія комп'ютерного зору зчитує жести і міміку глухої людини, а система обробки природної мови перетворює зібрані дані в просту англійську фразу. Недоліком є те, що для зчитування мови жестів повинна бути встановлена складна конструкція з трьох камер, сенсора і комп'ютера.

MotionSavvy – пристрій для озвучування американської мови жестів і навпаки [31]. Розроблений у 2012 році групою з Рочестерського технологічного інституту (Національного технічного інституту для глухих). Команда використовувала планшет з датчиком руху Leap Motion, приклад наведено на рисунку 1.10.



Рисунок 1.10 – Застосування MotionSavvy пристрою

Контролер Leap Motion 3D розпізнає рухи руками, які виконуються користувачем над поверхнею планшета. У результаті на екрані відображається зображення жесту і здійснюється його переклад в текст або озвучується.

Система Sign2 Conversion System [32] була розроблена для перетворення американської мови жестів (ASL) на англійська мову (письмову та розмовну).

Поточна система включає тільки жестикуляцію пальців у контрольованому середовищі з використанням комп'ютерного зору, але

довгостроковою метою є переклад повних пропозицій в природних середовищах, включаючи інтеграцію з персональними цифровими помічниками (PDA) і смартфонами (рис. 1.11).

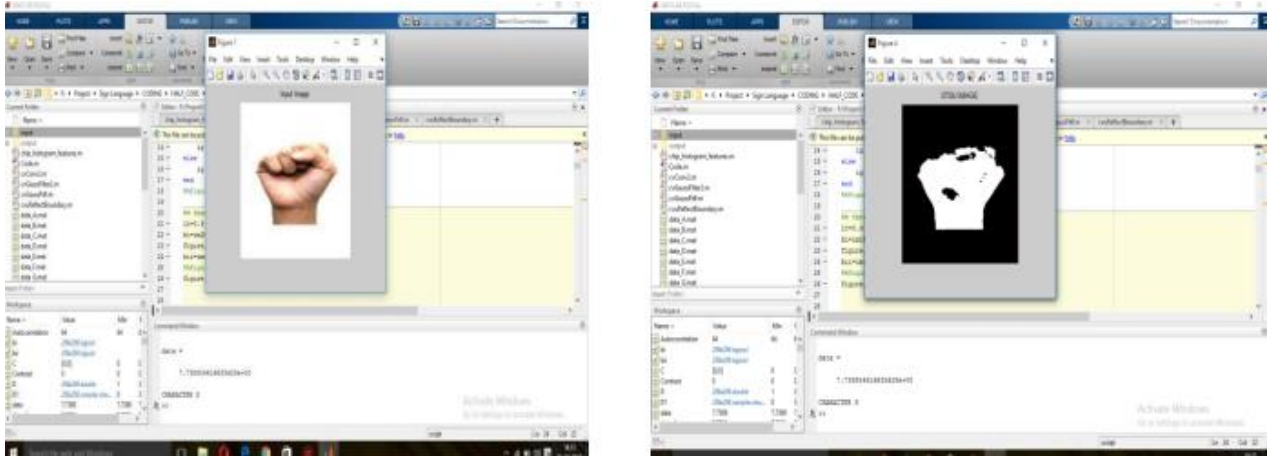


Рисунок 1.11 – Вхідні та вихідні дані програмного забезпечення Sign2 Conversion System

Слід зазначити, що не тільки великі компанії докладають зусиль, щоб зробити кращим та більш комфортним спілкування людей, які мають вади слуху, із зовнішнім середовищем. Глухий програміст Олексій Приходько представив прототип системи (рис. 1.12), яка здатна переводити мову жестів у текст його рідної мови.



Рисунок 1.12 – Прототип системи Олексія Приходька

Олексій Приходько – єдиний глухий програміст в світі, який працює над створенням автоматичного перекладача жестів на звукову мову. За його словами [33], сьогодні існує багато компаній, які стверджують, що змогли створити готовий перекладач для глухих, але «досконалого» серед них немає.

Програма, над якою працює Приходько, функціонує за аналогією з мозком людини. За допомогою камери вона розпізнає картинку і фіксує жести, які потім перетворює в зображення. Далі вони переводяться в моделі і обробляються системою, яка зіставляє їх з даними у нейронній мережі. Після повного процесу оброблення комп'ютер виводить на монітор вже відповідний жестах текстовий переклад [33].

Метод випадкових лісів (Randomized Decision Forests) є алгоритмом машинного навчання, що використовують в задачах класифікації, регресії та кластеризації. У задачах розпізнавання жестів даний метод використовують для розпізнавання позицій ключових характеристик тіла людини [34] та класифікації конфігурації руки (рис. 1.13).

Саме на цій технології заснована система розпізнавання жестів людини, випущена компанією Microsoft в 2010 році.



Рисунок 1.13 – Розпізнавання та класифікації конфігурації руки

Запропонований у [34] метод дозволяє розпізнавати позиції 20 частин тіла, у тому числі позиції долонь обох рук, отримуючи на вході одне далекоглядне зображення людини. Позиціонування бажаних точок тіла здійснюється за допомогою випадкових лісів рішень, які навчаються на вибірці зображень з 100 000 різних поз тіла людини.

### 1.5 Постановка задачі дослідження

Таким чином, є актуальним створення системи розпізнавання жестової мови, що надасть можливість безбар'єрного розуміння глухонімих людей або автоматичного фіксування жестів, які «повідомляють» про небезпечну життєву ситуацію чи загрозу життю людини.

Об'єктом дослідження є набір зображень жестів рук.

Метою дослідження є реалізація та вивчення методу розпізнавання жестової мови для безбар'єрного розуміння людей з вадами слуху або автоматичного фіксування спеціальних жестів.

Для досягнення мети необхідно вирішити такі завдання:

- проаналізувати сучасні системи розпізнавання жестової мови;
- здійснити класифікацію та проаналізувати існуючі методи розпізнавання жестової мови;
- виявити особливості азбуки та мови жестів;
- проаналізувати літературні джерела щодо апробації результатів розпізнавання жестової мови;
- виявити особливості розроблення методів розпізнавання жестової мови;
- здійснити вибір інструментального засобу для програмної реалізації методу розпізнавання жестової мови;
- програмно реалізувати метод розпізнавання жестової мови;
- протестувати розроблений застосунок та провести аналіз результатів;
- виявити перспективи подальшої роботи.

## 2 ОСОБЛИВОСТІ РОЗРОБЛЕННЯ МЕТОДІВ РОЗПІЗНАВАННЯ ЖЕСТОВОЇ МОВИ

### 2.1 Механізм оброблення зображення руки

Однією з головних задач розпізнавання руки є ідентифікація показаного статичного жесту при наявності множини еталонних конфігурацій руки.

Наведемо проблеми, з якими можуть стикатися системи розпізнавання жестів руки [31, 32]:

- необхідність розпізнавання жестів у реальному часі;
- відмінності в розмірі долоні / руки у різних людей;
- особливості жестів, властиві різним людям;
- шуми, присутні в зображеннях руки.

Вилучення зображення руки з дальностного зображення розглядається як сукупність двох підзадач [33]:

- обчислення позиції довільної точки долоні руки в дальностному зображенні;
- витягнення зображення руки по позиції точки долоні.

Розглянемо більш детально другу підзадачу.

Позначимо через  $dist(a, b)$  відстань між точками  $a, b \in R^n$ .

Нехай є розпізнана точка долоні  $(x_c, y_c)$  в дальностному зображенні  $d(x, y)$ . Розглянемо модель спостережуваного об'єкта у вигляді множини просторових точок  $\{(x, y, d(x, y))\}$ .

Крок 1. Створюється сфера навколо точки  $(x_c, y_c, d(x_c, y_c))$  з радіусом:

$$\lambda = \left[ \frac{C}{d(x_c, y_c)} \right], \quad (2.1)$$

де  $C$  – деяка константа, яка залежить від характеристик конкретного сенсора.

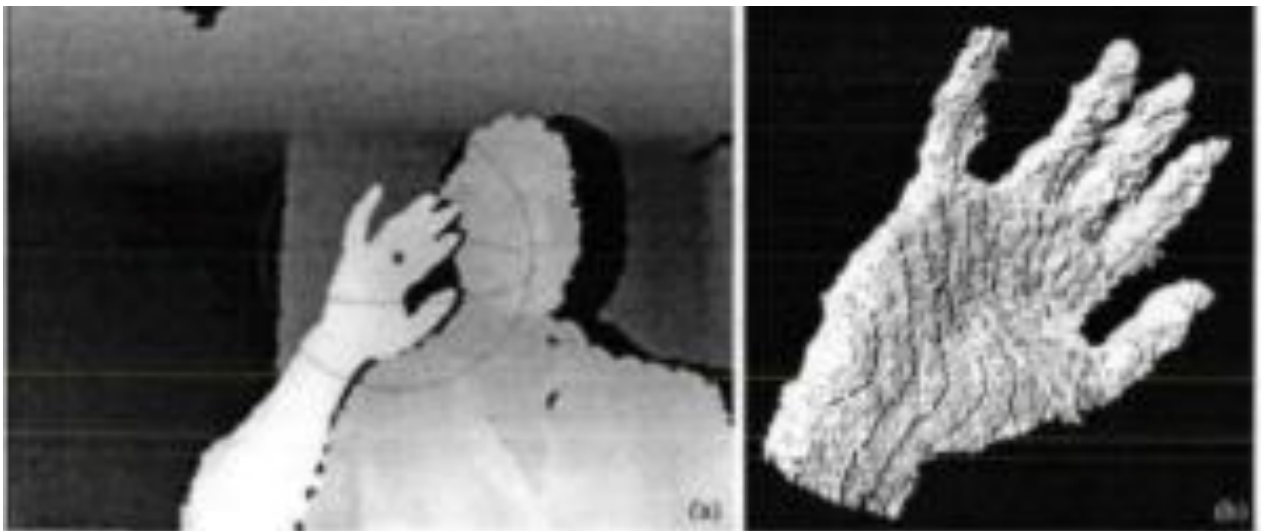
Крок 2. Усі точки дальностного зображення помічаються  $(x, y)$ , для яких  $(x, y, d(x, y))$  не входять до побудованої сфери, тобто ті, що відповідають умові  $dist(x, y, d(x, y)), (x_c, y_c, d(x_c, y_c)) > \lambda$ .

Крок 3. Усі точки дальностного зображення помічаються  $(x, y)$ , які не є  $r_0$ -зв'язаними з розпізнаною точкою долоні  $(x_c, y_c)$  у множині не помічених точок.

Крок 4. На основі вихідного зображення створюється нове дальностне зображення з розміром  $(2\lambda + 1) \times (2\lambda + 1)$ , слід зазначити, що помічені точки визначаються  $d(x, y) = 0$ .

Оброблення (витягнення) зображення руки здійснюється за час, що оцінюється  $O(n)$ , де  $n$  – кількість точок вихідного зображення [34, 35].

На рисунку 2.1(а) показані дальностне зображення людини у вигляді полутонового зображення, розпізнана точка долоні та сфера навколо цієї точки. На рисунку 2.1(б) показано виділене зображення руки [36].



(a)

(б)

Рисунок 2.1 – Зображення людини у вигляді полутонового зображення:

(а) – дальностне зображення людини, розпізнана точка долоні та сфера навколо знайденої точки; (б) – витягнення зображення руки

Зауважимо, що в наведеному зображенні рука знаходиться досить близько до обличчя людини і закриває частину обличчя, що не є перешкодою для отримання зображення руки [37].

Теорема 2.1 Множина точок  $\{(x, y)/d(x, y) > 0\}$ , отриманих алгоритмом витягнення зображення руки, є дискретною фігурою у витягнутому дальностному зображенні [38].

Доведення. Позначимо  $A = \{(x, y)/d(x, y) > 0\}$ . Для доведення теореми необхідно і достатньо довести, що множина  $A$  є  $r_0$ -зв'язаною та множина  $A$  не є підмножиною іншої  $r_0$ -зв'язаної множини [39].

Нехай  $\forall (x_0, y_0), (x'_0, y'_0) \in A \Rightarrow (x_0, y_0)$  та  $(x'_0, y'_0)$  є  $r_0$ -зв'язаними з точкою  $(\lambda + 1, \lambda + 1)$ . Звідси виходить, що  $\exists (x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$  та  $(x'_0, y'_0), (x'_1, y'_1), \dots, (x'_m, y'_m) \in A$ , такі, що  $(x_{i-1}, y_{i-1})$  та  $(x_i, y_i)$ ,  $(x'_{j-1}, y'_{j-1})$  та  $(x'_j, y'_j)$  є  $r_0$ -зв'язаними сусідами для усіх значень  $i \in \{1, 2, \dots, n\}$ ,  $j \in \{1, 2, \dots, m\}$ ,  $(x_n, y_n) = (x'_m, y'_m) = (\lambda + 1, \lambda + 1)$ .

Розглянемо послідовність точок  $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n), (x'_{m-1}, y'_{m-1}), (x'_{m-2}, y'_{m-2}), \dots, (x'_0, y'_0)$ . Згідно з визначенням точки  $(x_0, y_0)$  та  $(x'_0, y'_0) \in r_0$ -зв'язаними, що, в свою чергу, означає, що множина  $A$  є  $r_0$ -зв'язаною.

Теорема доведена. Далі вважається, що отримано витягнуте дальностне зображення руки.

## 2.2 Принципи розпізнавання складових кисті

У більшості випадків жести класифікують на два типи: статичні та динамічні. Під статичним жестом розуміється стан долоні у певний час. Статичні жести іноді називають позами, їхнє розпізнавання відбувається на статичному зображенні. Динамічні жести є послідовністю статичних жестів, та розпізнавання їх відбувається на відеопослідовності [40].

Розглянемо, розпізнавання складових кисті за допомогою статичного методу, який побудовано на розпізнаванні геометрії кисті руки (яка є унікальною біометричною характеристикою людини) за допомогою спеціальних пристроїв, що дозволяють отримувати тривимірний образ кисті руки (деякі виробники сканують форму декількох пальців) [41].

Отримані дані використовують для отримання унікальної згортки, що однозначно ідентифікує людину [42].

Існує два основних підходи до використання геометричних характеристик кисті руки. Перший з них ґрунтується на геометричних характеристиках руки. Другий вводить ще і образні характеристики руки (образи на стиках між фалангами пальців та візерунки кровоносних судин).

На рисунку 2.2 показано візерунок на долоні, що складається з п'яти основних ліній (зліва) та контрольних точок і 17 геометричних ознак руки (справа) [43].

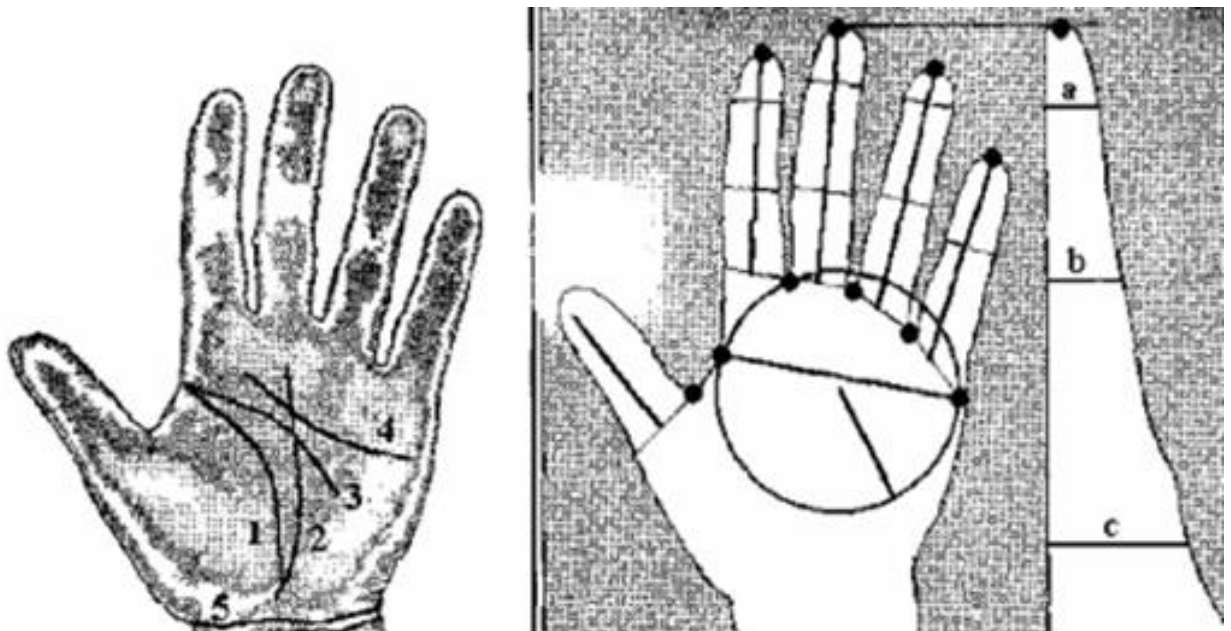


Рисунок 2.2 – Зображення долоні людини

Основними геометричними ознаками є:

- ширина долоні;
- радіус вписаного в долоню кола;

- довжина пальців;
- ширина пальців;
- висота кисті руки в трьох місцях.

Усі ці ознаки об'єднуються в єдиний вектор значень [43].

Метод ідентифікації по вектору значень досить простий.

Спочатку з користувача знімають декілька проєкцій його руки. Для кожної з цих проєкцій формується свій вектор значень. На основі декількох векторів значень створюється спеціальний клас. Далі всі ознаки в класі усереднюються, формуються ознаки еталонного образу (знаходиться центр класу) [44-55].

У процесі роботи вихідні образи можуть модифікуватися. У разі успіху порівняння нового образу з еталоном, він може бути включений до класу вихідних ознак. Порівнювати ж між собою два образи можна за декількома критеріями. Найбільш очевидний – найменша відстань від досліджуваного образу до еталону.

Складніший метод передбачає аналіз знімання чотирьох характеристик, три з яких – характерні розміри, а четверта – півтонове зображення складок шкіри на згині між фалангами. Такий метод фактично унеможлиблює хибність приладу.

У даній роботі розглянуто динамічний тип жестів.

Динамічне розпізнавання жестів за спрямованими імпульсними нейронними мережами (PCNN) для взаємодії людини з роботом у режимі реального часу запропоновано китайськими вченими і розкрито в роботі [56].

У джерелі [57] пропонується модифікований алгоритм PCNN, який називається прямим PCNN (DPCNN), для розпізнавання жестів у реальному часі без великої кількості навчальних даних. Тривимірні координати ключових каркасних точок тіла людини витягуються як вхідний об'єкт у вигляді послідовності даних, захоплених Kinect. Щоб розпізнати жест введення, подібність із ним обчислюється з матриці об'єктів, створених з елементів введення та заданого шаблону послідовностей ознак.

Для ефективного розпізнавання жестів матриця ознак перетворюється в неорієнтований граф. Схожість проблеми розпізнавання жестів стає задачею коротшого шляху. Щоб швидко і точно вирішити коротший шлях, DPCNN вибирає управління рухом, стимулює сусідні нейрони різними збудниками і зменшує ефект некорисних нейронів. Крім того, щоб скоротити час розпізнавання, введено спосіб раннього розпізнавання жестів, заснований на адаптивному вікні для розпізнавання неповних жестів. Запропонований метод раннього розпізнавання націлений на підвищення ефективності розпізнавання жестів у реальному часу та взаємодію людини і робота.

Модель розпізнавання динамічного жесту представлена у вигляді блоків на рисунку 2.3.



Рисунок 2.3 – Повна модель алгоритму розпізнавання динамічного типу жестів

Під час потоку динамічного розпізнавання жестів найбільш важливим кроком є швидке та точне обчислення найкоротшого шляху нейронною мережею з імпульсним зв'язком (PCNN). Тип сигналу і механізм оброблення PCNN подібний до фізіологічної основи в системі зорових нейронів людини. PCNN вибирає завдання з найкоротшим шляхом і обчислює подібності динамічного розпізнавання жестів.

Структуру PCNN можна розглядати як граф, що означає множину вузлів (нейронів). Працюючи із завданням найкоротшого шляху, алгоритм PCNN активує сусідні нейрони недетермінованим способом. Таким чином, обчислювальна складність PCNN залежить від масштабу мережі. У міру збільшення масштабу мережі час та витрати на обчислення також

збільшуються. Через цей недолік PCNN може займати багато часу при розпізнаванні жестів у реальному часі, незважаючи на його високу точність.

Розроблений алгоритм DPCNN вибирає напрямок потоку, стимулюючи сусідні нейрони різними збудниками, і розраховує найкоротший шлях, який сприймається як подібність між послідовностями ознак тестів і зразків. Крім того, алгоритм раннього розпізнавання, заснований на адаптивному вікні для кращої взаємодії між людиною та роботами.

Розглянемо бінарне зображення руки з функцією інтенсивності  $g(x,y)$ , отримане на основі далекого зображення руки  $d(x,y)$  за допомогою відповідного перетворення [58].

Скелетом дискретної фігури у бінарному зображенні називається пов'язана множина точок, шириною один піксель, які рівновіддалені від граничних пікселів фігури.

Для коректного математичного визначення скелета фігури використовують безперервну модель фігури, яка є апроксимацією дискретної моделі, де фігура визначається як зв'язана замкнута область на площині, обмежена кінцевим числом кривих, що не перетинаються. Жордановою кривою називають образ кола при безперервному ін'єктованому її відображенні в евклідову площину, де відображення вважається ін'єктованим, якщо дві різні точки прообразу відображаються в дві різні точки образу [58].

Нехай  $F$  є безперервна апроксимація дискретної фігури, тоді можна надати такі визначення.

Порожнім колом радіусу  $r$  фігури  $F$  називається замкнена множина точок:

$$S_r(x_0, y_0) = \{(x, y): dist((x, y), (x_0, y_0)) \leq r\}, \quad (2.2)$$

таке, що

$$S_r(x_0, y_0) \subset F, \quad (2.3)$$

$$r \geq 0. \quad (2.4)$$

Максимальним порожнім колом називається порожнє коло, яке не міститься в жодному іншому порожньому колі [59].

Скелетом фігури називається множина центрів її всіх максимальних порожніх кіл [60].

Алгоритми побудови скелетів дискретних фігур у бінарному зображенні можна розділити на 4 основні групи:

- алгоритми, засновані на обчисленні відстані точок до кордону фігури;
- алгоритми, засновані на обчисленні діаграми Вороного;
- алгоритми ітеративного витончення;
- алгоритми, засновані на безперервному поданні фігури.

Перша група алгоритмів працює в такий спосіб. Для бінарного зображення генерується карта відстаней, яка зберігає відстані від кожної не граничної точки зображення до найближчої граничної точки. Скелет фігури визначається локальними екстремумами карти відстаней (рис. 2.4).

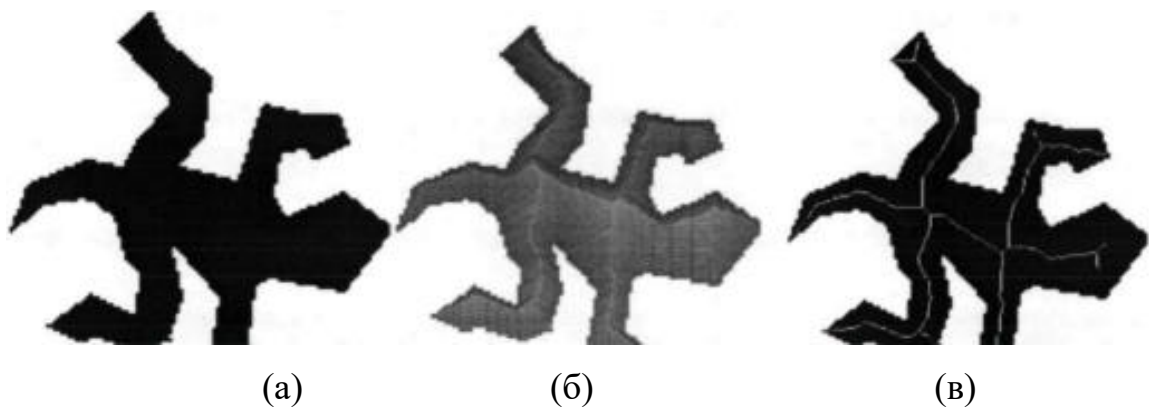


Рисунок 2.4 – Етапи алгоритму: (а) бінарне зображення; (б) карта відстаней у вигляді тривимірного об'єкта, де третя координата – це відстань точки до найближчої граничної точки; (в) скелет фігури [60]

Для заданої кінцевої множини об'єктів діаграма Вороного – це розбиття метричної площини на області, в яких знаходяться ці об'єкти, таким чином, що кожному об'єкту відповідала область площини, утворена точками, які знаходяться ближче до цього об'єкта, ніж до іншого.

Об'єктом у бінарному зображенні може виступати двовимірний пункт або лінійний відрізок. Для знаходження скелета фігури на множині точкових відрізків, що апроксимують межу фігури, будується діаграма Вороного, з якої далі виділяється скелет (рис. 2.5) [60].

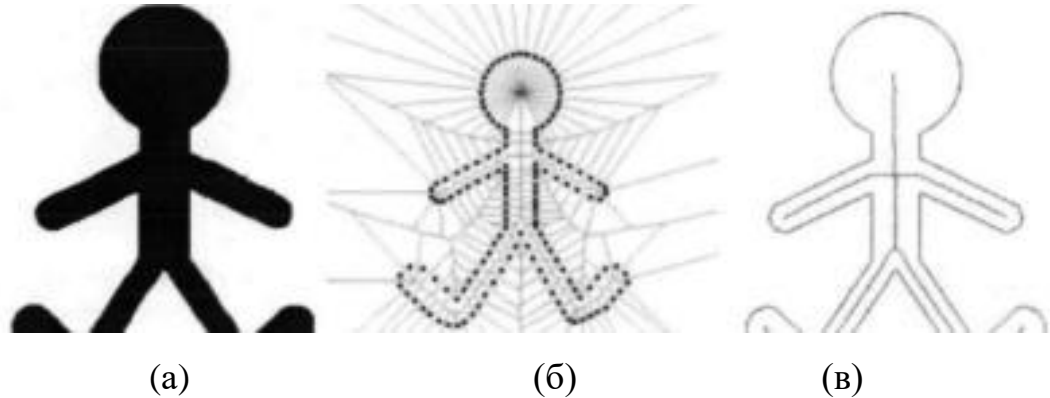


Рисунок 2.5 – Етапи алгоритму: (а) бінарне зображення; (б) діаграма Вороного для множини точок, що апроксимують межу фігури; (в) скелет фігури

Знаходження скелета фігури за допомогою витончення – це ітеративна редуція (ерозія) фігури, що нагадує поширення вогню в полі, коли вогонь одночасно починається по всіх краях і поступово рухається до центру поля [61]. Скелет фігури, за цією аналогією, є множиною точок, в яких відбувається зустріч кількох вогненних фронтів (рис. 2.6).



Рисунок 2.6 – Побудова скелета фігури у вигляді витончення

Четверта група алгоритмів, що використовує створення скелетного подання дискретної фігури, апроксимує дискретну межу фігури безперервної, після чого створює геометричний скелет безперервної фігури.

У більшості випадків апроксимація кордону здійснюється за допомогою багатокутника, а скелет, що складається з прямих ліній та сегментів парабол [62] або кривих Безьє першого та другого порядку [63], знаходиться за допомогою побудови узагальненої діаграми Вороного для багатокутної фігури [64] (рис. 2.7).

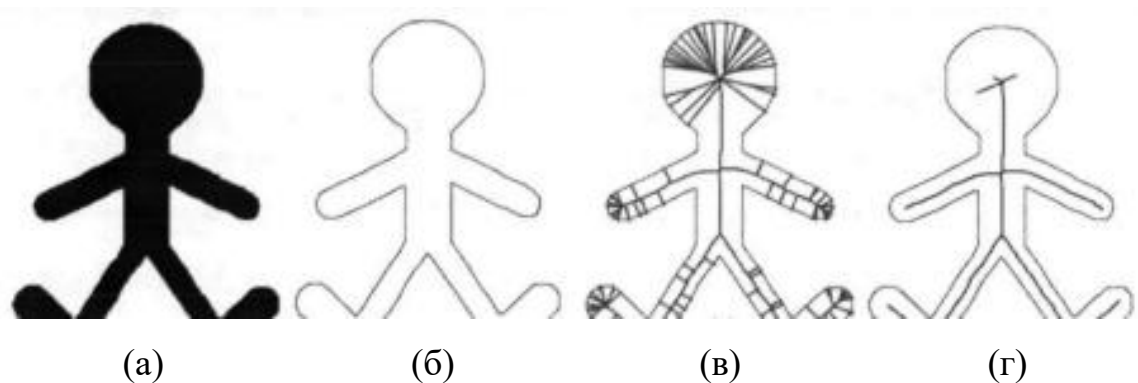


Рисунок 2.7 – Побудова скелета фігури за допомогою апроксимації дискретної фігури безперервної: (а) дискретна фігура; (б) апроксимований безперервний кордон; (в) безперервний скелет; (г) безперервний скелет після «стрижки»

Алгоритми, що базуються на обчисленні відстані точок до кордону фігури та ітеративного стоншення, дозволяють отримати скелет зображення в вигляді нового бінарного зображення, що обмежує можливості класифікації зображень та вимагає зберігання великих даних у базі еталонні конфігурації руки.

Отримати безперервний скелет фігури можна за допомогою діаграм Вороного або апроксимацією дискретної безперервної фігури та обчисленням скелета безперервної фігури. Обидва методи дозволяють отримати скелет фігури у вигляді плоского графа, але перший метод більш чутливий до нерівностей межі фігури, що можна спостерігати в [65].

Зауважимо, що, незважаючи на те, що всі розглянуті алгоритми призначені для обчислення середньої осі фігури, при обробці одного зображення різними алгоритмами на виході можуть бути отримані різні скелети (рис. 2.5 (в) і рис. 2.7 (в)).

Наприклад, існують різні способи апроксимації межі дискретної фігури за допомогою безперервного багатокутника, звідки слідує, що алгоритми четвертої групи можуть на виході видати різні скелети для однієї дискретної фігури. Іншу особливість геометричних скелетів можна спостерігати на рисунку 2.7 (в), де видно дуги, створені через нерівності межі дискретної фігури. Незначні зміни межі фігури можуть спричинити істотні зміни скелета фігури. Це означає, що завдання скелетизації є некоректно поставленим завданням [65]. Ці завдання вирішуються методом регуляризації – пошуком наближених рішень, стійких до малих змін форми фігури [60, 62, 63, 66].

У разі безперервних скелетів, результатом регуляризації зазвичай є підграф початкового скелета, який називають базовим скелетом фігури, а процес видалення дуг, які не входять до базового скелета, називають стрижкою скелета. Позначимо через  $V_G$  об'єднання всіх максимальних порожніх кіл фігури  $F$ , центри яких лежать на скелеті  $G$ .  $V_G$  називається силуетом скелета  $G$  [60].

Базовим скелетом фігури  $F$  називається мінімальний укорочений підграф  $G'$  її скелета  $G$ , для силуету  $V_G$  якого виконано умову:

$$H(F, V_{G'}) \leq \varepsilon, \quad (2.5)$$

де  $\varepsilon$  – задана позитивна величина;

$H(F, V_{G'})$  – хаусдорфова відстань між фігурою  $F$  та силуетом  $V_{G'}$  [67].

На практиці для стрижки скелета використовують інші методи, як, наприклад, видалення дуг, у центрі яких радіуси порожніх кіл менше заданої величини. Цю величину зазвичай називають регулювальним параметром. На рисунку 2.7 (г) показаний скелет фігури після стрижки, з регулюючим

параметром 1. Можна помітити, що після стрижки в скелеті видаляються всі дуги, які були створені через нерівності межі фігури.

Як згадувалось вище, далекосним зображенням характерний значний шум на межах фігури (рис. 2.1 (б), рис. 2.8 (а)), внаслідок чого регуляризації скелета з малими параметрами стрижки недостатньо для отримання скелета руки, інваріантного до даного шуму.

На рисунку 2.8 показано бінарне зображення руки, початковий скелет і скелети, отримані після стрижки з параметрами 1 та 4. Оптимальний параметр стрижки для бінарних зображень, отриманих на основі далекосних зображень руки, емпірично, дорівнює значенню 5.

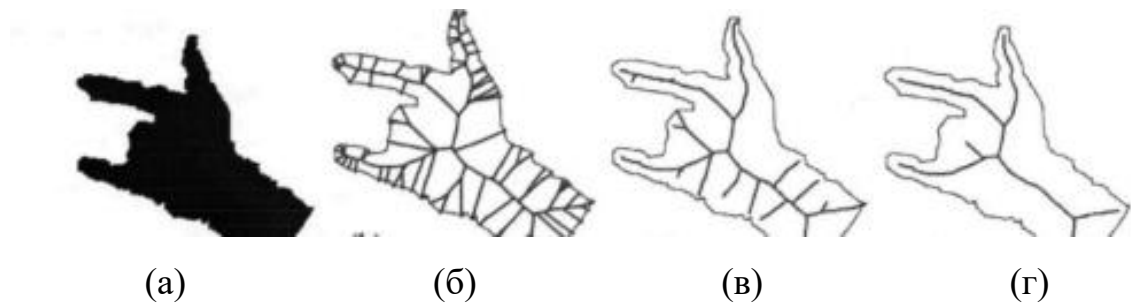


Рисунок 2.8 – Етапи алгоритму: (а) бінарне зображення руки; (б) скелет руки; (в) скелет руки після стрижки з регулюючим параметром 1; (г) скелет руки після стрижки з регулюючим параметром 4

Стрижка скелета, звісно, є єдиним методом отримання скелета, інваріантним до малих змін межі фігури. Отримати такий скелет можна також за допомогою початкового згладжування контуру фігури.

Нехай задано список точок  $L = \{p_i\}_{i=1}^n$ , що належать контуру фігури, упорядковані за годинниковою стрілкою. Алгоритм згладжування точок контуру здійснюється за допомогою кривих Безьє [68]:

Вхідні дані:

- точки контуру  $\{p_i\}_{i=1}^n$ ;
- ініціалізація:  $i := 1$  або  $2$ ;
- $p'_i := p_i$ .

Поки  $i \leq n - 2$ , то

$$p'_{i+1} := \frac{1}{4}p'_i + \frac{1}{2}p_{i+1} + \frac{1}{4}p_{i+2}, \quad (2.6)$$

$$i := i + 2.$$

Виведення: згладжені точки контуру руки  $\{p_1^i\}_{i=1}^n$ .

Результат застосування алгоритму на довільній кривій показано на рисунку 2.9.

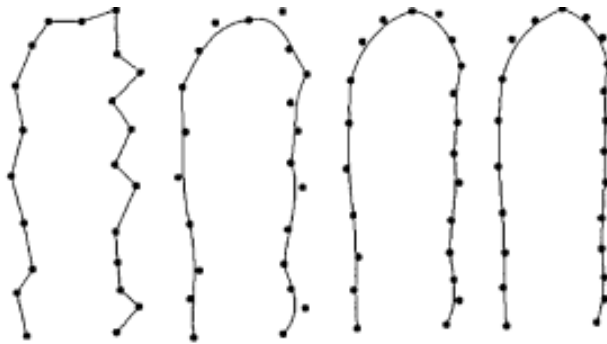


Рисунок 2.9 – Згладжування контуру за допомогою кривої Безьє після 1, 2 і 3 ітерацій

Цей підхід можна застосувати для скелетизації фігури за допомогою діаграм Вороного. Використовуючи алгоритми Suzuki та Abe [69] для бінарного зображення, знайдемо граничні точки руки. Виконаємо фільтрацію отриманих точок контуру так, щоб будь-які дві точки не знаходилися ближче за задану фіксовану відстань.

Згладжування точок проведемо за допомогою кількох ітерацій алгоритму описаного вище. Для отриманих точок створимо діаграму Вороного і потім видалимо зайві дуги, отримуючи тим самим скелет фігури. Результат застосування запропонованого алгоритму для бінарного зображення руки наведено на рисунку 2.10.

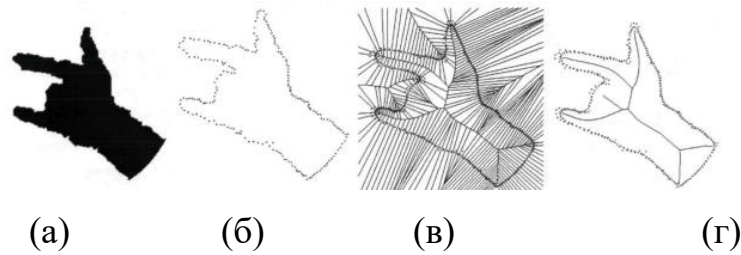


Рисунок 2.10 – Результат застосування запропонованого алгоритму для бінарного зображення руки: (а) бінарне зображення руки; (б) обрані точки контуру; (в) діаграма Вороного для згладжених точок; (г) скелет руки

Можна помітити, що у отриманому скелеті руки відсутні дуги, що створені через нерівності контуру фігури руки.

Таким чином, пропонується перетворити далекосне зображення на бінарне, а потім застосувати один з наступних методів скелетизації далекосного зображення руки:

- метод скелетизації за допомогою пошуку граничних точок та створення діаграми Вороного для цих точок;
- метод скелетизації за допомогою подання дискретної фігури у вигляді безперервної фігури та обчислення скелета цієї фігури.

Обидва методи дозволяють у реальному часі отримувати скелет фігури руки, інваріантний до шумів у контурах фігури.

Єдиним значним недоліком даного підходу є втрата первісної інформації про жест руки під час перетворення далекосного зображення на бінарне. На рисунку 2.11 [68] показано далекосне та бінарне зображення руки зі скелетом фігури руки.

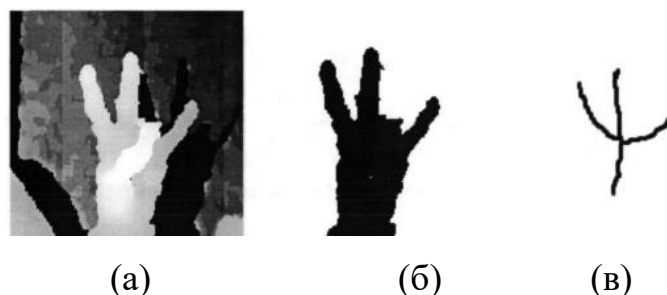


Рисунок 2.11 – Отриманий результат: (а) далекосне зображення руки; (б) бінарне зображення руки; (в) скелет руки

Можна помітити, що у скелеті руки немає жодної інформації про позицію великого і безіменного пальців руки. Цей недолік не мають тривимірні скелети, отримані на основі перетворення далекосного зображення  $d(x,y)$  у хмару точок  $\{(x,y,d(x,y))\}$ .

### 2.3 Методика побудови тривимірного подання верхньої кінцівки

Враховуючи те, що двовимірні скелети зображення руки не зберігають інформацію щодо відстані точок руки до сенсора, а криволінійні скелети тривимірного представлення зображення руки не дозволяють отримати скелет фігури в реальному часі, тому пропонується новий метод скелетизації, що дозволяє в реальному часі обчислити тривимірний скелет фігури в дальностному зображенні.

Нехай задано дальностне зображення руки  $d(x, y)$ .

Створимо функцію

$$d' : R^2 \rightarrow R, \quad (2.7)$$

де  $d(x, y)$  інтерполюється таким чином:

$$d'(a, b) = d(x', y'); \quad (2.8)$$

$$(x', y') = \arg \inf_{(x,y) \in W^2} \text{dist}((x, y), (a, b)), (a, b) \in R^2. \quad (2.9)$$

Позначимо через  $G$  геометричний скелет фігури руки в бінарному зображенні, яку завжди можна представити у вигляді плоского графа

$$G = (V, E), \quad (2.10)$$

де

$$V \subset R^2; \quad (2.11)$$

$$E \subseteq V^2. \quad (2.12)$$

Побудуємо тривимірний скелет  $G' = (V', E')$ ,  $V' \subset R^3$ ,  $E' \subseteq V'^2$ , керуючись певним алгоритмом.

Вхідні дані:  $d'(x, y)$ , інтерполяція  $d(x, y)$ ; багатокутник  $M$ , що апроксимує контур фігури руки; константа  $\delta$ , що визначає максимально допустиму довжину ребра графа.

Крок 1. Необхідно створити скелет  $G = (V, E)$  многокутника  $M$ .

Крок 2. Створюємо пусті множини вершин  $V'$  та ребер  $E'$ .

Крок 3. Перевірка довжини кожного ребра  $(u, v) \in E$ .

Якщо  $dist(u, v) \leq \delta$ , то існує декілька ситуацій:

– додаємо

$$u' = (x_u, y_u, d'(x_u, y_u)), \quad (2.13)$$

$$v' = (x_v, y_v, d'(x_v, y_v)) \in V', \quad (2.14)$$

де

$$u = x_u, y_u; \quad (2.15)$$

$$v = x_v, y_v; \quad (2.16)$$

– додаємо  $(u', v')$  в  $E'$ ;

– розбиваємо ребро  $(u, v)$  на  $\left\lceil \frac{dist(u, v)}{\delta} \right\rceil$  рівних частин;

– додаємо отримані вершини та ребра в  $V'$  та  $E'$  як в перших двох етапах третього кроку.

Тривимірний скелет вважається побудованим, якщо виконаний третій крок у алгоритмі. Висновок: тривимірний скелет  $G' = (V', E')$ .

Зауважимо, що наведений вище алгоритм методу не порушує зв'язність графа  $S = (V, E)$  та може бути використаним для відновлення початкового бінарного зображення, якщо разом з кожною точкою скелета зберігати радіальну функцію – радіус максимальної порожньої сфери в даній точці, та зберігати інформацію про поверхні фігури.

Обчислювальна складність методу скелетизації становить:

$$O(n \log n), \quad (2.17)$$

де  $n$  – число вершин багатокутника, що апроксимує контур фігури.

Нехай задані дискретна фігура в дальностному зображенні  $d(x, y)$  та багатокутник з  $n$  вершинами, що апроксимує контур даної фігури.

Запропонований метод скелетизації містить наступні основні етапи:

- обчислення двовимірного скелета  $\sim O(n \log n)$  [45];
- побудова тривимірного скелета  $\sim O(k)$ , де  $k$  – кількість ребер двовимірного скелета.

Згідно [45] скелет фігури є підмножиною її діаграми Вороного, кількість ребер якого асимптотично можна оцінити як  $O(n)$ . Тоді має місце нерівність:  $O(n \log n) + O(k) \leq O(n \log n) + O(n) \leq O(n \log n)$ .

На рисунку 2.12 представлено побудований тривимірний скелет руки, з різних сторін спостереження [57].

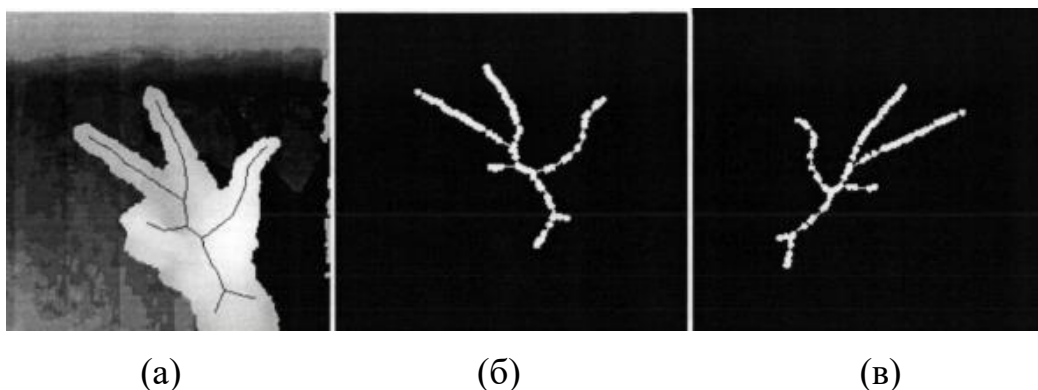


Рисунок 2.12 – Скелети: (а) тривимірні скелети дальностного зображення руки; (б) та (в) скелети з різних сторін спостереження

## 2.4 Методика розпізнавання азбуки жестів руки

Азбука рук (дактилологія) – своєрідна форма спілкування, що відтворює за допомогою пальців рук орфографічну форму слова, оскільки використовує дактильну абетку. Тим самим дактилологія поєднує в собі ознаки як мовлення, оскільки використовується для оперативного спілкування, так і письмової, оскільки має форму послідовності знаків відповідно до норм орфографії [46].

Іноді люди без вад слуху, приймають пальцевий алфавіт за жестову мову. Однак, це різні речі: у той час як пальцевим алфавітом передають звуки мови (літера за буквою), жестові знаки репрезентують цілі слова.

Елементи дактильної абетки аналогічні буквам писемної мови.

Дактилеми (літери з дактильного алфавіту) – різні положення пальців, що відтворюють знаки, функціонально аналогічні буквеному алфавіту; за обрисом деякі з них віддалено нагадують літери друкованого шрифту (зокрема, «о», «м», «г», «ш»).

Кожне положення пальців означає букву (рис. 2.13) [69].

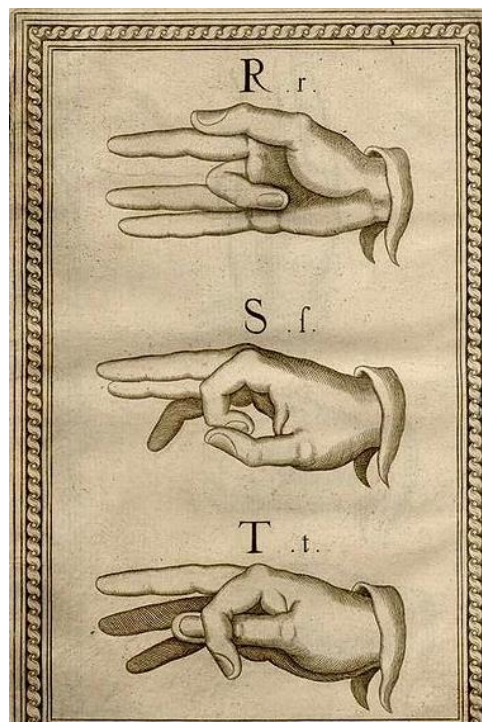


Рисунок 2.13 – Приклад використання жестової мови

Процес спілкування на дактилі відбувається таким чином, що той, хто говорить, показує букви на дактилі, а той, хто сприймає, стежить за рухом руки візуально. Якщо ж зір людини не дозволяє бачити знаки, і вона сприймає їх відчутно, то таке спілкування називається дактильно-контактною мовою [64]. Якщо ж при спілкуванні зі сліпоглухими людьми на долоні іншої людини пальцями відтворюють контур букв, то це вже не дактилологія, а лист на долоні або, так звана, дермографія. Вона використовується, якщо той, хто говорить, не володіє спеціальною абеткою.

Хоча пальцеві алфавіти можуть бути досить прийнятним засобом спілкування, у даного способу спілкування є також негативні фактори при діалозі, жоден з них не передбачає великих літер, символів пунктуації або символу пропуску між словами. Використання таких алфавітів обмежене щодо передачі назв та імен, іноземних слів і слів, що виражають емоції; якщо обидва співрозмовники не знають добре, про що йдеться.

Розділяють чотири типи азбуки рук:

- дактильна мова;
- маноральна мова;
- одноручна мова;
- дворучна мова.

При дактильному мовленні точно дотримуються традиційних норм правопису. У маноральній системі промови, тобто при візуальному зчитуванні з губ людина, що говорить, руками подає співрозмовнику допоміжні знаки, що допомагають глухому вловити різницю між дзвінкими і глухими, що важко помітити візуально (г – к; д – т; б – п і т.д.), твердими і м'якими (т – ть, с – сь і т.д.) звуками (наприклад, кисть, що лежить на грудях, означає дзвінкий, а віддалена від грудей – глухий звук).

У більшості «пальцевих абеток» використовується одна рука. Вважається, що це «зручніше», тому що друга рука вільна для інших дій. Проте, є системи, де використовуються дві руки (наприклад, у мовах сім'ї британської мови жестів) – британській, австралійській, новозеландській.

Найраніша згадка про алфавіт, що використовує дві руки, зустрічається у книзі «Дідаскалокофус, або Керівництво для глухих і німих людей» (1680 рік). Серед систем, що згодом з'явилися, та використовували дві руки була Дігіті-Лінгва (1698 рік). Сучасний пальцевий алфавіт із використанням двох рук, як вважають, походить від однієї з версій «Дігіті-Лінгва».

Для більш детального опису механізму розпізнавання азбуки жестів руки розглянемо на рисунку 2.14 деталізацію формальної постановки задачі розпізнавання жестів руки.

На вхід подається зображення.

Механізмами виступають автоматизовані алгоритми обробки зображень і алгоритми розпізнавання образів, участь користувача і програмно-технічних засобів.

З урахуванням наявної бази даних жестів ручної азбуки глухонімих видається результат розпізнавання жестів рук (у вигляді текстового напису), за умови, що такі присутні на вхідному зображенні.

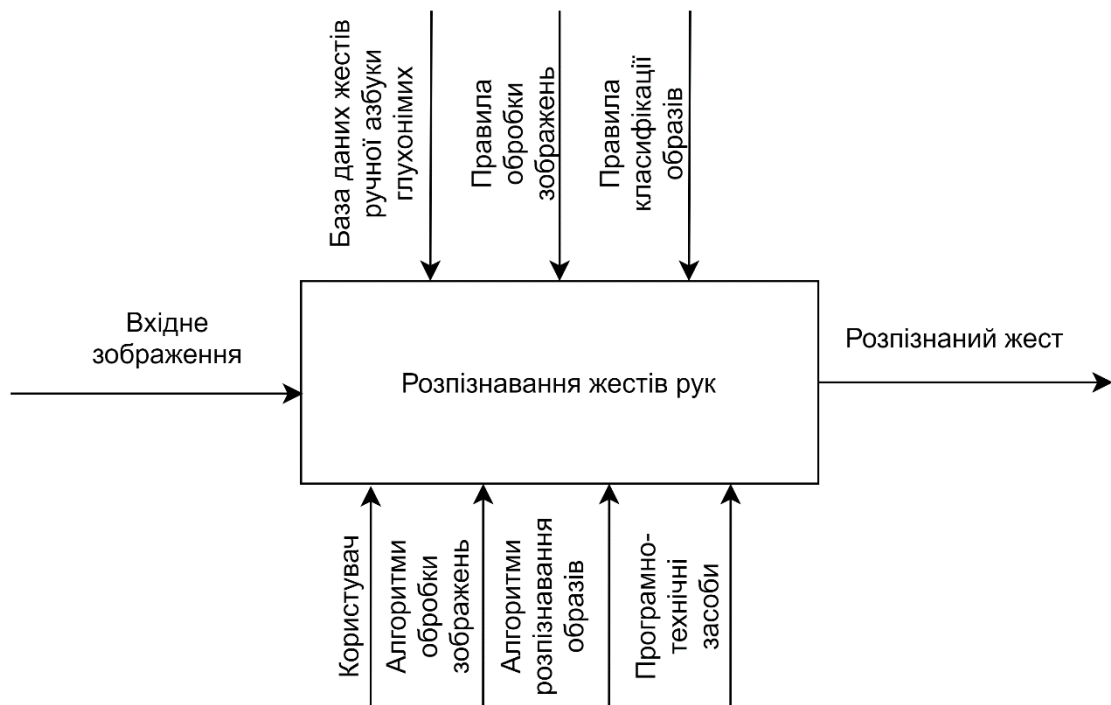


Рисунок 2.14 – Контекстна діаграма процесу розпізнавання жестів рук

Декомпозиція задачі розпізнавання жестів рук здійснюється відповідно до діаграми на рисунку 2.15.

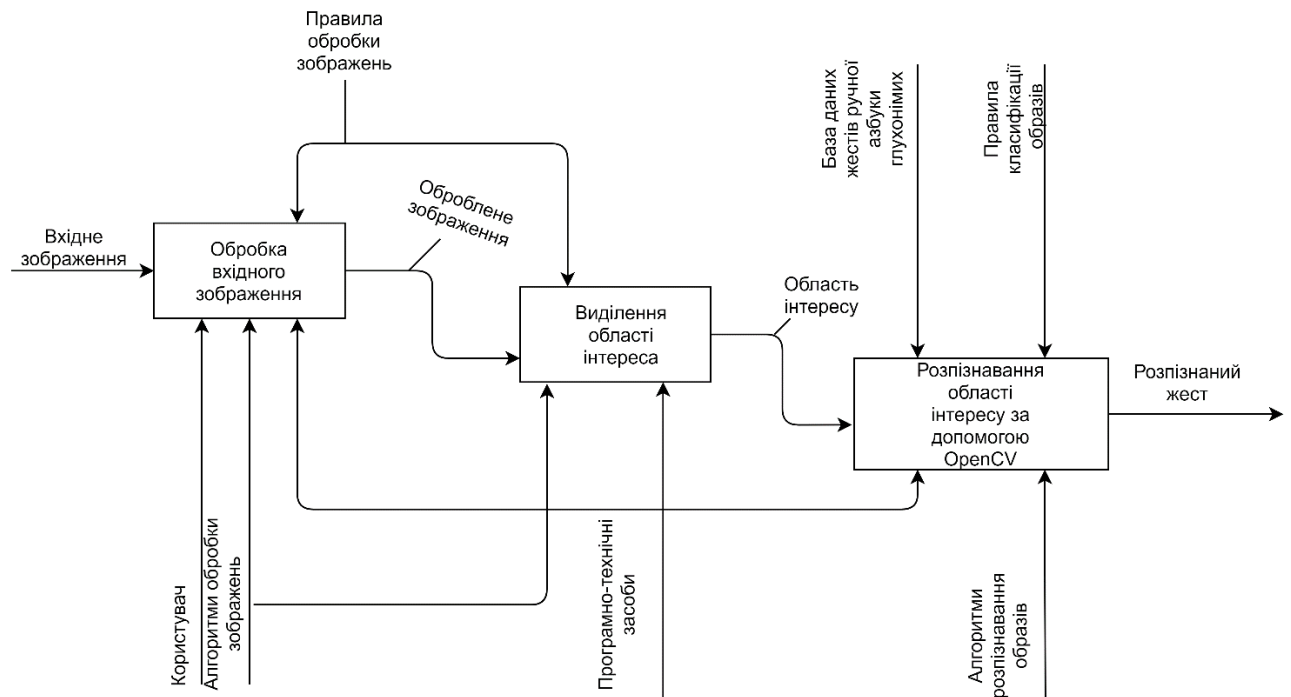


Рисунок 2.15 – Декомпозиція задачі розпізнавання жестів рук

Декомпозиція задачі розпізнавання жестів рук складається з наступних етапів:

- обробка вхідного зображення (задача виявлення);
- виділення області інтересу (задача відстеження);
- розпізнавання області інтересу (задача розпізнавання).

Завдання виявлення складається з процедур перетворення колірних компонентів, обчислення гістограми та зворотної проєкції, цифрової обробки.

Завдання відстеження включає процедури знаходження контурів та апроксимації знайденого контуру.

У даному дослідженні буде розроблено програмне забезпечення, яке дозволяє автоматизувати процес розпізнавання жестової мови у реальному часі.

### 3 РЕАЛІЗАЦІЯ ТА ДОСЛІДЖЕННЯ МЕТОДУ РОЗПІЗНАВАННЯ ЖЕСТОВОЇ МОВИ

3.1 Вибір інструментального засобу для реалізації методу розпізнавання жестової мови

Для реалізації програмного застосунку обрано мову програмування Python та бібліотеку OpenCV.

#### 3.1.1 Огляд мови Python та бібліотеки OpenCV

Python – інтерпретована мова, що дозволяє заощадити значну кількість часу, що, зазвичай, витрачається на компіляцію. Інтерпретатор можна використовувати інтерактивно, що дозволяє експериментувати з можливостями мови, писати шаблони програм або тестувати функції при розробці «знизу-вверх». Він також зручний як настільний калькулятор.

Python дозволяє писати дуже компактні й зручні для читання програми. Програми, написані мовою Python, звичайно значно коротші еквівалента на C або C++ з декількох причин:

- типи даних високого рівня дозволяють виразити складні операції однією інструкцією;
- групування інструкцій виконується за допомогою відступів замість фігурних дужок;
- немає необхідності в оголошенні змінних.

Python проста у використанні та водночас повноцінна мова програмування, що надає набагато більше засобів для структурування і підтримки великих програм, ніж shell.

З іншого боку, вона краще за С обробляє помилки, і, будучи мовою дуже високого рівня, має вбудовані типи даних високого рівня, такі як гнучкі масиви і словники, ефективна реалізація яких на С потребує значних витрат часу [69].

Завдяки більш загальним типам даних, Python застосовують до більш широкого кола задач, ніж Awk і навіть Perl, у той же час багато речей на мові Python робляться настільки ж просто.

Python дозволяє розбивати програми на модулі, що потім можуть бути використані в інших програмах. Python поставляється з великою бібліотекою стандартних модулів, які можна використовувати як основу для нових програм або як приклади при вивченні мови. Стандартні модулі надають засоби для роботи з файлами, системними викликами, мережевими з'єднаннями і навіть інтерфейсами до різних графічних бібліотек [70].

Python розширювана мова: знання С дозволяє додавати нові функції, що вбудовуються, або модулі для виконання критичних операцій з максимальною швидкістю або написання інтерфейсу до комерційних бібліотек, доступним тільки у двійковій формі. Інтерпретатор мови Python може бути вбудований у програму, написану на С, і використовувати його як розширення або командну мову для цієї програми.

Python використовується у даний час десятками тисяч програмістів в усьому світі і число людей, що використовують його, швидко зростає, подвоюється і потроюється щороку.

Python приваблює користувачів з ряду причин. Він використовується для розроблення програм і дозволяє провести розробку набагато швидше, ніж традиційні мови типу С, С++ або Java. Ця мова працює однаково добре на Windows, UNIX, Macintosh, і OS/2, може використовуватися для легкої розробки як малих застосунків чи сценаріїв, так і для розгортання великих програм.

Python пропонує доступ до масштабного і легкого у використанні комплекту 29 інструментальних засобів графічного інтерфейсу користувача.

Традиційні машинні мови типу C і Pascal мають ряд характеристик, наприклад, суворі типізація, базові типи, складні (і звичайно довгі) цикли, потреба у великих кількостях кодів для виконання відносно малих задач.

Java досить новий, але розділяє більшість характеристик, включених у цей перелік. Програмісти, знайомі з традиційними мовами погодяться, що відсутність суворі типізації полегшує роботу з Python.

OpenCV – це відкрита бібліотека комп'ютерного зору, яка призначена для аналізу, класифікації та оброблення зображень, вона використовується у таких мовах як C, C++, Python та Java.

### 3.1.2 Обґрунтування вибору середовища програмної реалізації

У рамках кваліфікаційної роботи розроблено програмний застосунок, головною задачею якого є розпізнавання жестової мови для безбар'єрного розуміння людей з вадами слуху або автоматичного фіксування спеціальних жестів.

Для реалізації обрано середовище Jupyter Notebook.

Jupyter Notebook – це середовище розроблення, де одразу можна бачити результат виконання коду та його окремих фрагментів. На відміну від традиційного середовища розроблення код можна розбити на частини та виконувати їх у довільному порядку [71].

У такому середовищі розроблення можна, наприклад, написати функцію і перевірити її роботу, без запуску програми повністю. А ще можна змінити порядок виконання коду. Можна окремо завантажити файл у пам'ять, окремо перевірити його, окремо обробити вміст.

Також в Jupyter Notebook є виведення результату одразу після фрагмента коду. Наприклад, можливо прямо в середині коду побачити побудований графік, отримати попередні цифри чи іншу візуалізацію.

Для знайомства з середовищем не обов'язково встановлювати застосунок на комп'ютер, адже робота в Jupyter Notebook можлива як на локальній машині, так і у інтернет-браузері.

Перевага такого підходу в тому, що не потрібно нічого встановлювати на комп'ютер – хмара все бере на себе, необхідно лише написати та запустити код.

Недолік такого підходу в тому, що програма може працювати не так швидко, як на локальній машині. Якщо потрібні якісь екзотичні бібліотеки, то у хмарі їх може не виявитися.

### 3.2 Етапи програмної реалізації методу розпізнавання жестової мови

Розглянемо основні етапи програмної реалізації методу розпізнавання жестової мови.

Етап 1. Створення колекції зображень з елементами жестової мови.

Лістинг 3.1 Створення колекції зображень:

```
import cv2
import os
import time
import uuid
IMAGES_PATH = 'Tensorflow/workspace/images/collectedimages'
labels = ['hello', 'thanks', 'yes', 'no', 'iloveyou']
number_imgs = 15
for label in labels:
    !mkdir { 'Tensorflow\workspace\images\collectedimages\\'+label }
    cap = cv2.VideoCapture(0)
    print('Collecting images for {}'.format(label))
    time.sleep(5)
```

```

for imgnum in range(number_imgs):
    ret, frame = cap.read()
    imgname = os.path.join(IMAGES_PATH, label, label+ '.'+ '{}.jpg'.
format(str(uuid.uuid1()))))
    cv2.imwrite(imgname, frame)
    cv2.imshow('frame', frame)
    time.sleep(2)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
    cap.release()

```

Етап 2. Додавання за допомогою застосунку LabelImg позначки з назвою жесту до кожного зображення у колекції.

Етап 3. Створення карти позначок.

Лістинг 3.2 Створення карти позначок:

```

WORKSPACE_PATH =
'C:/Users/Anastasia/RealTimeObjectDetection/Tensorflow/workspace'
SCRIPTS_PATH =
'C:/Users/Anastasia/RealTimeObjectDetection/Tensorflow/scripts'
APIMODEL_PATH =
'C:/Users/Anastasia/RealTimeObjectDetection/Tensorflow/models'
ANNOTATION_PATH = WORKSPACE_PATH+ '/annotations'
IMAGE_PATH = WORKSPACE_PATH+ '/images'
MODEL_PATH = WORKSPACE_PATH+ '/models'
PRETRAINED_MODEL_PATH = WORKSPACE_PATH+ '/pre-trained-
models'
CONFIG_PATH = MODEL_PATH+ '/my_ssd_mobnet/pipeline.config'
CHECKPOINT_PATH = MODEL_PATH+ '/my_ssd_mobnet/'

```

```

labels = [
    {'name': 'Hello', 'id': 1},
    {'name': 'Yes', 'id': 2},
    {'name': 'No', 'id': 3},
    {'name': 'Thank You', 'id': 4},
    {'name': 'I love You', 'id': 5}
]

with open(ANNOTATION_PATH + 'label_map.pbtxt', 'w') as f:
    for label in labels:
        f.write('item {\n')
        f.write('  name: {}\n'.format(label['name']))
        f.write('  tid: {}\n'.format(label['id']))
        f.write('}\n')

```

Етап 4. Створення TensorFlow-записів [71].

ЛІСТИНГ 3.3 Створення TensorFlow-записів:

```

!conda activate tensorflow

!python {SCRIPTS_PATH + '/generate_tfrecored.py'} -x {IMAGE_PATH +
'/train'} -l {ANNOTATION_PATH + '/label_map.pbtxt'} -o {ANNOTATION_PATH
+ '/train.record'}

!python {SCRIPTS_PATH + '/generate_tfrecored.py'} -x {IMAGE_PATH +
'/test'} -l {ANNOTATION_PATH + '/label_map.pbtxt'} -o {ANNOTATION_PATH
+ '/test.record'}

```

Етап 5. Завантаження попередньо підготовленої моделі TensorFlow-моделі.

Етап 6. Оновлення конфігурації для передачі навчання.

Лістинг 3.4 Оновлення конфігурації для передачі навчання:

```

import tensorflow as tf
from object_detection.utils import config_util
from object_detection.protos import pipeline_pb2
from google.protobuf import text_format

CONFIG_PATH =
MODEL_PATH+ '/' + CUSTOM_MODEL_NAME+ '/pipeline.config'
config = config_util.get_configs_from_pipeline_file(CONFIG_PATH)
config
pipeline_config = pipeline_pb2.TrainEvalPipelineConfig()
with tf.io.gfile.GFile(CONFIG_PATH, "r") as f:
    proto_str = f.read()
    text_format.Merge(proto_str, pipeline_config)
pipeline_config.model.ssd.num_classes = 2
pipeline_config.train_config.batch_size = 4
pipeline_config.train_config.fine_tune_checkpoint =
PRETRAINED_MODEL_PATH+ '/ssd_mobilenet_v2_fpnlite_320x320_coco17_tp
u-8/checkpoint/ckpt-0'
pipeline_config.train_config.fine_tune_checkpoint_type = "detection"
pipeline_config.train_input_reader.label_map_path= ANNOTATION_PATH
+ '/label_map.pbtxt'
pipeline_config.train_input_reader.tf_record_input_reader.input_path[:] =
[ANNOTATION_PATH + '/train.record']
pipeline_config.eval_input_reader[0].label_map_path =
ANNOTATION_PATH + '/label_map.pbtxt'
pipeline_config.eval_input_reader[0].tf_record_input_reader.input_path[:]
= [ANNOTATION_PATH + '/test.record']
config_text = text_format.MessageToString(pipeline_config)
with tf.io.gfile.GFile(CONFIG_PATH, "wb") as f:
    f.write(config_text)

```

Етап 7. Тестування отриманої моделі.

Етап 8. Завантаження протестованої моделі з контрольного пункту.

Етап 9. Фіксація та розпізнавання різних жестів рук у реальному часі за допомогою розробленого застосунку.

### 3.3 Тестування розробленого застосунку та аналіз результатів

Тестування програмного забезпечення є не менш важливим етапом розроблення програмного застосунку, адже воно допомагає допрацювати програмне забезпечення відповідно до вимог бізнесу та користувачів.

Після запуску сервера програми відкривається головне вікно застосунки, на перших секундах запуску має відбуватись розпізнавання жестів рук, результат роботи програми представлено на рисунках 3.1 – 3.12.

На рисунках 3.1 та 3.2 представлено результат розпізнавання жесту «Hello».

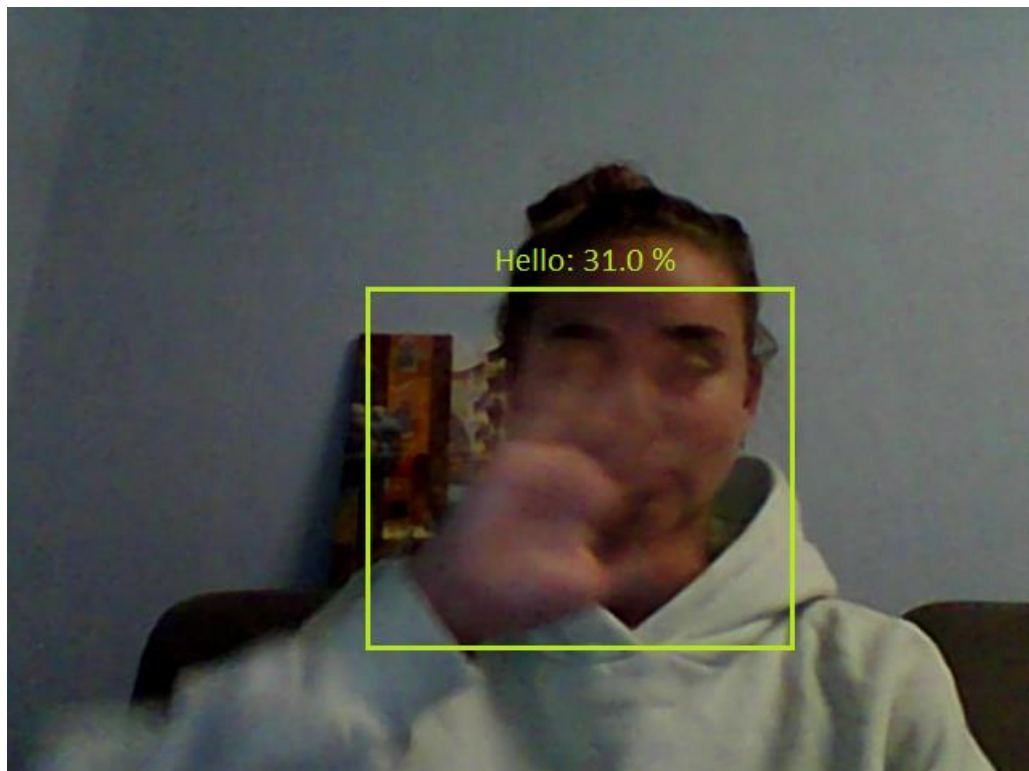


Рисунок 3.1 – Результат розпізнавання жесту «Hello» № 1

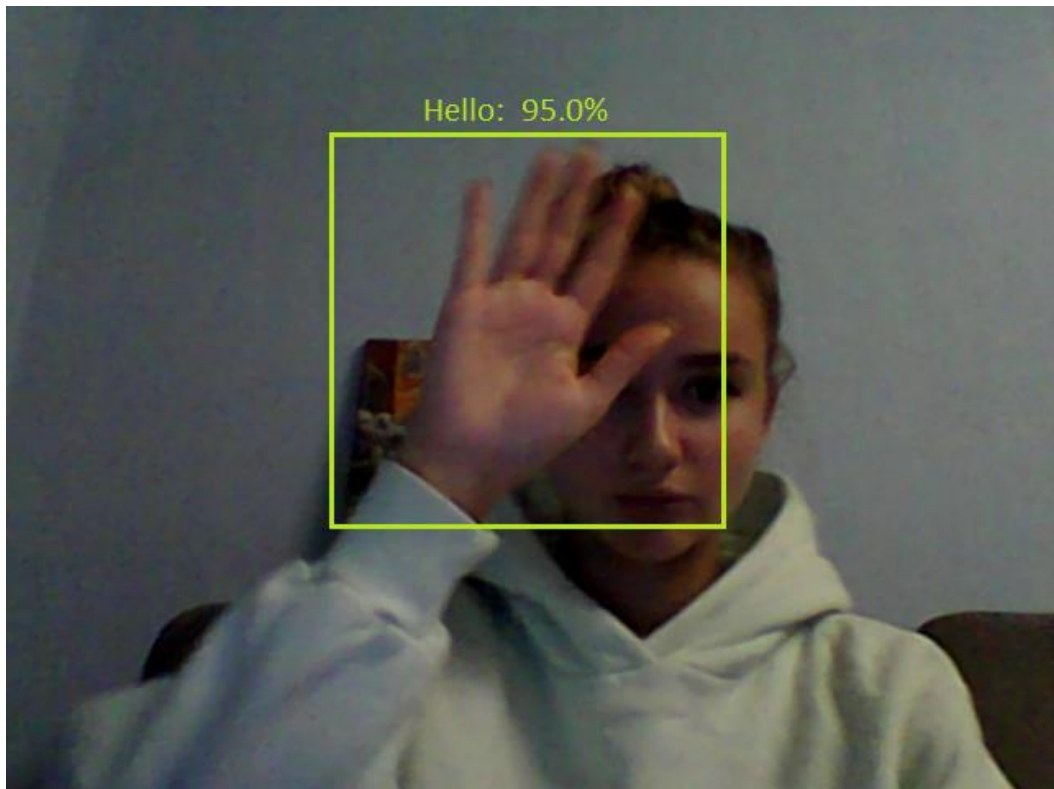


Рисунок 3.2 – Результат розпізнавання жесту «Hello» № 2

На рисунках 3.3 та 3.4 представлено результат розпізнавання жесту «I Love You».

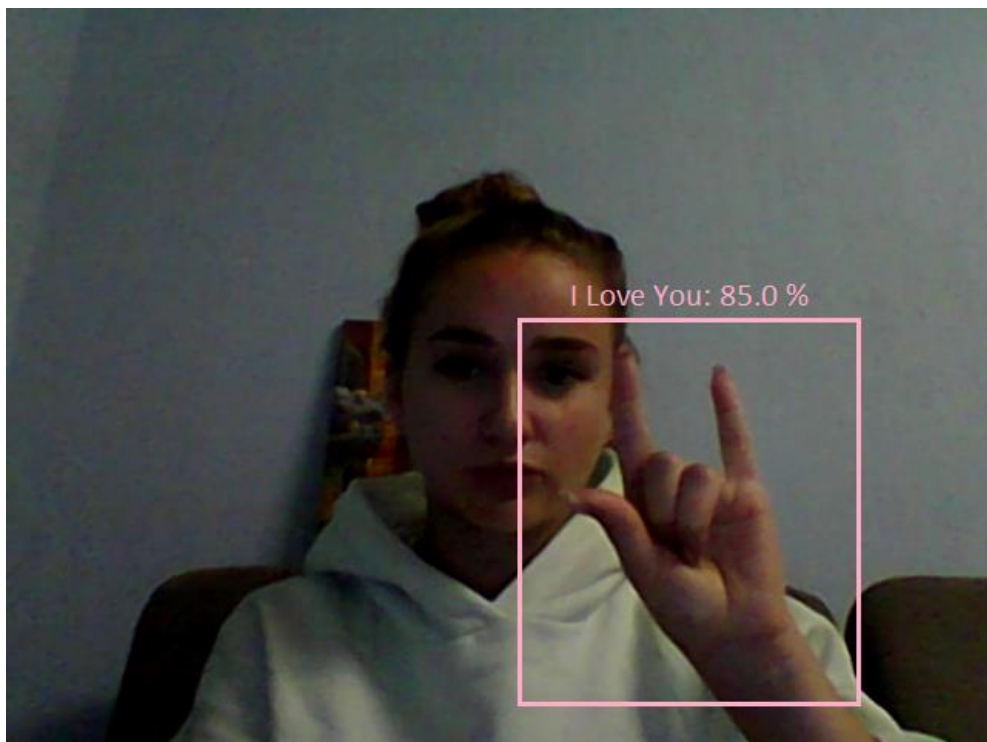


Рисунок 3.3 – Результат розпізнавання жесту «I Love You» № 1

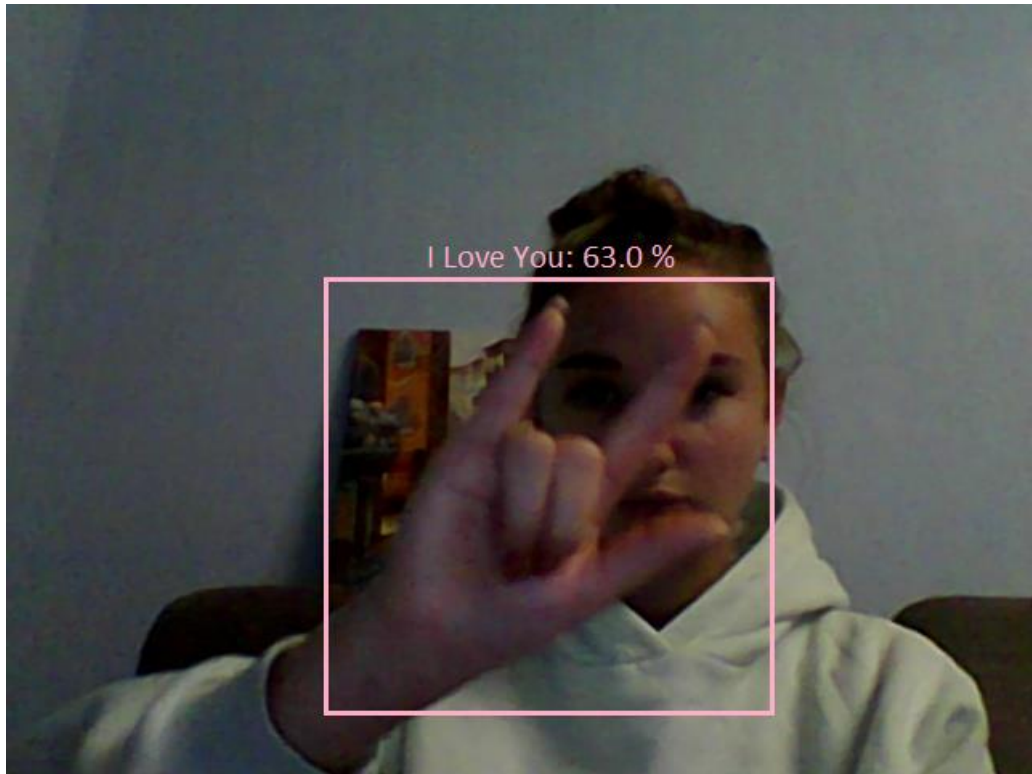


Рисунок 3.4 – Результат розпізнавання жесту «I Love You» № 2

На рисунках 3.5 та 3.6 представлено результат розпізнавання жесту «Thank You».



Рисунок 3.5 – Результат розпізнавання жесту «Thank You» № 1

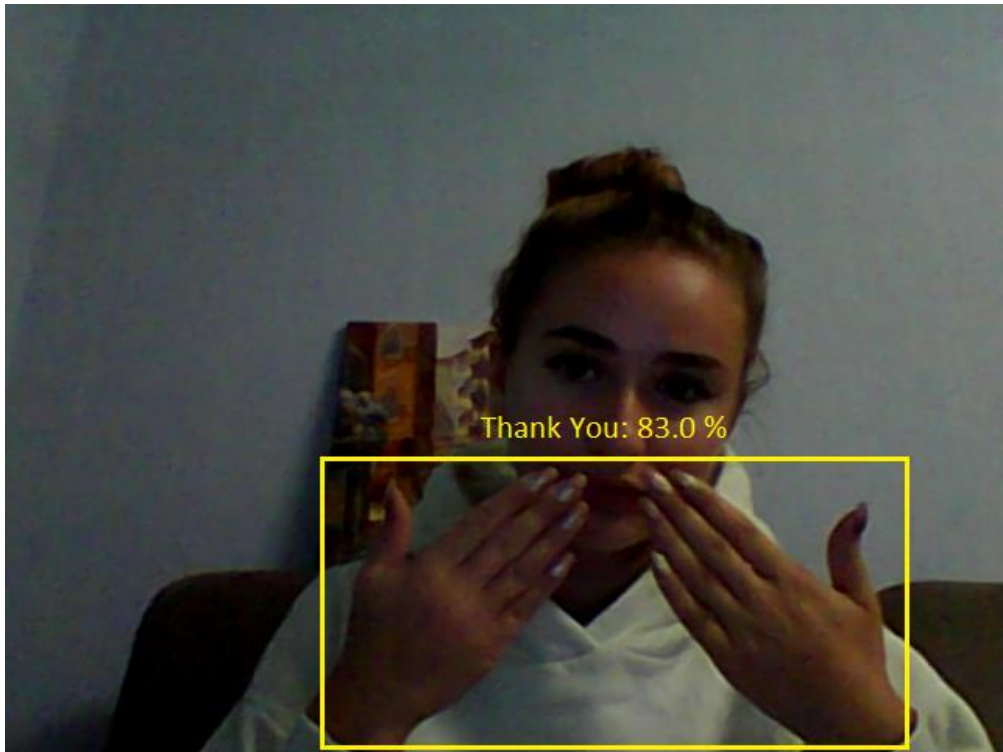


Рисунок 3.6 – Результат розпізнавання жесту «Thank You» № 2

На рисунках 3.7 та 3.8 представлено результат розпізнавання жесту «No».

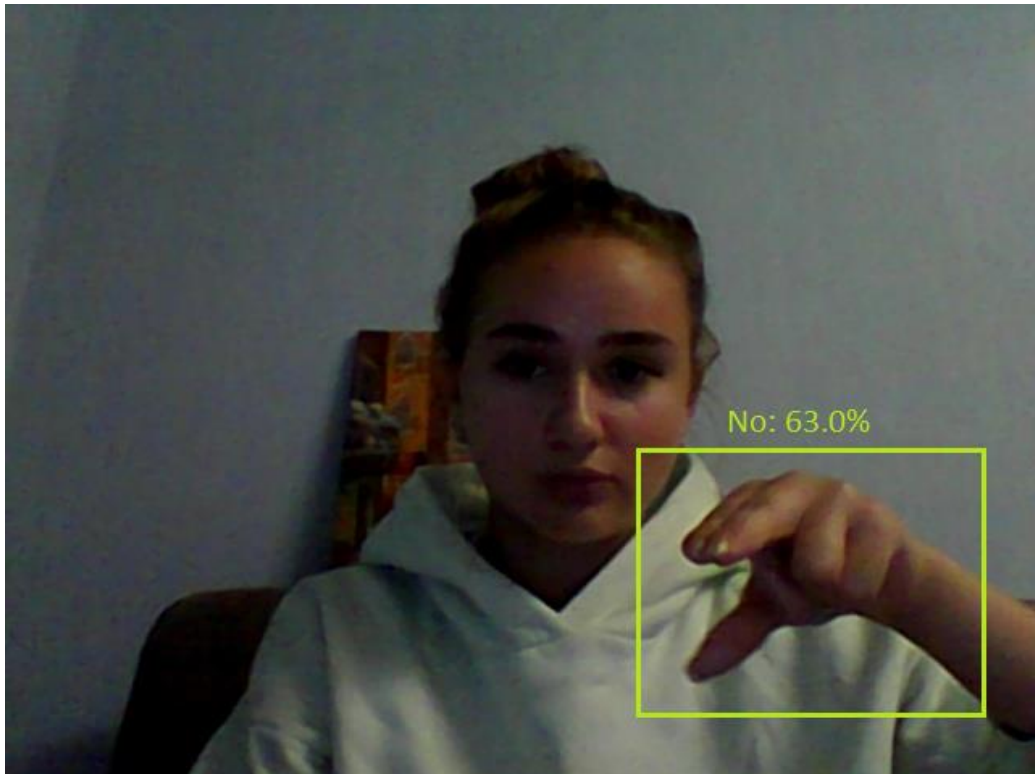


Рисунок 3.7 – Результат розпізнавання жесту «No» № 1

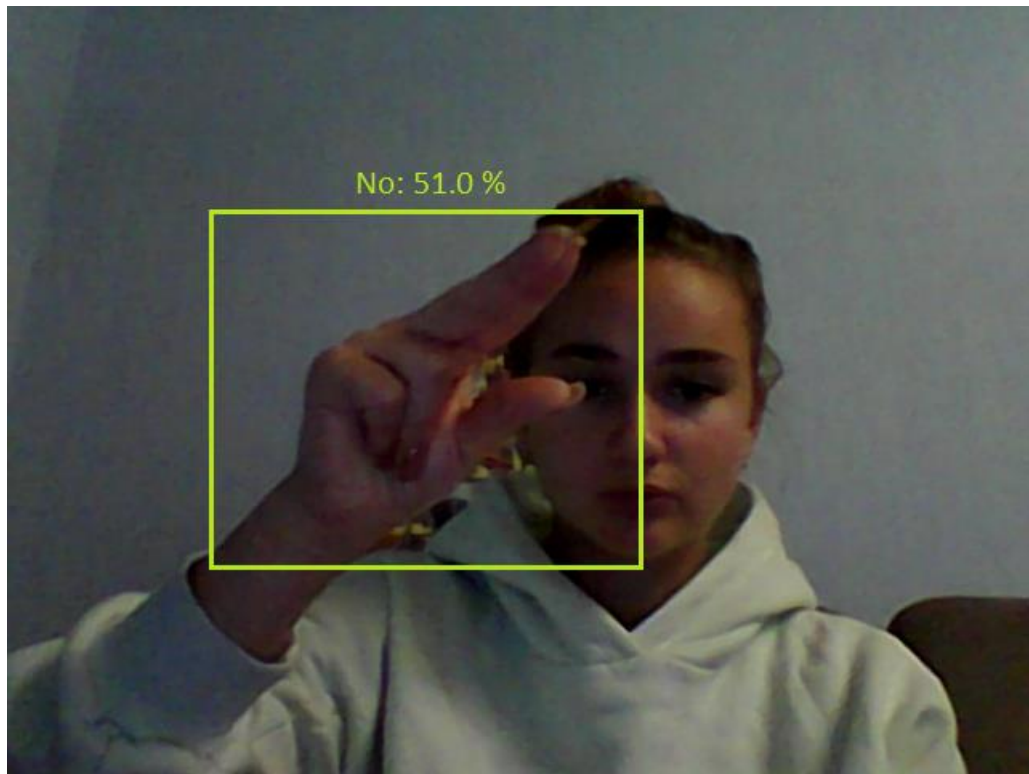


Рисунок 3.8 – Результат розпізнавання жесту «No» № 2

На рисунках 3.9 та 3.10 представлено результат розпізнавання жесту «Yes».

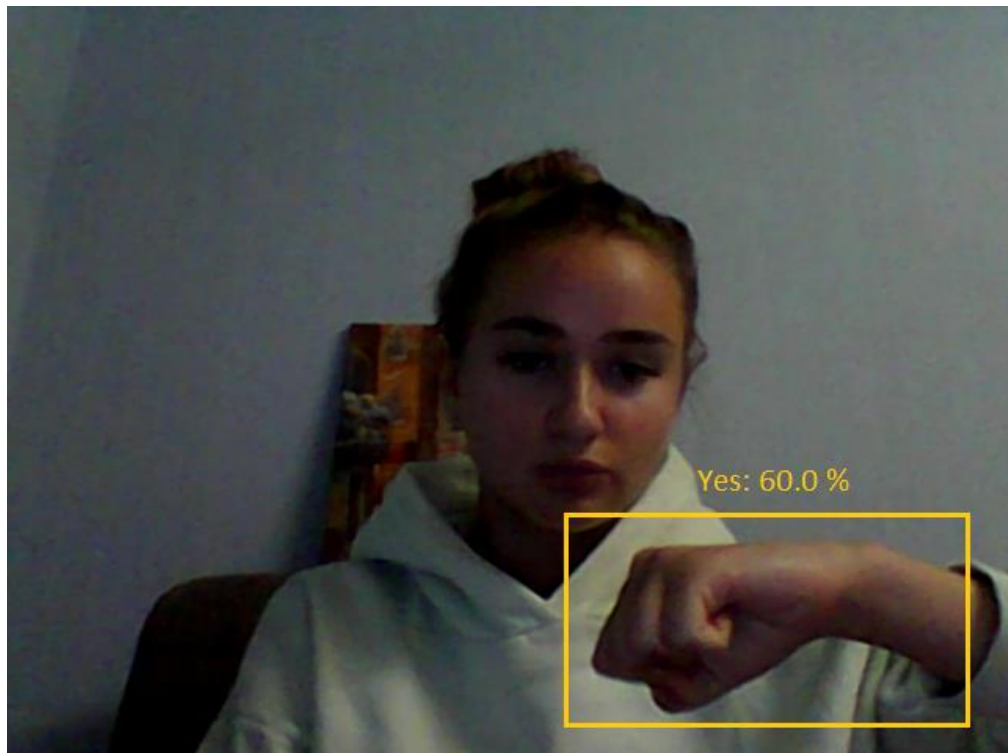


Рисунок 3.9 – Результат розпізнавання жесту «Yes» № 1

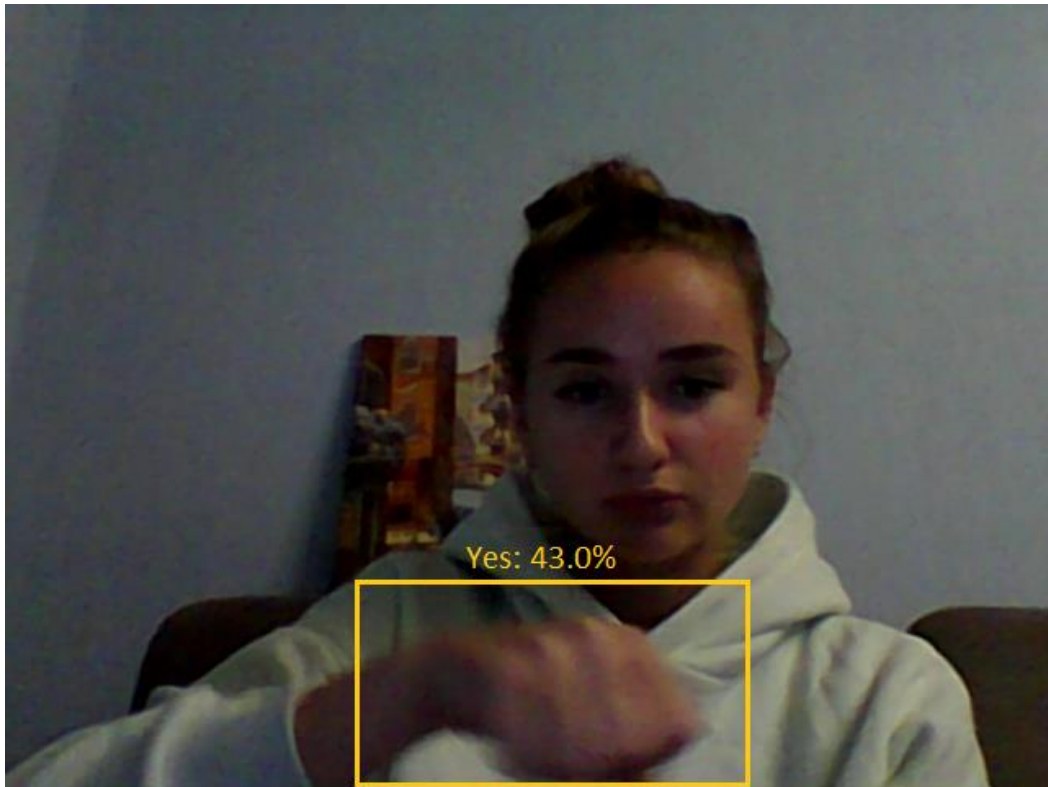


Рисунок 3.10 – Результат розпізнавання жесту «Yes» № 2

На рисунку 3.11 представлено результат розпізнавання жесту «Help».

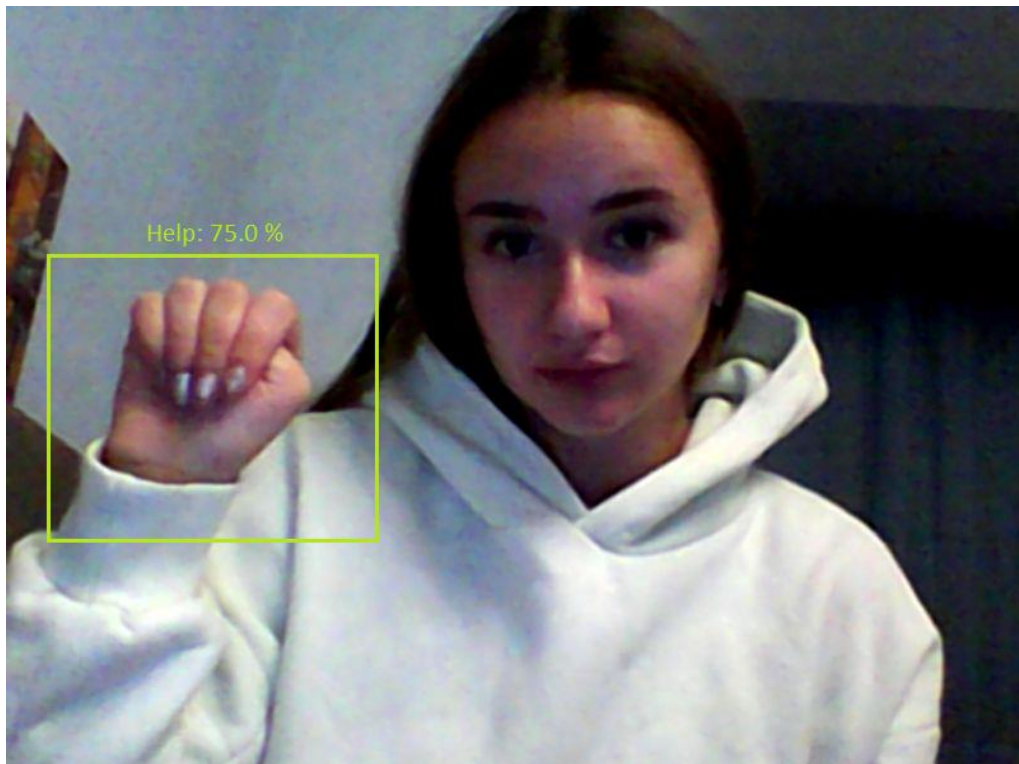


Рисунок 3.11 – Результат розпізнавання жесту «Help»

Також було проведено тестування програми шляхом показу жесту, якого немає у базі (рис. 3.12).

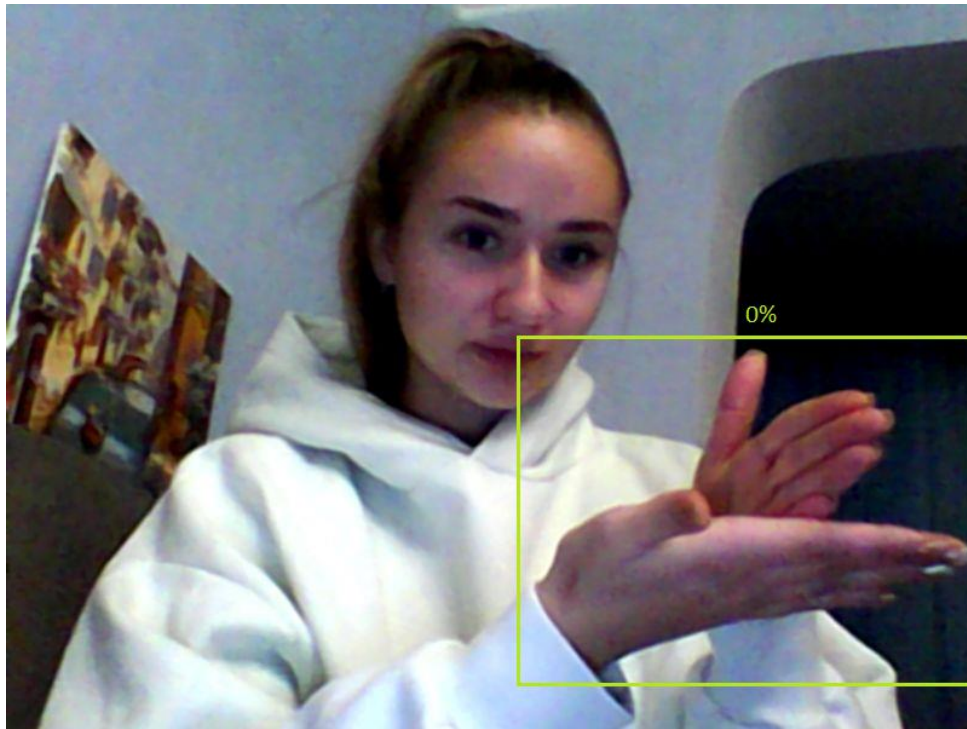


Рисунок 3.12 – Результат розпізнавання жесту, якого немає в базі

Згідно отриманих результатів тестування, можна зробити висновок, що програма є робочою та може використовуватись для безбар'єрного розуміння глухонімих людей або автоматичного фіксування жестів, які «повідомляють» про небезпечну життєву ситуацію чи загрозу життю людини.

У ході виконання кваліфікаційної роботи продемонстровано результати експерименту щодо 7 жестів, загалом у базі є 30 зразків жестів рук, а саме:

- «допоможіть»;
- «привіт»;
- «дякую»;
- «я тебе кохаю»;
- «так»;
- «ні»;
- «ім'я»;

- «я»;
- «необхідно»;
- «приємно»;
- «також»;
- «вчитись»;
- «працювати»;
- «шлюб»;
- «діти»;
- «скільки»;
- «вірити»;
- «сумніватись»;
- «не вірити»;
- «любити»;
- «читати»;
- «ніколи»;
- «важко»;
- «дім»;
- «добре»;
- «свідок»;
- «зараз»;
- «знайомитись»;
- «питати»;
- «чекати»;

### 3.4 Перспективи подальшої роботи

Розроблений програмний застосунок можливо запустити, маючи доступ до інтернету, ноутбука або комп'ютера з налагодженою відеокамерою.

Враховуючи ці фактори, під час удосконаленні даної розробки необхідно передбачити варіант створення мобільної або прикладної платформи, яка не потребуватиме інтернет-зв'язок, а також при більш детальному вивченні азбуки жестів – розширити колекцію жестів.

Дана розробка зумовлена не тільки, щоб зробити комфортним спілкування між глухонімими людьми та з глухонімими людьми, але й як інструмент автоматичного фіксування прохання про допомогу будь-якої людини, яка потрапила у кадр відеореєстратора.

У перспективі можна додати функцію виклику карети швидкої допомоги або поліції при ідентифікації жесту руки про необхідність надати допомогу.

## ВИСНОВКИ

Метою кваліфікаційної роботи була реалізація та вивчення методу розпізнавання жестової мови для безбар'єрного розуміння людей з вадами слуху або автоматичного фіксування спеціальних жестів. Адже, у різних сферах діяльності кожного дня виникає потреба спілкування між людьми, іноді це може викликати труднощі. Наприклад, при спілкуванні людей з людьми з вадами слуху.

Для того, щоб була можливість більш наочно продемонструвати процес розпізнавання жестів рук, прийнято рішення розробити програмний застосунок, де в режимі реального часу на базі створеної колекції жестів, можливо розпізнати жести рук шляхом відображення назви жесту на екрані та відсотку точності розпізнавання.

У ході тестування найвищий відсоток розпізнавання зафіксовано 95%, найменший – 43%. На отримання більш високого відсотку розпізнавання впливають такі фактори: гарне освітлення, якість зображення, якість подання жесту, складність форми самого жесту.

На основі визначеної постановки задачі, обґрунтовано вибір інструментів для програмної реалізації та реалізовано програмний код на платформі Jupyter Notebook.

У повному обсязі проведено аналіз поставленої задачі, обрано метод розпізнавання жестів рук. Даний спосіб вирішення задачі можливо впроваджувати у різних галузях, тим самим допомогти створити більше робочих місць для людей з вадами слуху.

Результати роботи апробовано у вигляді 2 тез доповідей під час Міжнародних науково-практичних конференцій «TRENDS IN THE SCIENTIFIC DEVELOPMENT» [72] та «INNOVATIVE TECHNOLOGIES IN SCIENCE AND PRACTICE» [73].

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Kobylin O., Gorokhovatskyi V., Tvoroshenko I., and Peredrii O. (2020) The application of non-parametric statistics methods in image classifiers based on structural description components, *Telecommunications and Radio Engineering*, 79(10), pp. 855-863.
2. Gorokhovatskyi V.O., Tvoroshenko I.S., and Vlasenko N.V. (2020) Using fuzzy clustering in structural methods of image classification, *Telecommunications and Radio Engineering*, 79(9), pp. 781-791.
3. Tvoroshenko I.S., and Gorokhovatsky V.O. (2020) Effective tuning of membership function parameters in fuzzy systems based on multi-valued interval logic, *Telecommunications and Radio Engineering*, 79(2), pp. 149-163.
4. Tvoroshenko I.S., and Gorokhovatsky V.O. (2019) Modification of the branch and bound method to determine the extremes of membership functions in fuzzy intelligent systems, *Telecommunications and Radio Engineering*, 78(20), pp. 1857-1868.
5. Gorokhovatskyi V., Rusakova N., and Tvoroshenko I. (2020) The application of image analysis methods and predicate logic in applied problems of magnetic monitoring, *Telecommunications and Radio Engineering*, 79(20), pp. 1801-1811.
6. Fang, G., Gao, W., & Zhao, D. (2006). Large-vocabulary continuous sign language recognition based on transition-movement models. *IEEE transactions on systems, man, and cybernetics-part a: systems and humans*, 37(1), 1-9.
7. Yao, G., Yao, H., Liu, X., & Jiang, F. (2006, August). Real time large vocabulary continuous sign language recognition based on OP/Viterbi algorithm. In *18th International Conference on Pattern Recognition (ICPR'06)* (Vol. 3, pp. 312-315). IEEE.

8. Vogler, C., & Metaxas, D. (2001). A framework for recognizing the simultaneous aspects of American Sign Language. *Computer Vision and Image Understanding*, 81(3), 358-384.
9. Wang, S.B., Quattoni, A., Morency, L.P., Demirdjian, D., & Darrell, T. (2006, June). Hidden conditional random fields for gesture recognition. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '06)* (Vol. 2, pp. 1521-1527). IEEE.
10. Ong, S.C., & Ranganath, S. (2005). Automatic sign language analysis: A survey and the future beyond lexical meaning. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 27(06), 873-891.
11. Dreuw, P., Steingrube, P., Deselaers, T., & Ney, H. (2009, June). Smoothed disparity maps for continuous american sign language recognition. In *Iberian Conference on Pattern Recognition and Image Analysis* (pp. 24-31). Springer, Berlin, Heidelberg.
12. Форсайт, Д., & Понс, Ж. (2004). Компьютерное зрение. Современный подход. М.: Вильямс, 928, 22.
13. Кобилін О.А., Творошенко І.С. (2021) Методи цифрової обробки зображень: навч. посібник. Харків: ХНУРЕ, 124 с.
14. Flah P. (2015) *Machine learning. The science and art of building algorithms that extract knowledge from data*, Moscow, Russia: DMK Press, 400 p.
15. Nong Ye. (2013) *Data Mining: Theories, Algorithms, and Examples*, Florida, USA: CRC Press, 349 p.
16. Asaad Ma. Babker, Abd Elgadir A. Altoum, Irina Tvoroshenko, and Vyacheslav Lyashenko (2019) Information Technologies of the Processing of the Spaces of the States of a Complex Biophysical Object in the Intellectual Medical System HEALTH, *International Journal of Advanced Trends in Computer Science and Engineering*, 8(6), pp. 3221-3227.
17. Yousef Ibrahim Daradkeh, and Iryna Tvoroshenko (2020) Application of an Improved Formal Model of the Hybrid Development of Ontologies in Complex Information Systems, *Applied Sciences*, 10(19), p. 6777.

18. Barandas, M., Gamboa, H., & Fonseca, J.M. (2015). A real time biofeedback system using visual user interface for physical rehabilitation. *Procedia Manufacturing*, 3, 823-828.
19. Freeman, W.T., Anderson, D.B., Beardsley, P., Dodge, C.N., Roth, M., Weissman, C.D., ... & Tanaka, K.I. (1998). Computer vision for interactive computer graphics. *IEEE Computer Graphics and Applications*, 18(3), 42-53.
20. Shin, D.H., Hwang, Y., & Choo, H. (2013). Smart TV: are they really smart in interacting with people? Understanding the interactivity of Korean Smart TV. *Behaviour & information technology*, 32(2), 156-172.
21. Davis, J.W., & Bobick, A. (1998, November). Virtual PAT: a virtual personal aerobics trainer. In *Workshop on Perceptual User Interfaces* (pp. 13-18).
22. Davis, J.W., & Bobick, A. F. (1998, November). A robust human-silhouette extraction technique for interactive virtual environments. In *International Workshop on Capture Techniques for Virtual Environments* (pp. 12-25). Springer, Berlin, Heidelberg.
23. Bobick, A., & Davis, J. (2001). The representation and recognition of action using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3), 257-267.
24. Bobick, A.F., Intille, S.S., Davis, J.W., Baird, F., Pinhanez, C.S., Campbell, L.W., ... & Wilson, A. (1999). The KidsRoom: A perceptually-based interactive and immersive story environment. *Presence*, 8(4), 369-393.
25. Freeman, W.T., & Roth, M. (1995, June). Orientation histograms for hand gesture recognition. In *International workshop on automatic face and gesture recognition* (Vol. 12, pp. 296-301). IEEE Computer Society, Washington, DC.
26. Zhou, H., Lin, D.J., & Huang, T.S. (2004, June). Static hand gesture recognition based on local orientation histogram feature distribution model. In *2004 Conference on computer vision and pattern recognition workshop* (pp. 161-161). IEEE.
27. Lathuiliere, F., & Herve, J.Y. (2000, September). Visual tracking of hand posture with occlusion handling. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000* (Vol. 3, pp. 1129-1133). IEEE.

28. Wang, R.Y., & Popović, J. (2009). Real-time hand-tracking with a color glove. *ACM transactions on graphics (TOG)*, 28(3), 1-8.
29. Theobalt, C., Albrecht, I., Haber, J., Magnor, M., & Seidel, H.P. (2004). Pitching a baseball: tracking high-speed motion with multi-exposure images. In *ACM SIGGRAPH 2004 Papers* (pp. 540-547).
30. Dorner, B. (1994). *Chasing the colour glove: Visual hand tracking* (Doctoral dissertation, Theses (School of Computing Science)/Simon Fraser University).
31. M. Ayaz Ahmad, Volodymyr Gorokhovatskyi, Iryna Tvoroshenko, Nataliia Vlasenko, Syed Khalid Mustafa (2021) The Research of Image Classification Methods Based on the Introducing Cluster Representation Parameters for the Structural Description, *International Journal of Engineering Trends and Technology*, 69(10), pp. 186-192.
32. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., Gadetska S., and Al-Dhaifallah M. (2021) Methods of Classification of Images on the Basis of the Values of Statistical Distributions for the Composition of Structural Description Components, *IEEE Access*, 9, pp. 92964-92973.
33. Зайцева, Г.Л. (2004). *Жестовая речь. Дактилология*. Гуманитар. изд. центр ВЛАДОС.
34. Огурцова, Н., & Сутырина, М. (2021). *Теория и практика сурдоперевода. Дактилология*. Litres.
35. Tvoroshenko I., and Tkachenko D. (2020) Mechanisms of image classification based on descriptors of local features, *Abstracts of IV International Scientific and Practical Conference «Integration of scientific bases into practice»* (October 12-16, 2020). Stockholm, Sweden, pp. 443-448.
36. Lamberti, L., & Camastra, F. (2011, September). Real-time hand gesture recognition using a color glove. In *International Conference on Image Analysis and Processing* (pp. 365-373). Springer, Berlin, Heidelberg.
37. Lathuiliere, F., & Herve, J.Y. (2000, September). Visual tracking of hand posture with occlusion handling. In *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000* (Vol. 3, pp. 1129-1133). IEEE.

38. Peters J.F. (2017) *Foundations of computer vision: Computational Geometry, Visual Image Structures and Object Shape Detection*, Cham, Switzerland: Springer International Publisher, 417 p.

39. Творошенко І.С. (2021) Технології прийняття рішень в інформаційних системах: навч. посібник. Харків: ХНУРЕ, 120 с.

40. Всесвітня організація здоров'я. URL: <https://www.who.int/ru> (дата звернення 10.09.2021).

41. M. Ayaz Ahmad, Irina Tvoroshenko, Jalal Hasan Baker, Liubov Kochura, Vyacheslav Lyashenko (2020) Interactive Geoinformation Three-Dimensional Model of a Landscape Park Using Geoinformatics Tools, *International Journal on Advanced Science, Engineering and Information Technology*, 10(5), pp. 2005-2013.

42. Randles, B.M., Pasquetto, I.V., Golshan, M.S., & Borgman, C.L. (2017, June). Using the Jupyter notebook as a tool for open science: An empirical study. In *2017 ACM/IEEE Joint Conference on Digital Libraries (JCDL)* (pp. 1-2). IEEE.

43. Tvoroshenko I., and Dziubenko M. (2020) Modern methods of analysis of the movement scheme using video detection of vehicles, *Abstracts of V International Scientific and Practical Conference «Study of modern problems of civilization»* (October 19-23, 2020). Oslo, Norway, pp. 422-428.

44. Pansare, J.R., Gawande, S.H., & Ingle, M. (2012). Real-time static hand gesture recognition for American Sign Language (ASL) in complex background.

45. Majidi, C. (2014). Soft robotics: a perspective-current trends and prospects for the future. *Soft robotics*, 1(1), 5-11.

46. Fels, S.S., & Hinton, G.E. (1993). Glove-talk: A neural network interface between a data-glove and a speech synthesizer. *IEEE transactions on Neural Networks*, 4(1), 2-8.

47. Digital life: Розмовляюча рукавичка. URL: [https://zoom.cnews.ru/rnd/article/item/digital\\_life\\_govoryashchaya\\_perchatka\\_noubukmutant\\_i\\_robotygladiatory](https://zoom.cnews.ru/rnd/article/item/digital_life_govoryashchaya_perchatka_noubukmutant_i_robotygladiatory) (дата звернення: 15.09.2021).

48. Starner, T., Weaver, J., & Pentland, A. (1998). Real-time american sign language recognition using desk and wearable computer based video. *IEEE Transactions on pattern analysis and machine intelligence*, 20(12), 1371-1375.

49. Kinect Sign Language Translator. URL: <https://www.microsoft.com/en-us/research/blog/kinect-sign-language-translator-part-1/> (дата звернення: 11.10.2021).

50. Творошенко, І.С. (2018). Особливості застосування сучасних принципів штучного інтелекту до розробки ефективних механізмів моделювання складних систем. *Science and Technology of the Present Time: Priority Development Directions of Ukraine and Poland*, 118-121.

51. Tvoroshenko, I. (2019). Development of models of spatial analysis of status of interactive processes of complex systems.

52. Кучеренко Е.И., Филатов В.А., Творошенко И.С., Байдан Р.Н. (2005) Интеллектуальные технологии в задачах принятия решений технологических комплексов на основе нечеткой интервальной логики, *Восточно-Европейский журнал передовых технологий*, № 2. С. 92-96.

53. Творошенко, І.С., Мгеброва, В.Р., & Білий, В.В. (2016). Практичні аспекти створення вихідної інформації для проведення геоінформаційного аналізу у сфері управління нерухомістю.

54. Кучеренко, Е.И., Творошенко, И.С. (2010) Прикладные аспекты моделирования нечетких процессов в сложных системах. *Збірник наукових праць Харківського університету Повітряних сил*, (1), 127-131.

55. Творошенко, И.С., Дехтярь, А.П. (2005, June) Информационные технологии в задачах компьютерной диагностики с использованием интеллектуальных систем. In *Клиническая информатика и Телемедицина. Компьютерная Медицина–2005: материалы междунар. научн.-технич. конф., Харьков* (p. 138).

56. SignAll (2021). URL: <http://signall.us> (дата звернення: 23.10.2021).

57. MotionSavvy (2021). URL: <http://motionsavvy.com> (дата звернення: 25.10.2021).

58. Johnson, J.L., & Padgett, M.L. (1999). PCNN models and applications. *IEEE transactions on neural networks*, 10(3), 480-498.
59. Нагапетян, В.Э. (2013). *Методы распознавания жестов руки на основе анализа дальностных изображений* (Doctoral dissertation, Диссертация на соискание ученой степени кандидата физико-математических наук, РУДН).
60. Славенко, С.Б., & Сімкін, О.І. (2020). Діаграма Вороного та її застосування.
61. Manjrekar, M.D., Steimer, P.K., & Lipo, T.A. (2000). Hybrid multilevel power conversion system: A competitive solution for high-power applications. *IEEE transactions on industry applications*, 36(3), 834-841.
62. Олексій Приходько. URL: <https://incrussia.ru/news/perevod-ryaj/> (дата звернення 25.10.2021).
63. Тихонов, А.Н., Гончарский, А.В., Степанов, В.В., & Ягола, А.Г. (1990). Численные методы решения некорректных задач.
64. Домахина, Л.Г. (2009). Регуляризация скелета для задачи сравнения формы. *Математические методы распознавания образов*, 14(1), 342-345.
65. Батищев, Д.С., Михелев, В.М., & Утянский, А.А. (2020). Метод сегментации перекрывающихся форменных элементов крови на микроскопических медицинских изображениях. *Экономика. Информатика*, 47(4).
66. Чуй, Ю.В., & Мартинюк, П.М. (2016). Дослідження впливу тонких напівпроникних включень на процеси фільтраційної консолідації ґрунтів в двовимірному випадку. *Математичне та комп'ютерне моделювання. Серія: Технічні науки*, (13), 191-201.
67. Бешенкова, Е.В. (2008). Вариативность в орфографии. *Вестник Тамбовского университета. Серия: Гуманитарные науки*, (2).
68. Bai, X., Latecki, L.J., & Liu, W.Y. (2007). Skeleton pruning by contour partitioning with discrete curve evolution. *IEEE transactions on pattern analysis and machine intelligence*, 29(3), 449-462.

69. Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., ... & Moore, R. (2013). Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1), 116-124.

70. Urain, M.K. (2009). La influencia de Aristóteles en la historia de la lengua de signos. In *Tendencias actuales en la investigación diacrónica de la lengua: actas del VIII Congreso nacional de la asociación de jóvenes investigadores de historiografía e historia de la lengua española (AJIHLE), Barcelona, del 2 al 4 de abril de 2008* (p. 361). Edicions Universitat Barcelona.

71. Van Rossum, G., & Drake Jr, F.L. (1995). *Python tutorial* (Vol. 620). Amsterdam: Centrum voor Wiskunde en Informatica.

72. Tvoroshenko I., and Kharchenko A. (2021) Some aspects of modern development for sign language recognition systems, *Abstracts of II International scientific and practical conference «Trends in the scientific development» (September 28 – October 01, 2021). Vancouver, Canada*, pp. 349-354.

73. Kharchenko A. (2021) Features of development of sign language recognition methods, *Abstracts of VI International Science Conference «Innovative technologies in science and practice» (October 26 – 28, 2021). Haifa, Israel*, pp. 418-421.