

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 перший (бакалаврський)

**РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАСТОСУНКУ ДЛЯ УПРАВЛІННЯ**  
**КУРСОРОМ ЗА ДОПОМОГОЮ ЖЕСТІВ РУК**  
(тема)

Виконав:  
студент 4 курсу, групи ІТІНФ-18-2

Тарасов Д.К.  
(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)

2022 р.

## Харківський національний університет радіоелектроніки

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

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

« \_\_\_\_ » \_\_\_\_\_ 2022 р.

**ЗАВДАННЯ**  
НА КВАЛІФІКАЦІЙНУ РОБОТУстудентові Тарасову Данилу Кириловичу  
(прізвище, ім'я, по батькові)1. Тема роботи Розроблення програмного застосунку для управління курсором за допомогою жестів рук

затверджена наказом університету від 16 травня 2022 року № 541Ст

2. Термін подання студентом роботи до екзаменаційної комісії 28 травня 2022 р.

3. Вихідні дані до роботи науково-методична та науково-технічна література, матеріали конференцій, дані інтернет-мережі, бібліотека комп'ютерного зору з відкритим кодом OpenCV, мова програмування Python, середовище розробки Microsoft Visual Studio Code.

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

1. Аналіз існуючих програмних застосунків управління курсором за допомогою жестів рук.

2. Моделювання структури програмного застосунку управління курсором за допомогою жестів рук.

3. Розроблення програмного застосунку для управління курсором за допомогою жестів рук.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри) Актуальність проблеми, постановка задачі, етапи розроблення застосунку, тестування застосунку, висновки, 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 (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	18.04.2022	
2	Аналіз завдання, підбір літератури	18.04.22-21.04.22	
3	Аналіз літератури з досліджуваної проблеми	22.04.22-25.04.22	
4	Аналіз технічних засобів	26.04.22-30.04.22	
5	Розробка методу	01.05.22-14.05.22	
6	Програмна реалізація	15.05.22-23.05.22	
7	Оформлення пояснювальної записки	24.05.22-26.05.22	
8	Перевірка на плагіат	27.05.22	
9	Рецензування	28.05.22	
10	Підготовка презентації та доповіді	29.05.22-30.05.22	
11	Занесення роботи в електронний архів	31.05.22	
12	Попередній захист кваліфікаційної роботи	06.06.22	

Дата видачі завдання 18 квітня 2022 р.

Студент \_\_\_\_\_

(підпис)

Керівник роботи \_\_\_\_\_  
(підпис)

доц. Творошенко І.С.  
(посада, прізвище, ініціали)

## РЕФЕРАТ/ABSTRACT

Пояснювальна записка до кваліфікаційної роботи: 56 с., 25 рис., 1 дод., 42 джерела.

КЕРУВАННЯ КУРСОРОМ, РОЗПІЗНАВАННЯ ЖЕСТИВ РУК, НЕЙРОННА МЕРЕЖА, OPEN CV, КОМП'ЮТЕРНА МИША, ВІРТУАЛЬНА МИША, ВЗАЄМОДІЯ ЛЮДИНИ ТА КОМП'ЮТЕРА.

Об'єктом роботи є послідовність різноракурсних жестів рук.

Метою роботи є розроблення програмного застосунку для керування курсором за допомогою жестів рук.

Використано методи машинного навчання для розпізнавання та помічення кінцівок рук та їх жестів. Проведено дослідження методів розпізнавання рук та їх жестів. Досліджено методи розробки нейронної мережі для детектування рук та алгоритм керування курсором.

У результаті роботи здійснена програмна реалізація системи керування мишею за допомогою рук.

CURSOR CONTROL, HAND GESTURE RECOGNITION, NEURAL NETWORK, OPEN CV, COMPUTER MOUSE, VIRTUAL MOUSE, HUMAN AND COMPUTER INTERACTION.

The object of the work is a sequence of different hand gestures.

The aim of the work is to develop a software application for cursor control using hand gestures.

Machine learning methods are used to recognize and notice hand limbs and their gestures. A study of methods for recognizing hands and their gestures. Methods of neural network development for hand detection and cursor control algorithm are studied.

As a result of work the software implementation of the mouse control system by means of hands is carried out.

## ЗМІСТ

Перелік умовних позначень, символів, одиниць, скорочень і термінів.....	7
Вступ.....	8
1 Аналіз існуючих програмних застосунків управління курсором за допомогою жестів рук .....	9
1.1 Сучасний стан розвитку програмних застосунків управління курсором за допомогою жестів рук в Україні та за кордоном.....	9
1.1.1 Використання технологій керування жестами у автопромисловості.....	11
1.1.2 Використання технологій керування жестами у медицині...	12
1.1.3 Використання технологій керування жестами у військовій промисловості.....	15
1.1.4 Використання технологій керування жестами у віртуальній та доповненій реальності.....	17
1.2 Аналіз літературних джерел щодо існуючих підходів розроблення програмних застосунків управління курсором за допомогою жестів рук .....	19
1.3 Постановка задачі.....	21
2 Моделювання структури програмних застосунків управління курсором за допомогою жестів рук.....	22
2.1 Застосування методів системного аналізу до предметної області «Управління курсором за допомогою жестів рук».....	22
2.1.1 Розпізнавання на основі кольору за допомогою маркера для рукавичок.....	23
2.1.2 Розпізнавання на основі руху.....	24
2.1.3 Розпізнавання на основі скелета.....	26
2.2 Особливості нейронної мережі R-CNN для управління курсором за допомогою жестів рук.....	27

	6
2.2.1 R-CNN.....	28
2.2.2 Fast R-CNN.....	29
2.2.3 Faster CNN.....	31
2.3 Моделювання структури та наповнення програмного застосунку для управління курсором за допомогою жестів рук.....	32
3 Розроблення програмного застосунку для управління курсором за допомогою жестів рук .....	35
3.1 Вибір інструментальних засобів для реалізації поставленої задачі	35
3.2 Етапи розроблення програмного застосунку для управління курсором за допомогою жестів рук.....	37
3.3 Тестування реалізованого програмного застосунку та аналіз результатів.....	41
3.4 Перспективи подальшої роботи.....	46
Висновки.....	48
Перелік джерел посилання.....	49
Додаток А Фото процесу розпізнавання жестів.....	54

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

МН – машинне навчання

ШІ – штучний інтелект

ADAS – Advanced Driver Assistance Systems

AGRS – Automotive Gesture Recognition System

VR – Virtual Reality

AR – Augmented Reality

MX – Mixed Reality

CNN – Convolutional Neural Network

VSC – Visual Studio Code

HCI – Human and Computer Interaction

ToF – Time of Flight

## ВСТУП

Жестами вважаються рухи тіла людини, які зазвичай означають міміку обличчя людини або жест рук та пальців. Розпізнавання жестів дозволить людині мати зв'язок з комп'ютером без всяких маніпуляторів-посередників, як миші та клавіатури [1]. За допомогою таких технологій можна управляти пристроями, такими як екрани, телевізори, відеоігри, перемикати канали на телевізорі, змінювати гучність на комп'ютері за допомогою рухів кінцівок рук та жестів. Розпізнавання жестів допомагають технічним пристроями бути більш доступними людям з різними фізичними вадами та особливостями. Старомодні способи введення інформації, такі як текстовий та графічні інтерфейси все ще обмежуються маніпуляторами, як миша та клавіатура. Технології розпізнавання роблять такі пристрої, як миша та клавіатури, сенсорні екрани та ігрові маніпулятори, непотрібними та застарілими.

Під час Covid-19 багато людей стали пильними та обачними з поверхнями, до яких вони дотикаються. Миття рук, дезінфекція поверхні, використання санітайзерів дали зрозуміти, що спілкуватися з людьми на відстані та користування громадськими пристроями, як системи оплати карток та банкомати на відстані може зберегти життя. Багато рішень стали пропонуватися людям для полегшення контакту людей та пристроїв [2].

Розпізнавання жестів рук – це технологія, яка дозволяє аналізувати рухи кінцівок та виконувати різні маніпулятивні дії [3]. Такі технології дозволяють спростити життя людей та збільшити можливості юзабіліті людини та техніки.

Технології потрібні для полегшення комунікації людини з машиною та можуть бути використані в розважальних та навчальних цілях.

# 1 АНАЛІЗ ІСНУЮЧИХ ПРОГРАМНИХ ЗАСТОСУНКІВ УПРАВЛІННЯ КУРСОРОМ ЗА ДОПОМОГОЮ ЖЕСТІВ РУК

1.1 Сучасний стан розвитку програмних застосунків управління курсором за допомогою жестів рук в Україні та за кордоном

Людина може обмінюватися різними видами інформації на різних рівнях його розуміння, вербальних чи невербальних. В процесі діалогу важливу роль відіграють манери та жести партнерів. Психологи зазначають, що 60 відсотків нашої комунікації передається невербальними методами, інші 40 – вербально. Тому при створенні систем взаємодій людини та комп'ютера є важливим використання методів розпізнання жестів [4].

Щоб вдало розпізнавати жести та їх класифікувати, потрібно вміти аналізувати зорову інформацію. Використання технологій комп'ютерного зору допоможе автоматизувати цей процес та його вдосконалити за рахунок машинного навчання.

Комп'ютерне бачення є важливою сферою по розширенню можливостей комп'ютерів з ідентифікації та визначення об'єктів та людей на зображеннях та відео. Як і інші типи ШІ, комп'ютерний зір орієнтується на виконанні та автоматизації завдань, які імітують людські можливості. І тут комп'ютерне зір намагається імітувати зір та сприйняття людини. Спектр практичного застосування технологій комп'ютерного зору стало невід'ємним та центральним компонентом багато інновацій та бізнес-рішень [5].

Останні кілька років різко зріс попит на розробки методів машинного навчання для застосунків з урахуванням комп'ютерного зору. Машинне навчання та комп'ютерний зір доповнюють один одного. Комп'ютерний зір використовує методи машинного навчання для автоматизації отримання візуальних моделей, перетворення сигналів на символи, створення систем обробки зображень, що навчаються, і вивчення того, коли застосовувати якийсь алгоритм у системі технічного зору [6].

Алгоритми МН можуть використовуватися як мінімум двома способами [7]:

- для покращення сприйняття навколишнього середовища з метою ідентифікації та класифікації об'єктів;
- для зменшення різниці між внутрішніми уявленнями про навколишнє середовище та поданням знань, необхідних для вилучення відповідної інформації із зображень.

Системи розпізнавання жестів починають інтегрувати в системи дистанційного зв'язку. Zoom, популярне програмне забезпечення для відеодзвінків, нещодавно представило функцію розпізнавання жестів у своїй настільній версії. Функція розпізнавання жестів дозволить користувачам позначати смайли з піднятою рукою, коли ви піднімаєте руку, або показувати смайли з великим пальцем вгору, коли ви показуєте його на вебкамері. Окрім випадкових збоїв, функція розпізнавання жестів дозволяє Zoom подолати розрив між природним і цифровим спілкуванням.

Пошукова система Clearview AI для ідентифікації людей за обличчям та ходом зараз допомагає Україні потенційно перевіряти людей на контрольно-пропускних пунктах. Технологія дозволяє легше розкривати крадіїв, боротися з дезінформацією, пошук біженців, які розлучені зі своєю родиною та ідентифікувати загиблих, навіть якщо є пошкодження обличчя. Також сервіс допомагає уряду в розвінчанні неправдивих публікацій в соціальних мережах ,пов'язаних з війною.

Спеціальне програмне забезпечення для багатозонного датчика часу польоту (ToF) ST VL53L5CX забезпечує безконтактне керування жестами, зберігаючи при цьому конфіденційність. Поєднання датчика VL53L5CX і програмного забезпечення STSW-IMG035 робить виявлення жестів простішим і доступнішим для застосування в широкому діапазоні споживчих і промислових застосувань. Датчик ToF VL53L5CX забезпечує 64 зони з високою точністю в діапазоні до 400 см і широким полем зору 63° по діагоналі з квадратними краями [8].

Разом сенсор і програмне забезпечення обчислюють в режимі реального часу координати руки, що дозволяє відстежувати руки та розпізнавати такі жести, як натискання, прокручування та контроль рівня.

Автомобільна система розпізнавання жестів (AGRS) – це вбудований автомобільний пристрій, який інтерпретує команди людини у вигляді рухів рук або жестів. Ці системи доступні як у сенсорному, так і в безконтактному інтерфейсі користувача, і використовують входи для керування іншими автомобільними електронними пристроями, щоб зменшити відволікання водія під час водіння і таким чином підвищити безпеку та комфорт. Ці системи зазвичай використовуються для автомобільних, мультимедійних, інформаційно-розважальних та навігаційних пристроїв.

Європа була провідним регіональним ринком автомобільної системи розпізнавання жестів (AGRS) у 2020 році з найбільшим зростанням доходів. Факторами, які сприяють віковому доходу цього регіонального ринку, є високий дохід на душу населення, збільшення продажів розкішних та автономних автомобілів, швидка інтеграція автомобільних систем розпізнавання жестів у транспортні засоби та суворі державні правила щодо безпеки автомобілів. Іншим регіоном, який стимулює зростання європейського ринку, є деякі з провідних гравців галузі, включаючи Continental AG, SoftKinetic, NXP Semiconductors NV та Cognitec Systems.

### 1.1.1 Використання технологій керування жестами у автопромисловості

Управління декількома пристроями під час водіння краде увагу водіїв з дороги і стає причиною аварій у 1 з 3 випадків. Багато дослідницьких зусиль спрямовані на розробку, виробництво та тестування людсько-машинних інтерфейсів, які дозволяють керувати пристроями автомобіля, не відволікаючи уваги водіїв [1].

Автомобільна система розпізнавання жестів (AGRS) – це вбудований автомобільний пристрій, який інтерпретує команди людини у вигляді рухів рук або жестів. Ці системи доступні як у сенсорному, так і в безконтактному інтерфейсі користувача, і використовують входи для керування іншими автомобільними електронними пристроями, щоб зменшити відволікання водія під час водіння і таким чином підвищити безпеку та комфорт. Системи зазвичай використовуються для автомобільних, мультимедійних, інформаційно-розважальних та навігаційних пристроїв.

З 2016 року компанія BMW використовує в лінійці преміальних автомобілів мультимедійну та інтерактивну систему керування жестами. Авто дозволяє збільшувати та зменшувати гучність музики, приймати та відхиляти телефонний дзвінок та міняти нахил огляду камери.

Компанія LG Electronics поставляє для автомобілів Mercedes – Benz передову систему розпізнавання жестів, за допомогою якої водії можуть керувати функціями автомобіля. Камера, яка вмонтована в стелю дозволяє слідкувати за жестами водія, дозволяючи йому керувати круїз-контролем та гучністю звуку в аудіосистемі. LG планує розвинути можливості системи такими функціями, як перемикання швидкості коробки передач та датчиком, який стежить, щоб водій не задрімав за кермом. Деякі автовиробники оснащують автомобілі системами розпізнавання пішоходів та зчитування дорожніх знаків. Передові системи допомоги водієві (ADAS) – це складові елементи повністю безпілотних автомобілів, які в майбутньому повністю змінять автотранспорт і те, як на ньому їздять люди.

### 1.1.2 Використання технологій керування жестами у медицині

Розглядаючи інші прикладні соціальні сфери застосування комп'ютерного зору та технології розпізнавання жестів неможливо обминути медицину, зокрема, хірургію та лабораторні дослідження.

Часто такі сфери вимагають практично ідеальних умов до абсолютної біологічної стерильності приміщень. Для створення таких умов існують так звані «чисті приміщення», де в повітрі підтримуються в певному заданому діапазоні розмір і число таких частинок, як пил, мікроорганізми та хімічні пари. Такі приміщення, як правило, будуються і використовуються так, щоб звести до мінімуму надходження та накопичення таких частинок усередині них.

Для зменшення забруднення в чистих приміщеннях високого класу застосовують спеціальні системи вентиляції, при яких потік повітря рухається зверху вниз. При такому потоці повітря частки бруду від людей та обладнання не розлітаються по всьому приміщенню, а збираються потоком біля підлоги. Звідси впливають особливості підтримання та визначення показників чистоти, специфічні вимоги до контрольних приладів, лічильників частинок у повітрі та ін. Проте навіть у таких умовах присутність людини як біологічного організму створює труднощі у проведенні операції чи наукового експерименту.

Хоча персонал, як правило, одягнений у халати із синтетичного матеріалу, які легко очищаються та виділяють мінімальну кількість частинок, кількість щілин та кутів у чистих приміщеннях зведені до мінімуму, підлогові покриття на основі вінілу накладаються із зачіпанням на стіни, а джерела світла встановлюються у втопленому варіанті, тим не менш, це не вирішує проблему повністю. Приблизно 70% пилу має природне походження, решта 30% є відходами життєдіяльності людини.

Єдино можливим вирішенням даної проблеми може бути повне «усунення» людини як біологічного організму від участі в дослідженні або хірургічної операції. Однак існуючі рішення рідко дозволяють здійснювати досить точні маніпуляції з інструментами, що використовуються в ході операції або експерименту, на відстані.

Альтернативним вирішенням проблеми буде повна імітація рухів рук хірурга чи лаборанта.

Здійснити це можна при використанні двох складових: біометричних роботизованих рук, що повністю імітують рух кистей рук і пальців, та віртуального асистента на базі програмного фреймворку здатного точно розпізнавати та інтерпретувати рухи кистей рук людини та передавати їх роботизованому пристрою [9].

З останнім завданням з легкістю може виправити модуль, що базується на ARRMKit'e MedTech від ARRM.IO. Незважаючи на те, що це рішення здатне розпізнавати жести з високою швидкістю та точністю, не вимагає великих обчислювальних потужностей та дорогого апаратного забезпечення, що дозволить залучити до участі в операції або експерименті фахівців із будь-якої точки планети.

З'явиться можливість проведення небезпечних експериментів на безпечній відстані, не наражаючи навіть на мінімальну небезпеку здоров'я та життя людей. Умови всередині чистих приміщень будуть максимально наближені до ідеальних, а наявність сторонніх частинок зведена до мінімуму.

Навіть у нинішньому стані чисті кімнати забезпечують високу якість процесу відновлення та лікування після складних операцій, наприклад, пов'язаних з пересадкою кісткового мозку, а також при захворюваннях кровоносної та імунної системи.

Чисті кімнати у пологових відділеннях результативно знижують рівень смертності дітей та матерів, що відбувається через післяпологові інфекції, важкий токсикоз вагітних та внаслідок передчасних пологів.

Безперечно, можна легко уявити ще множину варіантів того, яких проривних результатів дозволить досягти вдосконалення цього важливого винаходу людства, проте основна думка залишається незмінною – це буде новою віхою розвитку світової науки та медицини.

### 1.1.3 Використання технологій керування жестами у військовій промисловості

Важко уявити, як можна застосувати комп'ютерний зір та віддалене керування жестами у військовій справі. Такого роду справи часто потребують прямого контакту для вирішення тієї чи іншої задачі. Але іноді питання збереження життя військового та мінімізації втрат військового складу постають перед офіцерами та генералами, і відповідь можна знайти в сфері передових комп'ютерних технологій.

Інженерно – саперна справа потребує обережності та відповідальності. Сапер не має права на помилку та повинен бути освіченим фахівцем, який знає своє діло досконало та виконує його при будь яких умовах. В Україні служби надзвичайних ситуацій розмінують багато вибухівки та мін, використовуючи інструменти та навчених для цього собак. Але все ж одно є випадки неконтрольованих детонацій снарядів, які несуть потенційну небезпеку для людини.

Для вирішення проблеми потенційної небезпеки може бути віддалене розмінування боєприпасів за допомогою роботів з антропоморфними маніпуляторами, керовані планшетом з маніпуляторами. Однак ці маніпулятори не зможуть повторити точні рухи людської руки і не завжди є ефективні.

Альтернативним рішення є дратові рукавички компанії DataGlove, які подають дані про положення та обертання рук за допомогою датчиків у комп'ютер та передавати його на механізованого робота з антропоморфними маніпуляторами, які будуть проводити точні маніпуляції з небезпечними снарядами. За допомогою такого рішення з'явиться можливість проводити небезпечні операції з нульовою загрозою для життя.

Розпізнавання жестів – це здатність виявити контекстне значення руху людини за допомогою алгоритмів. Це дозволяє людям і машинам взаємодіяти один з одним, не вимагаючи механічних пристроїв введення [10].

Відбулося значне зростання інтересу до розробки роботизованих систем зі здатністю розуміти контекстно-орієнтовані команди.

Прикладом людського контекстного руху є люди, які подають команди один одному за допомогою жестів. Ці дослідження були зосереджені на створенні моделі розпізнавання, яка розуміє 12 команд рівня загону, наприклад, зібратися, послухати, зупинитися і прийти сюди (рис. 1.1).

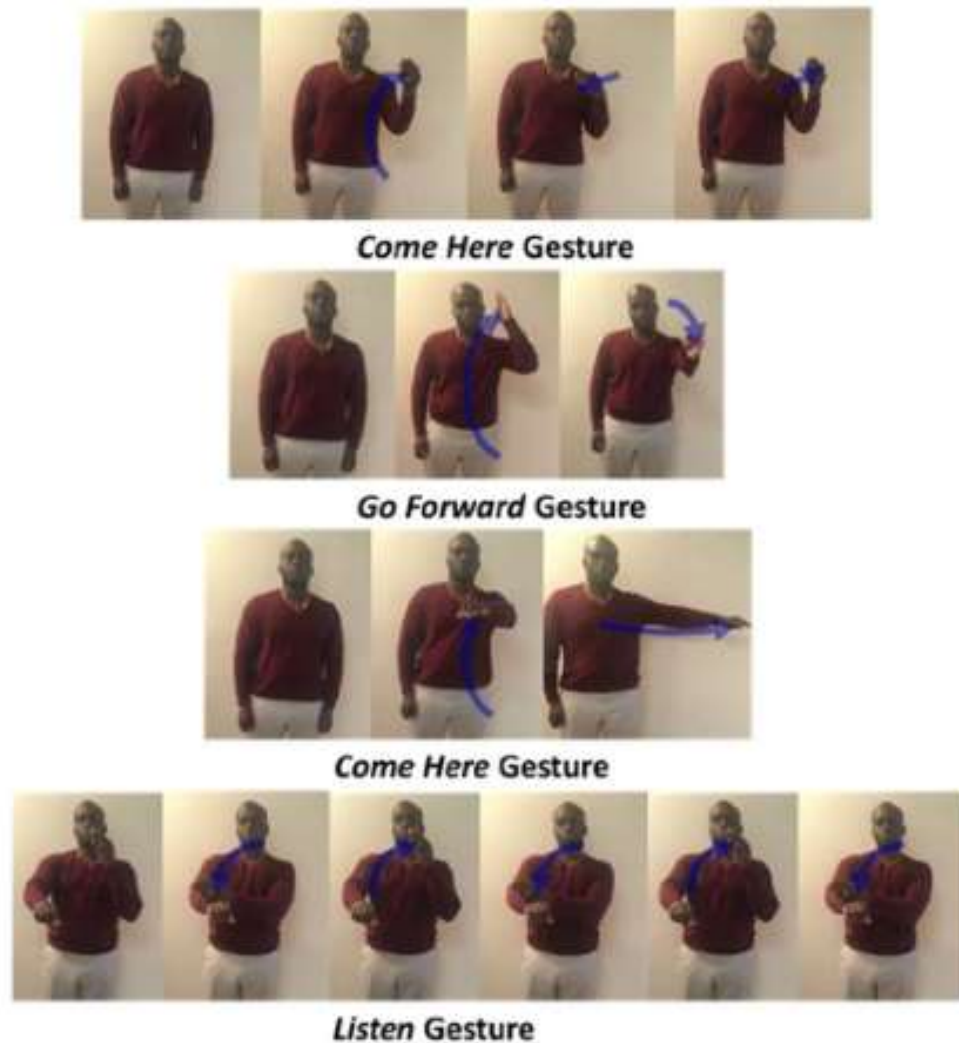


Рисунок 1.1 – Мова жестів команд для загону

Модель розроблена з використанням мультикласифікаційного підходу машинного навчання з функцією активації логістичної регресії. Вхідні дані для моделі зібрані за допомогою скелетної моделі Microsoft Kinect для визначення зміни положення лівої руки, правої руки, ліктя та плеча відносно один одного протягом усього руху.

Зміна підходу до позиції була прийнята, оскільки очікується, що модель буде функціонувати як з оптичними, так і з неоптичними датчиками, такими як інерційні одиниці вимірювання.

Логістична модель показала, що загалом ефективна на 96,75% при розрізненні наборів даних із даного набору членів. Модель рішення була ефективною на 90 відсотків у визначенні класу жестів, який представляє даний набір даних. Ця модель зараз інтегрується в систему розпізнавання для подальшого тестування ненавчених наборів даних.

Довгострокова мета цього проєкту – забезпечити роботизовані системи не тільки для вивчення навчених контекстних жестів, але й для вивчення нових команд за допомогою методів глибокого навчання та нейронних мереж. Бажання полягає в тому, щоб роботизовані системи були партнерами у бойових діях загону, взаємодіючи з бійцями [2].

#### 1.1.4 Використання технологій керування жестами у віртуальній та доповненій реальності

Зараз такий час, коли можливо об'єднати реальний та віртуальний світи та отримати неймовірні перетинання та комбінації.

Завдяки технологіям віртуальної (VR) та доповненої (AR) реальностей можливо створити науково – фантастичний всесвіт, яка буде невід'ємною частиною нашої реальності [11].

Доповнена реальність – це технологія, спрямована на доповнення реальності віртуальними елементами. По суті, це можна назвати змішаною реальністю (MR), яка створена за допомогою комп'ютера та застосункових елементів сприйняття реальності. Спробувати її можна, одягнувши розумні окуляри та за допомогою камер смартфонів та планшетів. Інструменти доповненої реальності можна використовувати в різних сферах, ігрових або сферах маркетингу.

Фільтри Snapchat та Instagram додають до фотографій та відео різні ефекти, як маски, макіяж, елементи одягу та можуть змінити людину до невпізнанності.

Ігровий хіт 2016 року Pokemon GO – гра компанії Nintendo, де за допомогою камери телефону люди ловлять фантастичних маленьких тварин в парках і офісах та можуть взаємодіяти з ними. Гра потребує фізичної активності та рухливості, щоб її пройти.

Сфера маркетингу та торгівлі пропонує покупцям багато розумних рішень. Наприклад, віддалена примірка товарів. Дистриб'ютор спортивного одягу Adidas пропонує покупцям віртуально приміряти кросівки за допомогою функції у застосунку. За допомогою смартфона застосунок накладає цифрове зображення кросівка на ногу користувача та слідкує за рухами тіла для відображення 3D моделі. Функція допоможе компанії краще виявити побажання покупців та зменшити кількість повернення товару.

Технології доповненої реальності роблять процес навчання безпечним, переміщуючи його у віртуальну реальність. AR дозволяє моделювати та практикувати небезпечні сценарії подій для таких важливих професій, як лікарі, пожежники, військові та льотчики з мінімальними ризиками.

У травні 2021 року в Nevada Spine Clinic хірургами була виконана важка операція на хребті. З використання роботизованої платформи Medtronic Mazor X та AR – гарнітури xvision операція, що, зазвичай, займає 6–7 годин, була завершена менше ніж за 2 години.

Морські піхотинці США за допомогою гарнітури Microsoft HoloLens MR розробили систему віртуального занурення для тренування. Віртуальна імітація допомагає військовим покращити навички та підготуватись до реальних битв.

З кожною новою технологією спосіб життя трішки змінюється. В майбутньому пристрої розширеної реальності зможуть отримувати відчуття та досвід користувача, дадуть можливість дивитися фільми та спортивні змагання з ефектом присутності.

Декілька десятків років тому ми не могли уявити таким світ. Розширена та інші види реальності є кроком у неймовірне майбутнє.

## 1.2 Аналіз літературних джерел щодо існуючих підходів розроблення програмних застосунків управління курсором за допомогою жестів рук

Розробка застосунків керування курсором за допомогою жестів рук та систем розпізнавання жестів є доволі актуальною у наш час.

За останні 15 років дискусій та досліджень навколо теми розпізнавання та детектування жестів та рухів кінцівок зазнали великого прогресу. Написано багато статей та монографій, проведено тести та експерименти.

В процесі написання роботи були опрацьовані важливі та актуальні дослідження, короткий огляд яких приведено нижче.

У статті [3] розглянуто спосіб керування електроприладами за допомогою керування жестами рук, щоб усунути необхідність використання пульта дистанційного керування. У статті представлено систему, розроблену за допомогою MATLAB та середовища Arduino, яка на основі максимізації очікувань може керувати різними приладами, використовуючи розпізнавання жестів рук за допомогою вебкамери у реальному часі.

Метою наукової публікації [4] є підвищення ефективності систем розпізнавання жестів на основі нейромережових алгоритмів аналізу зображень. Виконано аналітичний огляд програмних рішень та підходів до реалізації систем розпізнавання жестів, включаючи стадії виявлення, відстеження та розпізнавання. Представлено структурну схему систем розпізнавання жестів та алгоритм розпізнавання на основі нейронних мереж. Розроблено та обрано дескриптори зразків та структуру класифікаторів для досягнення кращих результатів.

У роботі [5] наголошується на використанні машинного навчання та нейронної мережі згортки (CNN) для розпізнавання жестів рук у реальному

часі, незважаючи на розмір та просторове положення рук на зображенні, надаючи вхідні дані для реалізації моделі, яка класифікує жести по різних категоріям. CNN використовує три шари, де два є прихованими шарами, а один згорткою. Модель розроблена з трьох класів, які містять персоналізовані жести. Розглядаються різні види занять: перша допомога, їжа та вода. Модель можна використовувати для комфорту у польоті мандрівниками, а також там де є потреба у цих жестах.

Публікація [6] дає огляд на актуальну ситуацію з використанням систем відеокommунікацій для людей з вадами слуху та зору. Описано функції розпізнавання мови жестів у системах відеокommунікацій та дозволяють подолати бар'єр та вивести спілкування на новий рівень.

У роботі [7] описується проблеми систем «розумного» дому та аналіз і виділення об'єкту з відеопотоку в реальному часі. Описується задача відстеження об'єкта, узгодження зображення з базою даних та пошук дублікатів. Приведено різні методи вибору об'єкта та відстеження та його алгоритм. Описані критерії ,які важливі при виборі метода роботи з об'єктом.

Робота [8] пропонує спосіб новий підхід до розпізнавання жестів рук, який базується на швидшому алгоритмі глибокого навчання регіональної згорткової нейронної мережі (Faster-RCNN) з п'ятьма шарами нейронної мережі. Використовується тестова база даних з жестами, кілька об'єктів вибирається в якості об'єкта обробки. Модель встановлюється з певними номерними зразками для навчання. Результати перевірки моделі показують, що алгоритм може ідентифікувати категорії жестів ефективно, швидко і точно з низькими витратами на обчислення. Точність може досягати 99,2%, що має велике значення для застосування взаємодії людини з комп'ютером.

### 1.3 Постановка задачі

На сьогоднішній день актуальність даної роботи є важливою, оскільки системи розпізнання жестів, незважаючи на наукові статі та роботи по цій тематиці, не є добре розвинутими та інтегрованими в широкий спектр сфер життя людей. Треба знайти оптимальні методи та підходи для розпізнання жестів та вирішення пов'язаних з цією сферою проблем.

Об'єктом роботи є послідовність різноракурсних жестів рук.

Метою роботи є розроблення програмного застосунку для керування курсором за допомогою жестів рук.

Враховуючи мету роботи, необхідні вирішити поставлені завдання:

- дослідити методи визначення точок на зображенні рук;
- проаналізувати методи визначення кінцівок рук;
- розробити методіку керування курсору за допомогою жестів;
- виконати всі етапи розроблення застосунку для керування комп'ютером за допомогою рук;
- проаналізувати підходи для ідентифікації рук як маніпуляторів за різними видами ознак;
- провести тестування розробленого застосунку та проаналізувати результати;
- визначити перспективи подальшої роботи.

## **2 МОДЕЛЮВАННЯ СТРУКТУРИ ПРОГРАМНИХ ЗАСТОСУНКІВ УПРАВЛІННЯ КУРСОРОМ ЗА ДОПОМОГОЮ ЖЕСТІВ РУК**

### **2.1 Застосування методів системного аналізу до предметної області «Управління курсором за допомогою жестів рук»**

Основною метою вивчення розпізнавання жестів є впровадження систем, які може виявляти конкретні людські жести та використовувати їх для передачі інформації або для командування та контролю. Такі системи включають не тільки відстеження руху людини, а й інтерпретацію цього руху на команди [10].

Для інтерпретації жестів для взаємодії людини та комп'ютера (НСІ) зазвичай використовуються два підходи. Перший підхід заснований на зчитуванні даних рукавичок, а другий підхід заснований на комп'ютерному баченні, без необхідності носити датчики. Датчики на рукавичках можна використовувати для фіксації руху та положення рук. На додачу, рукавиці можуть легко надати точні координати розташування долоні та пальців, орієнтацію за допомогою датчиків, прикріплених до рукавичок [11].

Однак, цей підхід вимагає від користувача фізичне підключення до комп'ютера, що блокує легкість взаємодії між користувачем і комп'ютером. Крім того, ціна на ці пристрої досить висока. Однак сучасний підхід на основі рукавичок використовує технологію дотику, яка є більш перспективною і вважається промислово – тактильною технологією. Така рукавичка дає тактильний зворотний зв'язок, завдяки якому користувач відчуває форму, текстуру, рух і вагу віртуального об'єкта.

Датчик на основі бачення камери є поширеною, придатною до застосування технікою, оскільки забезпечує безконтактний зв'язок між людиною та комп'ютером [12]. Різні конфігурації можна використовувати камери, такі як монокуляр, риб'яче око, ToF [13].

Однак, ця техніка пов'язана з кількома проблемами, включаючи зміну освітлення, проблеми з фоном, складний фон, час обробки залежить від роздільної здатності та частоти кадрів і переднього плану [14].

Основні методи, які використовуються для розпізнання образів:

- розпізнавання на основі кольору за допомогою маркера для рукавичок;
- розпізнавання на основі руху;
- розпізнавання на основі скелета.

### 2.1.1 Розпізнавання на основі кольору за допомогою маркера для рукавичок

Цей метод використовує камеру для відстеження руху руки за допомогою рукавиці з різними кольоровими мітками (рис. 2.1).



Рисунок 2.1 – Рукавиці з кольоровими мітками

Цей метод використовується для взаємодії з 3D-моделями, що дозволяє деякі обробки, такі як масштабування, переміщення, малювання та запис за допомогою віртуальної клавіатури з гарною гнучкістю [15].

Кольори на рукавичці дозволяють датчику камери відстежувати та визначати місцезнаходження долоні та пальців, що дозволяє витягти геометричну модель форми кисті [16, 17].

Перевагами цього методу є його простота використання та низька ціна в порівнянні з рукавичкою з датчиками для зчитування даних [15]. Однак, це все одно вимагає носіння кольорових рукавичок і обмежує ступінь природної і спонтанної взаємодії з НСІ [17].

### 2.1.2 Розпізнавання на основі руху

Розпізнавання на основі руху можна використовувати для цілей виявлення об'єкта через серію кадрів зображень. Головна проблема розпізнати рухи – це коли під час розпізнавання активні ще декілька жестів, а також динамічний фон негативно впливає на результат [18].

Крім того, втрата жесту може бути викликано помилкою вилучення області з відстежуваним жестом та вплив далеких відстаней на зовнішній вигляд частини зображення [19].

Запропоновано два етапи ефективного виявлення рук. Рука виявлена для кожної рамки і центральна точка використовуються для відстеження руки. Другий етап застосування моделі відповідно до кожного типу жесту використовує набір функцій, які відстежують рух, щоб забезпечити кращу класифікацію, де на основний недолік кольору шкіри впливають різні типи освітлення. Алгоритм AdaBoost (рис. 2.2) дає збільшення продуктивності любого алгоритму машинного навчання, що використовується для виявлення об'єктів, характеристик та моделювання руху [12].

Алгоритм краще працює зі слабкими навчальними алгоритмами, тому такі моделі можуть досягти високої точності завдяки вирішенню задач класифікації. Найбільш поширеними алгоритмами є однорівневі дерева рішень.

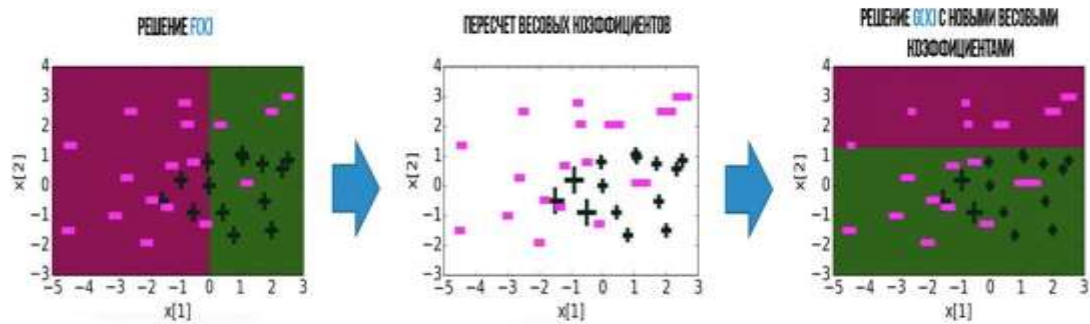


Рисунок 2.2 – Работа алгоритма AdaBoost

Слабкий навчальний алгоритм є класифікатором, який погано працює відносно точності. Але слабкі класифікатори легко обчислюються, тому можна додавати більше змінних до алгоритму, щоб створити сильніший класифікатор [20].

Кроки реалізації алгоритму:

- спочатку всім точкам присвоєно рівні ваги;
- модель будується на вибірці даних;
- створюються передбачення для всіх даних моделі;
- за прогнозами та справжніми значеннями обчислюються помилки;
- у наступній ітерації моделі найбільші ваги надаються точкам даних, на передбаченні яких алгоритм помилився;
- вага визначається за величиною помилки. Чим більша помилка, тим більша вага;
- процес повторюється, поки функція помилки не перестане змінюватись.

Перевагами алгоритму є простота та швидкість реалізації.

Є можливість комбінувати з будь – яким алгоритмом машинного навчання та використовувати для роботи з числовими або текстовими даними.

Недоліком є вразливість алгоритму до рівномірно розподілених даних. Дуже слабкі класифікатори можуть призвести до поганих результатів та повторному перенавченні [21].

### 2.1.3 Розпізнавання на основі скелета

Розпізнавання на основі скелета визначає параметри моделі, які можуть покращити виявлення комплексних ознак [12].

Різні представлення даних скелета руки використовуються для класифікації, де він описує геометричні атрибути та обмеження та легко перекладає ознаки та кореляції даних, щоб зосередитися на геометричних і статистичних ознаках. Найпоширенішими ознаками є орієнтація суглоба, простір між суглобами, розташування скелетного суглоба та ступінь кута між суглобами та траєкторії та кривизна суглобів (рис. 2.3).

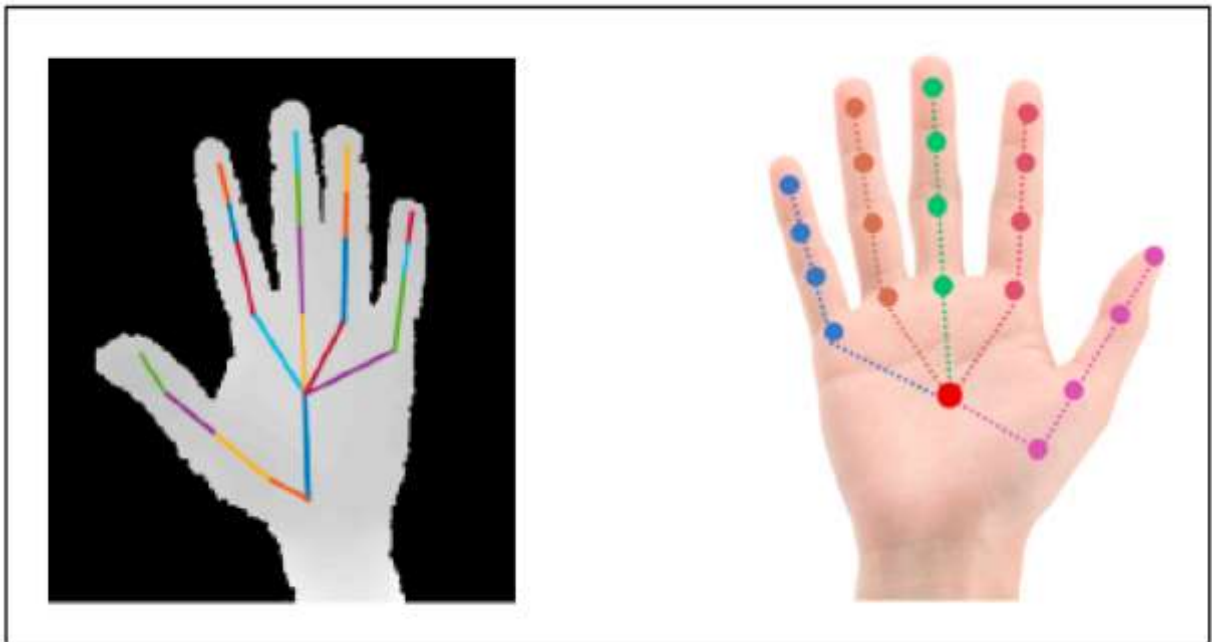


Рисунок 2.3 – Скелетна модель руки

Новий підхід до скелетного жестів руками на основі моделі глибокого навчання з використанням паралельних згорткових нейронних мереж (CNN) для обробки вручну положення скелетних суглобів було введено в [22], запропонована система має обмеження, де працює тільки з повною послідовністю кадрів.

Послідовність тривимірних даних скелету руки позначено як вектор  $s$ :

$$s = (p_1 \dots p_n)^T, \quad (2.1)$$

де компоненти  $p_i$  є багатовимірними часовими послідовностями.

Кожна складова  $p_i = (p_i(t))_{t \in \mathbb{R}}$  є багатоваріантною послідовністю з трьома одновимірними компонентами:

$$p_i = (x^{(i)} + y^{(i)} + z^{(i)}), \quad (2.2)$$

де всі представляють часову послідовність позицій  $p_i(t)$   $i$ -го скелетного суглоба.

Кожен скелетний суглоб представляє собою її чітку та точну артикуляцію у фізичному світі.

## 2.2 Особливості нейронної мережі R-CNN для управління курсором за допомогою жестів рук

Комп'ютерний зір є галузею штучного інтелекту, яке стосується того, як комп'ютери можуть розуміти цифрові зображення та відео на рівні людей. Невід'ємною частиною комп'ютерного зору є виявлення об'єктів. Виявлення об'єктів допомагає оцінити пози, виявити транспортний засіб, спостерігати тощо. Різниця між алгоритмами виявлення об'єктів та алгоритмами класифікації полягає в тому, що в алгоритмах виявлення намагаємося намалювати обмежувальну рамку навколо об'єкта, який цікавить, щоб знайти його на зображенні. Крім того, не обов'язково малювати лише одну обмежувальну рамку у випадку виявлення об'єкта, може бути багато обмежувальних рамок, що представляють різні об'єкти у зображенні, і не знаєте, скільки їх наперед [23].

Основна причина, чому не можете продовжити вирішення цієї проблеми, побудувавши стандартну згорткову мережу, а потім повністю зв'язаний шар, полягає в тому, що довжина вихідного шару є змінною, а не постійною, це тому, що кількість входжень об'єктів, що цікавлять, є не фіксований.

Наївним підходом до вирішення цієї проблеми було б взяти різні регіони, що цікавлять зображення, і використовувати CNN для класифікації присутності об'єкта в цій області [24].

Проблема цього підходу полягає в тому, що об'єкти, що цікавлять, можуть мати різне просторове розташування всередині зображення та різні співвідношення сторін [25-28]. Таким чином, вам доведеться вибрати величезну кількість регіонів, і це може вибухнути з точки зору обчислень. Тому такі алгоритми, як R-CNN та його модифікації були розроблені, щоб знайти ці випадки та швидко знайти їх.

### 2.2.1 R-CNN

Одним із підходів визначення знаходження об'єкта на зображенні, є R-CNN (Neural Network Region Convolution).

Його архітектура складається з кількох послідовних виконуваних кроків:

Крок 1. Визначення набору гіпотез.

Крок 2. Витягнення із запропонованих регіонів ознак за допомогою сверткової нейронної мережі та їх кодування у векторі.

Крок 3. Класифікація об'єкта всередині гіпотези на основі вектора з Кроку 2.

Крок 4. Покращення (корректировка) координат гіпотези.

Крок 5. Все повторюється з Кроку 2, поки не будуть оброблені всі гіпотези з Кроку 1.

Щоб обійти проблему виділення величезної кількості регіонів, у [25] Росс Гіршик та інші запропонували метод, за допомогою якого використовуємо вибіркового пошуку для вилучення лише 2000 регіонів із зображення, автори назвали їх пропозиціями регіонів. Тому тепер замість того, щоб намагатися класифікувати величезну кількість регіонів, можна просто працювати з 2000 регіонами [29]. Ці 2000 пропозицій регіонів генеруються за допомогою алгоритму вибіркового пошуку, який зазначено нижче.

Вибірковий пошук:

- створити початкову підсегментацію, згенерувавши багато регіонів – кандидатів;
- використати жадібний алгоритм для рекурсивного об'єднання подібних регіонів у більші регіони;
- використати згенеровані регіони, щоб створити остаточні пропозиції регіонів – кандидатів.

Ці 2000 пропозицій регіонів – кандидатів деформуються в квадрат і подаються в згорткову нейронну мережу, яка створює 4096 вимірний вектор ознак як вихід. CNN діє як екстрактор ознак, а щільний вихідний шар складається з об'єктів, витягнутих із зображення, а вилучені об'єкти подаються в SVM для класифікації присутності об'єкта в цій пропозиції регіону – кандидата.

Алгоритм також передбачає чотири значення, які є значеннями зміщення, щоб збільшити точність рамки. Наприклад, з огляду на пропозицію регіону, алгоритм передбачав би присутність людини, але обличчя цієї особи в пропозиції регіону могло бути скорочено навпіл. Таким чином, значення зміщення допомагають налаштувати обмежувальну рамку пропозиції регіону.

Проблеми з R-CNN. Навчання мережі все ще займає величезну кількість часу, оскільки доведеться класифікувати 2000 пропозицій регіонів на одне зображення.

Його неможливо реалізувати в реальному часі, оскільки для кожного тестового зображення потрібно близько 47 секунд. Алгоритм селективного пошуку є фіксованим алгоритмом. Тому на цьому етапі навчання не відбувається. Це може призвести до створення поганих пропозицій щодо регіонів-кандидатів [15].

### 2.2.2 Fast R-CNN

Щоб виявити більш точний вибір об'єктів, швидка модель R-CNN зазвичай генерує багато пропозицій регіонів у вибраному пошуку [26]. Щоб зменшити пропозиції регіонів без втрати точності, швиденький R-CNN пропонує замінити вибірковий пошук у мережі пропозицій регіону (рис. 2.4).

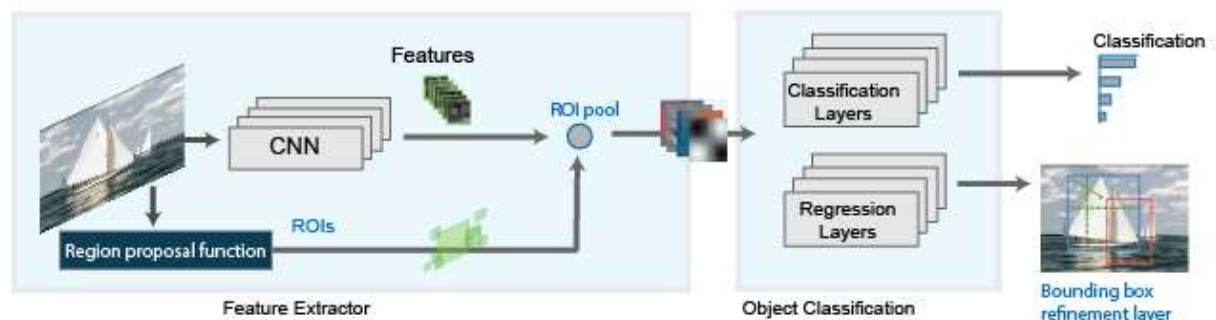


Рисунок 2.4 – Алгоритм Fast R-CNN

Процес обробки зображення змінився і виглядає таким чином:

- вибір карт ознак зображень (не для кожної гіпотези окремо, а для всього зображення);
- пошук гіпотез (аналогічний R-CNN на основі селективного пошуку);
- складання кожної гіпотези з місцем на карті ознак – використання єдиного набору визначених ознак для кожної гіпотези (координати гіпотез можна однозначно встановити з розташуванням на мапі ознак);

– класифікація будь-якої гіпотези та виправлення координат обмеженої рамки (цю частину можна запускати паралельно, оскільки більше не залежить від SVM-класифікації).

У концепції R-CNN запропонована гіпотеза за допомогою окремої роботи за допомогою CNN. Для вирішення цієї проблеми був розроблений шар регіону інтересів (RoI) [16]. Цей шар дозволяє єдино обробляти зображення за допомогою нейронної мережі, одержуючи кожен вихідну карту ознак, яка далі використовується для обробки гіпотези.

Основною задачею шару RoI є встановлення координат гіпотез (координати обмежуючих рамок) з відповідними координатами карт ознак. Виконуючи «зріз» карти ознак, шар RoI подає його на вхід повнозв'язного шару для подальшого визначення класу та поправок по координатам.

### 2.2.3 Faster CNN

Наступним логічним покращенням є спосіб видалення залежності від алгоритма селективного пошуку. Для цього подамо всю систему як композицію двох модулів (рис. 2.5) – визначення гіпотез та їх обробки [30].

Перший модуль буде реалізовуватися за допомогою мережі регіональних пропозицій (RPN), а другий аналогічно Fast R-CNN (починаючи з шару RoI).

Робота із зображенням відбувається таким чином:

- вилучення карт із зображеннями за допомогою нейронної мережі;
- генерація на основі отриманої карти визначення гіпотези – визначення приблизних координат та належність об'єкта будь-якому класу;
- складання координат гіпотез за допомогою RoI з картою зображень, отриманих на першому етапі;
- класифікація гіпотез (на визначенні конкретного класу) і додаткове уточнення координат.

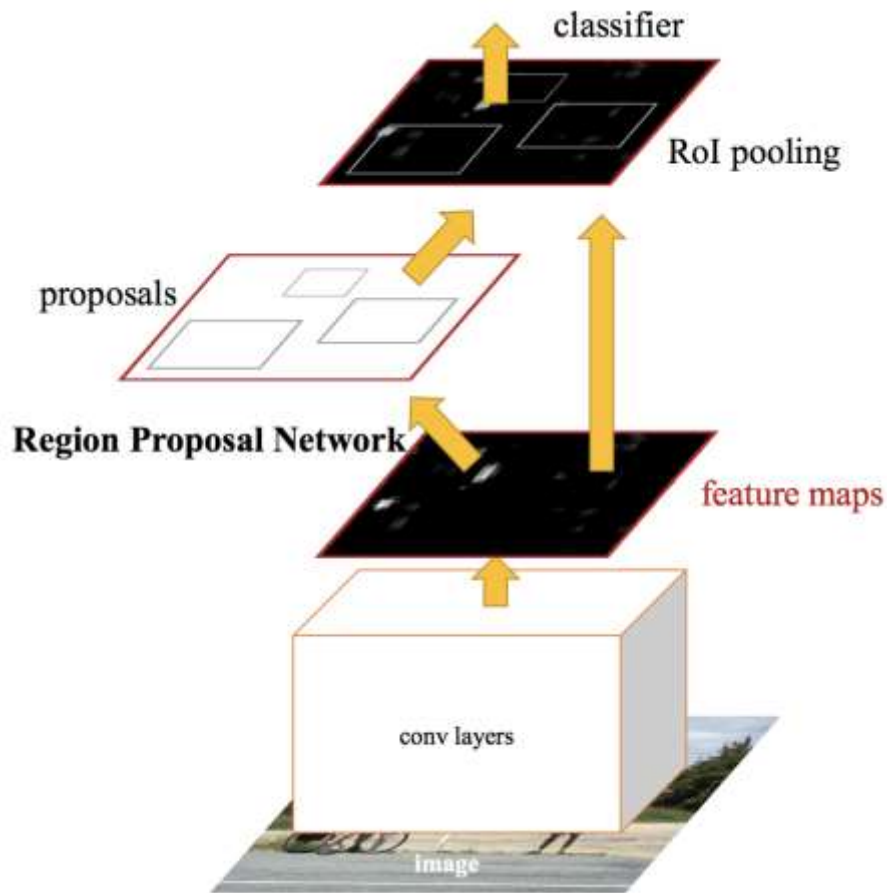


Рисунок 2.5 – Алгоритм Faster R-CNN

### 2.3 Моделювання структури та наповнення програмного застосунку для управління курсором за допомогою жестів рук

Функції та умов, що використовуються в системі, пояснюються на блок-схемі роботи алгоритма керування курсором (рис. 2.6).

Камера, що використовується в системі керування мишею. Запропонована система керування жестами заснована на кадрах, які були зняті вебкамерою. За допомогою бібліотеки комп'ютерного зору Python OpenCV створюється об'єкт захоплення відео, і вебкамера починає записувати відео. Вебкамера фіксує та передає кадри у систему.

Зйомка відео та обробка. Система віртуальної миші AI використовує вебкамеру, де фіксується кожен кадр до завершення програми.

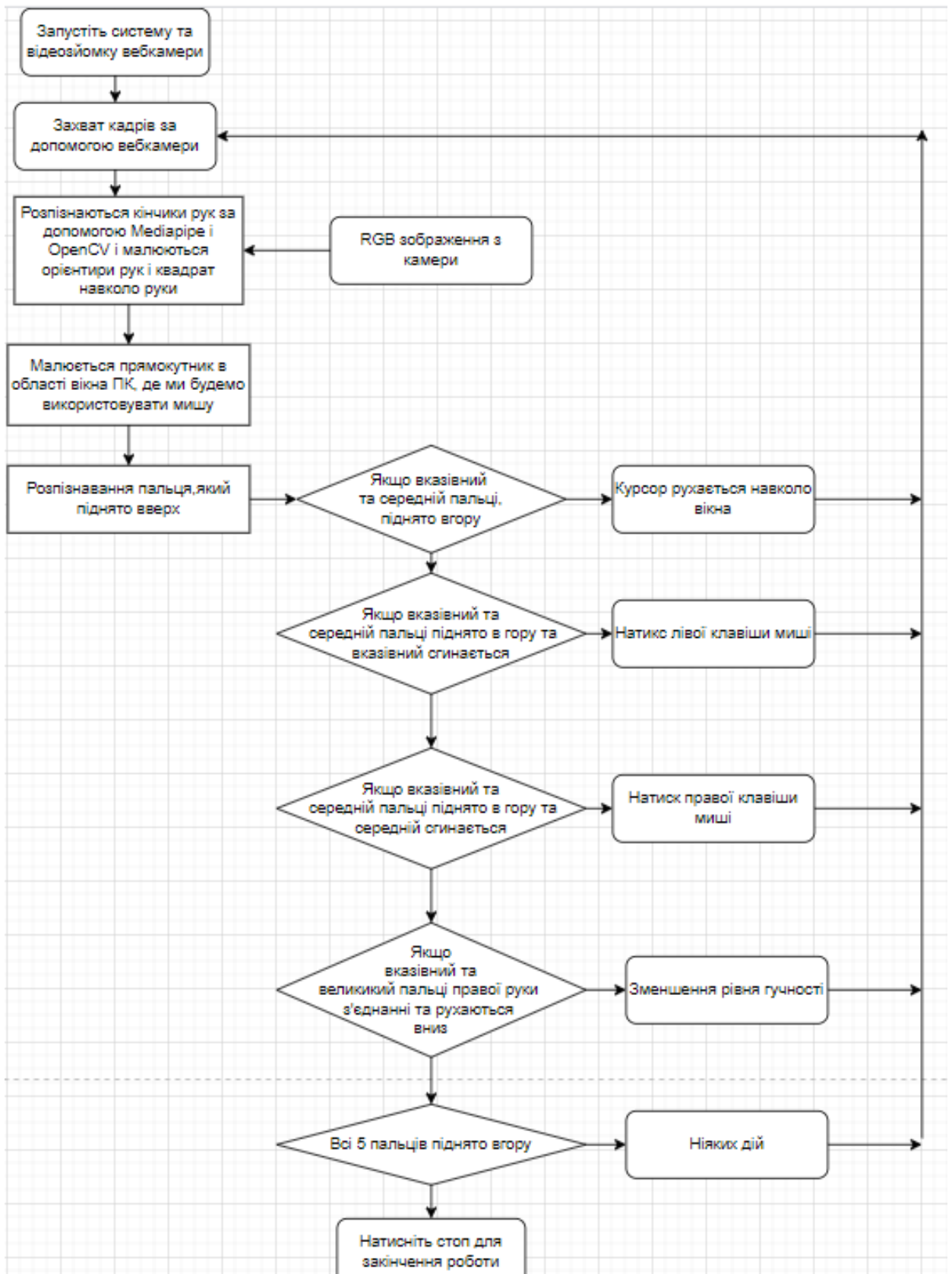


Рисунок 2.6 – Блок-схема алгоритму розпізнання жестів та керування курсором

Відеокадри обробляються від колірного простору BGR до RGB, щоб знайти руки у відеокадр за кадром.

Прямокутна область для переміщення у вікні екрану. Система використовує алгоритм трансформації, вона перетворює координати кінчика пальця з екрана вебкамери на повноекранний екран комп'ютера для керування мишею. Коли виявлено руки та коли знаходимо, який палець піднятий для виконання певної функції миші, прямокутне поле малюється відносно частини екрану комп'ютера в області вебкамери, де переміщаємося по всьому екрану за допомогою курсору миші.

Визначення того, який палець піднятий, виконання певної функції миші. На цьому етапі визначаємо, який палець знаходиться вгорі, використовуючи ідентифікатор кінчика відповідного пальця, який знайшли за допомогою пакета функцій MediaPipe, і відповідні координати пальців, які знаходяться вгорі, після цього виконується конкретна функція миші.

Функції миші залежно від жестів рук і виявлення кінчиків рук за допомогою комп'ютерного зору. Курсор миші переміщується по вікну комп'ютера.

Якщо вказівний палець вгору з наконечником  $Id = 1$  або вказівний палець з наконечником  $Id = 1$  і середній палець з наконечником  $Id = 2$  вгору, курсор миші переміщується по вікну комп'ютера за допомогою пакета AutoPy Python. Миша виконує клацання лівою кнопкою.

Якщо і вказівний палець з  $Id = 1$ , і великий палець з  $Id = 0$  підняті, а відстань між двома пальцями менше 30 пікселів, комп'ютер змушений виконувати клацання лівою кнопкою миші за допомогою функції `rnp.rpt`.

## 3 РОЗРОБЛЕННЯ ПРОГРАМНОГО ЗАСТОСУНКУ ДЛЯ УПРАВЛІННЯ КУРСОРОМ ЗА ДОПОМОГОЮ ЖЕСТИВ РУК

### 3.1 Вибір інструментальних засобів для реалізації поставленої задачі

Для виявлення жестів рук та відстеження кінцівок рук використовується фреймворк MediaPipe, а бібліотека OpenCV використовується для комп'ютерного зору та машинного навчання.

MediaPipe – це фреймворк, який використовується для застосування в конвеєрі машинного навчання, і є фреймворком з відкритим кодом Google. Фреймворк MediaPipe підтримує різні апаратні та операційні платформи, такі як Android, iOS та Windows. Інструмент корисний для міжплатформної розробки, оскільки структура побудована з використанням відео та аудіо даними у реальному часі.

Фреймворк MediaPipe є багатомодальним, де цю структуру можна застосувати до різних аудіо та відео проєктів. Фреймворк MediaPipe використовується розробником для побудови та аналізу систем за допомогою графіків, а також для розробки систем таких направленостей, де фреймворк застосовується. Кроки, задіяні в системі, яка використовує MediaPipe, виконуються в конфігурації конвеєра. Створений конвеєр може працювати на різних платформах, що забезпечує масштабованість на мобільних і настільних комп'ютерах.

Структура MediaPipe базується на трьох фундаментальних частинах: оцінка продуктивності, структура для отримання даних датчиків і набір компонентів, які називаються калькуляторами, їх можна використовувати повторно.

Конвеєр – це графік, який складається з компонентів, які називаються калькуляторами. У свою чергу, калькулятор є обчислювальним блоком, де кожен калькулятор з'єднаний потоками, через які проходять масиви даних.

Існують декількох видів: калькулятори попередньої обробки використовуються для обробки зображень та медіа, калькулятори виведення дозволяють інтегрувати нейронну мережу TensorFlow для машинного навчання, калькулятори постобробки виконують сегментацію та класифікацію об'єктів.

Розробники можуть замінити або визначити власні калькулятори в будь-якому місці графіка, створюючи власну програму. Об'єднані калькулятори та потоки створюють діаграму потоку даних. Графік створюється за допомогою MediaPipe, де кожен вузол є калькулятором, а вузли з'єднані потоками.

Модель однокадрового детектора використовується для виявлення та розпізнавання руки або долоні в режимі реального часу. Модель одноразового детектора використовується фреймворком MediaPipe. У модулі виявлення рук спочатку тренують для моделі виявлення долоні, оскільки тренувати долоні легше. Крім того, детектування об'єктів працює значно краще на невеликих предметах, таких як долоня або кулак. Модель орієнтира кисті складається з розташування координат суглоба або пальця в області кисті.

OpenCV – це бібліотека комп'ютерного зору, яка містить алгоритми обробки зображень для виявлення об'єктів. OpenCV – це бібліотека мови програмування Python [31-35]. Програми комп'ютерного зору в реальному часі можна розробляти за допомогою бібліотеки комп'ютерного зору. Бібліотека OpenCV використовується для обробки зображень і відео, а також для аналізу, наприклад, виявлення обличчя та виявлення об'єктів [36].

Середовищем розробки застосунку є Microsoft Visual Studio Code – редактор коду, доступний на різних платформах, таких як Windows, Linux та MacOS. Засіб є безкоштовним та дозволяє створювати та редагувати сучасні проекти.

Visual Studio Code є спрощеним та потужним редактором, який може запускатися на будь-якому пристрої, не дивлячись на його конфігурацію.

Екосистема Visual Studio Code підтримує багато розширень для декількох мов програмування, які надають додаткові функції до основних технологій.

Особливою перевагою VSC перед іншими конкурентами є наявність вбудованого зневажувача. Після налагоджування є можливість шукати помилки в коді прямо із редактора [28]. Наприклад, перевірити деякий відрізок коду, створивши точку зупинки та стежити за його виконанням.

У редактора є розширена вбудована консоль, де можна бачити результати роботи та повідомлення про помилки. Зневажувач можна налагодити під різні мови програмування та задачі.

### 3.2 Етапи розроблення програмного застосунку для управління курсором за допомогою жестів рук

Програмний застосунок складається з двох частин – головний клас, який відповідає за початок розпізнання жестів та їх класифікацію; частину, де кодовані всі пальці руки та функції з їх конвертування для розпізнання окремих функцій та самі функції для реалізації кожного жесту.

Робота програми починається з відкриття вебкамери та нейронна мережа намагається розпізнати руки та кінцівки пальців (рис. 3.1).



Рисунок 3.1 – Опис всіх суглобів руки

Кожен суглоб та з'єднання руки помічається червоною крапкою, що з'єднуються між собою зеленим відрізком, повторюючи скелетоподібну форму руки.

Спочатку руку треба тримати відкритою та прямою для кращого розпізнавання жестів. Кодування суглобів рук позначено у коді (рис. 3.2).

```
# Кодування жестів
class Gest(IntEnum):
    # Бінарне кодування
    FIST = 0
    PINKY = 1
    RING = 2
    MID = 4
    LAST3 = 7
    INDEX = 8
    FIRST2 = 12
    LAST4 = 15
    THUMB = 16
    PALM = 31

    # Додаткові відображення
    V_GEST = 33
    TWO_FINGER_CLOSED = 34
    PINCH_MAJOR = 35
    PINCH_MINOR = 36

# Кодування для кількох рук
class HLabel(IntEnum):
    MINOR = 0
    MAJOR = 1
```

Рисунок 3.2 – Кодування суглобів руки

Для створення застосунку за допомогою бібліотек глибокого навчання OpenCV Python було створено RCNN (регіони з згортковою нейронною мережею).

Спочатку мережа виділяє частину екрану, де помітно руку людини. Потім класифікатор намагається визначити положення руки та її відстань від камери. Щоб правильно визначити, яка це рука та пальці, нейронна мережа по замовчуванню позначає цифрами суглоби та з'єднує на руці.

При кращій деталізації зображення, мережа корегує позначення з'єднань.

Код класифікатора позначено на рисунку 3.3.

```

def __init__(self):
    GestureController.gc_mode = 1
    GestureController.cap = cv2.VideoCapture(0)
    GestureController.CAM_HEIGHT = GestureController.cap.get(cv2.CAP_PROP_FRAME_HEIGHT)
    GestureController.CAM_WIDTH = GestureController.cap.get(cv2.CAP_PROP_FRAME_WIDTH)

def classify_hands(results):
    left , right = None, None
    try:
        handedness_dict = MessageToDict(results.multi_handedness[0])
        if handedness_dict['classification'][0]['label'] == 'Right':
            right = results.multi_hand_landmarks[0]
        else :
            left = results.multi_hand_landmarks[0]
    except:
        pass

    try:
        handedness_dict = MessageToDict(results.multi_handedness[1])
        if handedness_dict['classification'][0]['label'] == 'Right':
            right = results.multi_hand_landmarks[1]
        else :
            left = results.multi_hand_landmarks[1]
    except:
        pass

    if GestureController.dom_hand == True:
        GestureController.hr_major = right
        GestureController.hr_minor = left
    else :
        GestureController.hr_major = left
        GestureController.hr_minor = right

```

Рисунок 3.3 – Класифікатор

При розпізнанні жестів є багато задач, які потрібно розібрати. Шум та погана деталізація зображення можуть неправильно детектувати руку.

Для цього був розроблений фільтр, який прибирає зайві шуми з екрана та робить картинку більш чіткою. Код зазначено на рисунку 3.4.

Для кожного жесту написана своя функція розпізнавання. Статичні жести, як натиск лівої та правої кнопки миші, потребують стабілізації курсору та отримання правильного положення руки. Для стабілізації руки система повинна розпізнати відкриту руку.

Коли виявлено руки і коли знайдено, який палець піднятий для виконання певної функції миші, прямокутне поле малюється відносно вікна комп'ютера в області вебкамери, де рука переміщається по всьому вікну за допомогою курсору миші. За допомогою бібліотеки MediaPipe визначається палець, який знаходиться відкритим, використовуючи ідентифікатор кінчика відповідного пальця і відповідно до цього виконується конкретна функція миші.

```

# Коливання через шум відеозйомки
def get_gesture(self):
    if self.hand_result == None:
        return Gest.PALM

    current_gesture = Gest.PALM
    if self.finger in [Gest.LAST3,Gest.LAST4] and self.get_dist([8,4]) < 0.05:
        if self.hand_label == HLabel.MINOR :
            current_gesture = Gest.PINCH_MINOR
        else:
            current_gesture = Gest.PINCH_MAJOR

    elif Gest.FIRST2 == self.finger :
        point = [[8,12],[5,9]]
        dist1 = self.get_dist(point[0])
        dist2 = self.get_dist(point[1])
        ratio = dist1/dist2
        if ratio > 1.7:
            current_gesture = Gest.V_GEST
        else:
            if self.get_dz([8,12]) < 0.1:
                current_gesture = Gest.TWO_FINGER_CLOSED
            else:
                current_gesture = Gest.MID

    else:
        current_gesture = self.finger

    if current_gesture == self.prev_gesture:
        self.frame_count += 1
    else:
        self.frame_count = 0

    self.prev_gesture = current_gesture

    if self.frame_count > 4 :
        self.ori_gesture = current_gesture
    return self.ori_gesture

```

Рисунок 3.4 – Фільтр шумів

Жести зміни гучності динаміків та прокручування сторінки потребують відстежування постійної зміни вертикального положення руки.

Для цього було встановлено значення в 5 кадрів на секунду. Код зазначено на рисунку 3.5.

```

# Утримуйте кінцеве положення протягом 5 кадрів, щоб змінити статус положення руки
def pinch_control(hand_result, controlHorizontal, controlVertical):
    if Controller.framecount == 5:
        Controller.framecount = 0
        Controller.pinchlv = Controller.prevpinchlv

        if Controller.pinchdirectionflag == True:
            controlHorizontal() #x
        elif Controller.pinchdirectionflag == False:
            controlVertical() #y

    lvx = Controller.getpinchxlv(hand_result)
    lvy = Controller.getpinchylv(hand_result)

    if abs(lvy) > abs(lvx) and abs(lvy) > Controller.pinch_threshold:
        Controller.pinchdirectionflag = False
        if abs(Controller.prevpinchlv - lvy) < Controller.pinch_threshold:
            Controller.framecount += 1
        else:
            Controller.prevpinchlv = lvy
            Controller.framecount = 0

    elif abs(lvx) > Controller.pinch_threshold:
        Controller.pinchdirectionflag = True
        if abs(Controller.prevpinchlv - lvx) < Controller.pinch_threshold:
            Controller.framecount += 1
        else:
            Controller.prevpinchlv = lvx
            Controller.framecount = 0

```

Рисунок 3.5 – Зміна вертикального положення руки

Результатами роботи є розпізнання руки як об'єкта керування курсором та позначення всіх його пальців та суглобів (рис. 3.6).

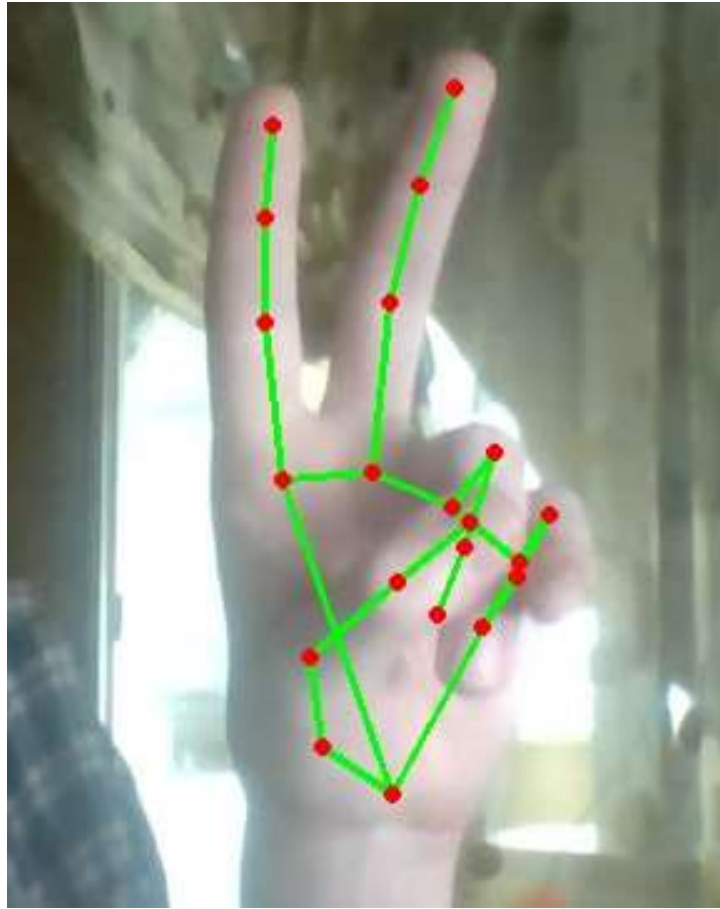


Рисунок 3.6 – Результати роботи

### 3.3 Тестування реалізованого програмного застосунку та аналіз результатів

Тест проводили 25 разів 3 особи, у результаті чого було 600 жестів з ручним маркуванням, цей тест виконано в різних умовах освітлення та на різній відстані від екрана, кожна людина перевіряла систему розпізнавання жестів 10 разів у нормальному освітленні, 5 разів в умовах слабкого освітлення, 5 разів на близькій та 5 разів на великій відстанях від вебкамери, а результати експерименту зведені на рисунку (рис. 3.7) та гістограмі (рис. 3.8).

Розпізнавання жестів	Вдалі спроби	Невдалі спроби	Точність
Рух мишею	100	0	100%
Ліва кнопка миші	98	2	98%
Правої кнопка миші	99	1	99%
Прокручування сторінки	90	10	90%
Збільшення/зменшення гучності динаміків	95	5	95%
Ніяких дій	100	0	100%
Результати	582	18	97%

Рисунок 3.7 – Результати розпізнавання

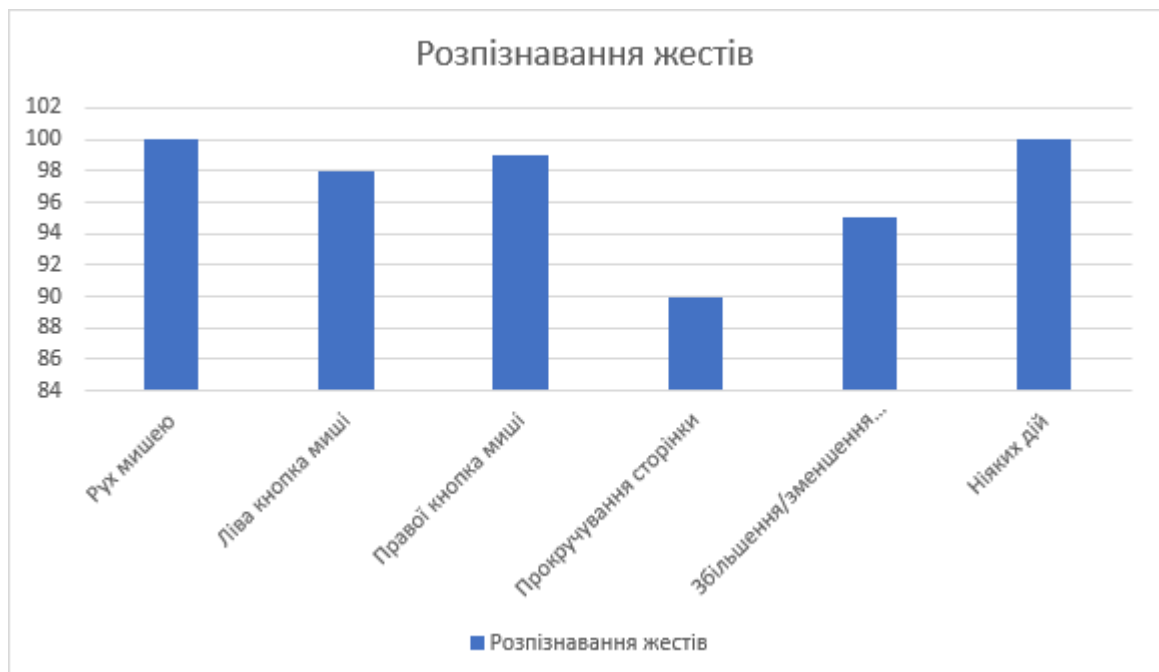


Рисунок 3.8 – Гістограма результатів

Хочу зауважити, що якість роботи застосунка напряму залежить від зовнішніх пристроїв сприйняття зорової інформації, у даному випадку це вебкамера. Результати роботи можуть відрізнятися від типу вебкамери та якості захоплення зображення.

У ході розробки застосунку, були спроектовані та реалізовані такі жести керування мишею:

- рух мишею;

- натиск лівої кнопки миші;
- натиск правої кнопки миші;
- прокручування сторінки;
- збільшення та зменшення гучності динаміків.

Жест «рух мишею» (рис. 3.9) виконується за допомогою випрямлення двох пальців – вказівного та середнього.

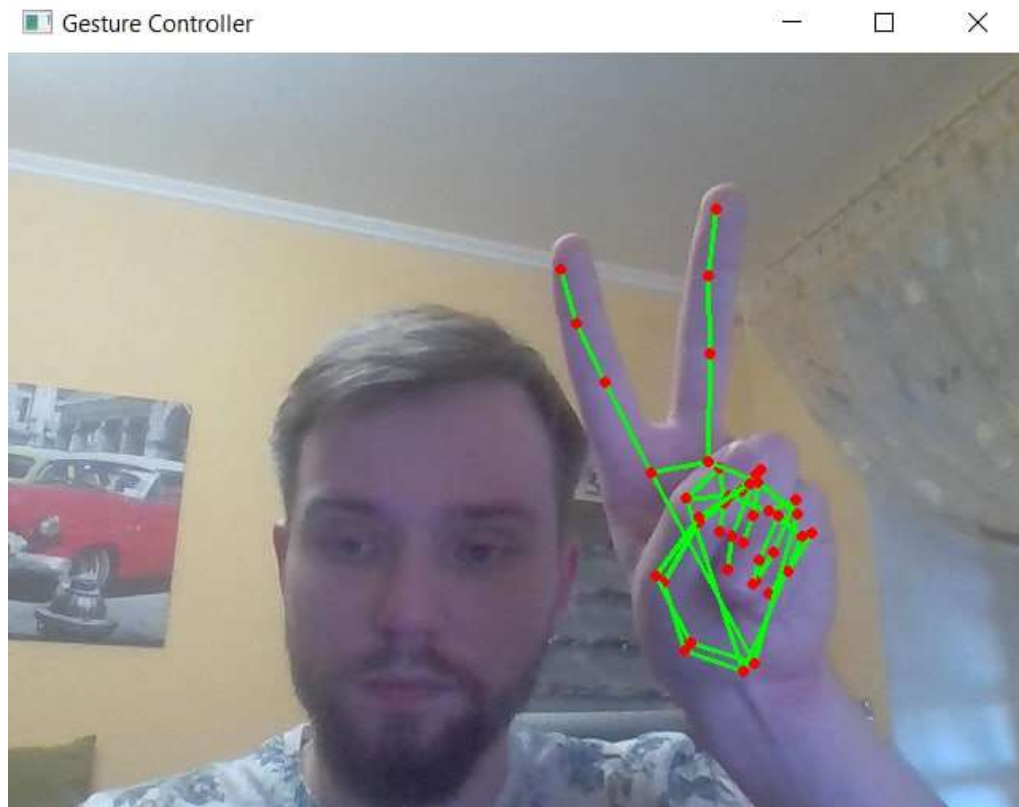


Рисунок 3.9 – Жест «рух мишею»

Такий жест імітує саму комп'ютерну мишу: вказівний палець відповідає за ліву кнопку мишу, середній – за праву. Жест «рух мишею» повторює рухи за курсором миші та його можна переміщати по всій поверхні екрану.

Жест «натиск лівої кнопки миші» (рис. 3.10) повторює попередній жест з невеликим доповненням – згинанням вказівного пальця. Жест імітує натиск лівої кнопки миші та виконує всі маніпуляційні дії, як відкриття та закриття папок, файлів та програм.

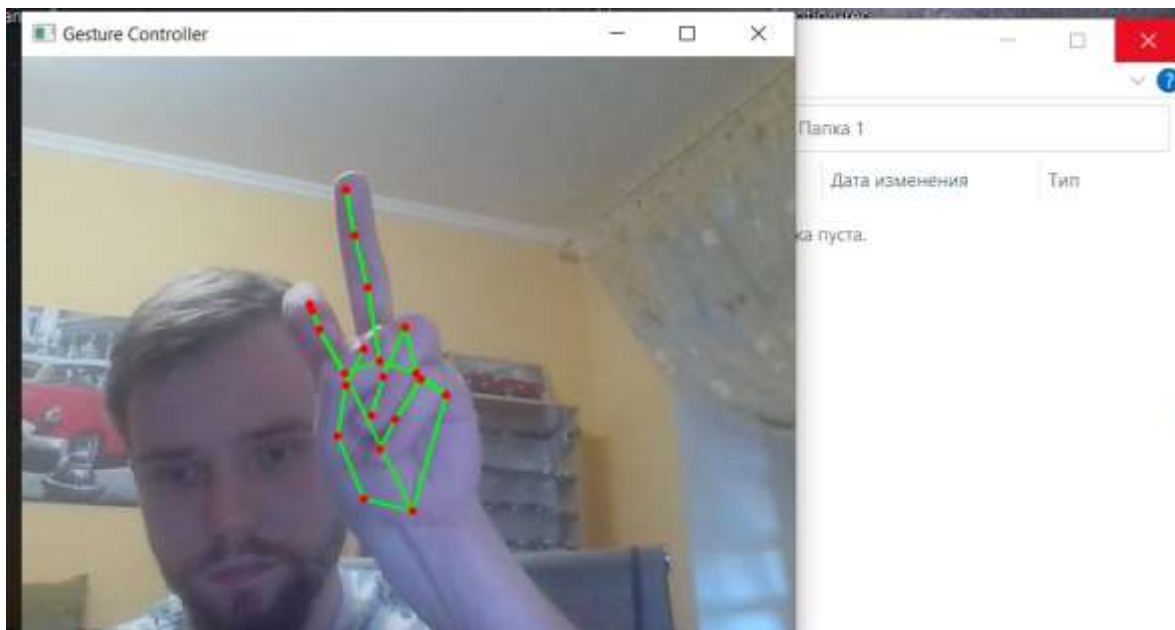


Рисунок 3.10 – Жест «натиск лівої кнопки миші»

Жест «натиск правої кнопки миші» (рис. 3.11) повторює попередній жест, тільки зі згинанням середнього пальця. Жест імітує натиск правої кнопки миші. Основна функція кнопки – відкриття контекстного меню.

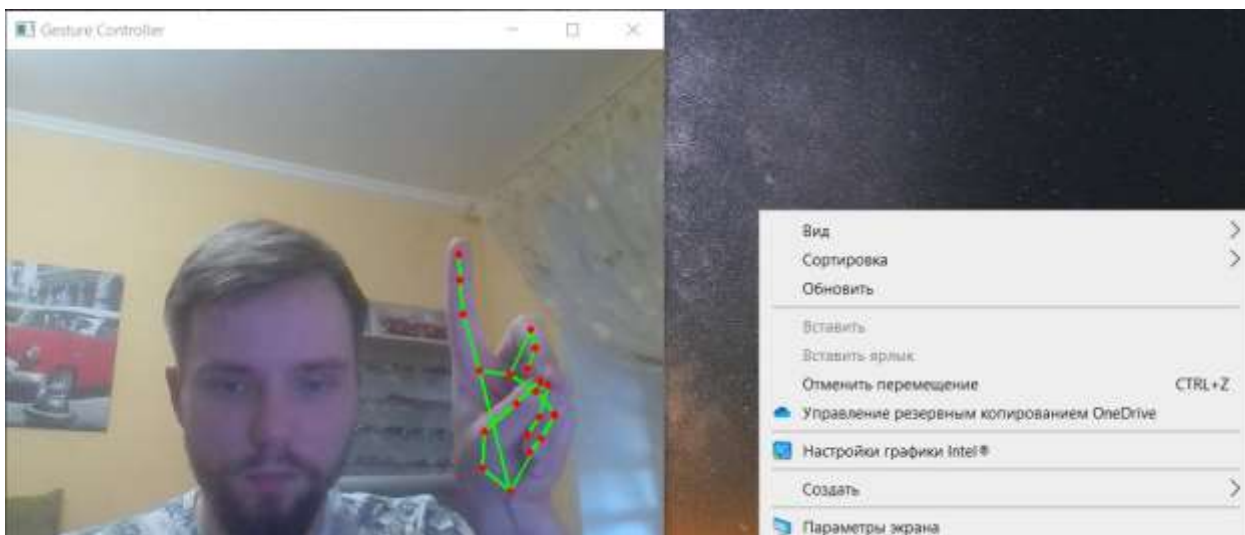


Рисунок 3.11 – Жест «натиск правої кнопки миші»

Жест «прокручування сторінки» (рис. 3.12) виконується лівою рукою та повторює дії колесика миші – прокрутка тексту, що не поміщається у виділену область вікна.

Переміщення жесту вертикально вгору переміщає текст ввєрх, а вниз переміщає сторінку с текстом вниз.

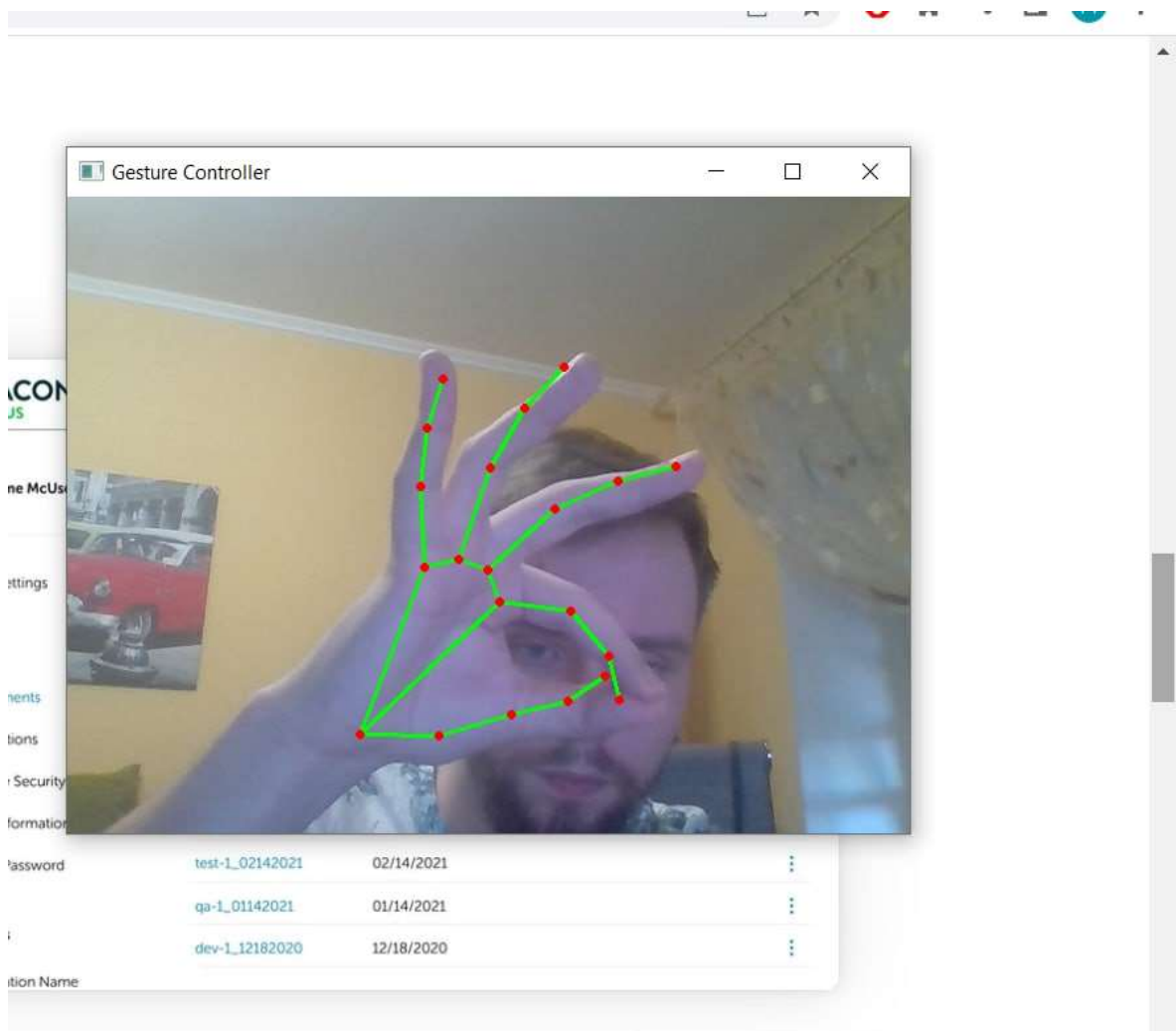


Рисунок 3.12 – Жест «прокручування сторінки»

Жест «збільшення та зменшення гучності динаміків» (рис. 3.13) повторює жест прокручування сторінки, але виконується дія правою рукою. Переміщення жесту вертикально вгору збільшує гучність, а переміщення вниз зменшує. Додаткові знімки тестування жестів наведено у Додатку А.

Якщо говорити про результати роботи, то, проаналізувавши процес детектування кінцівок рук, можна заявити наступне. В ідеальних умовах, з гарним освітленням та якісним зображенням вебкамери, програма вірно визначала рухи кінцівок та види жестів.

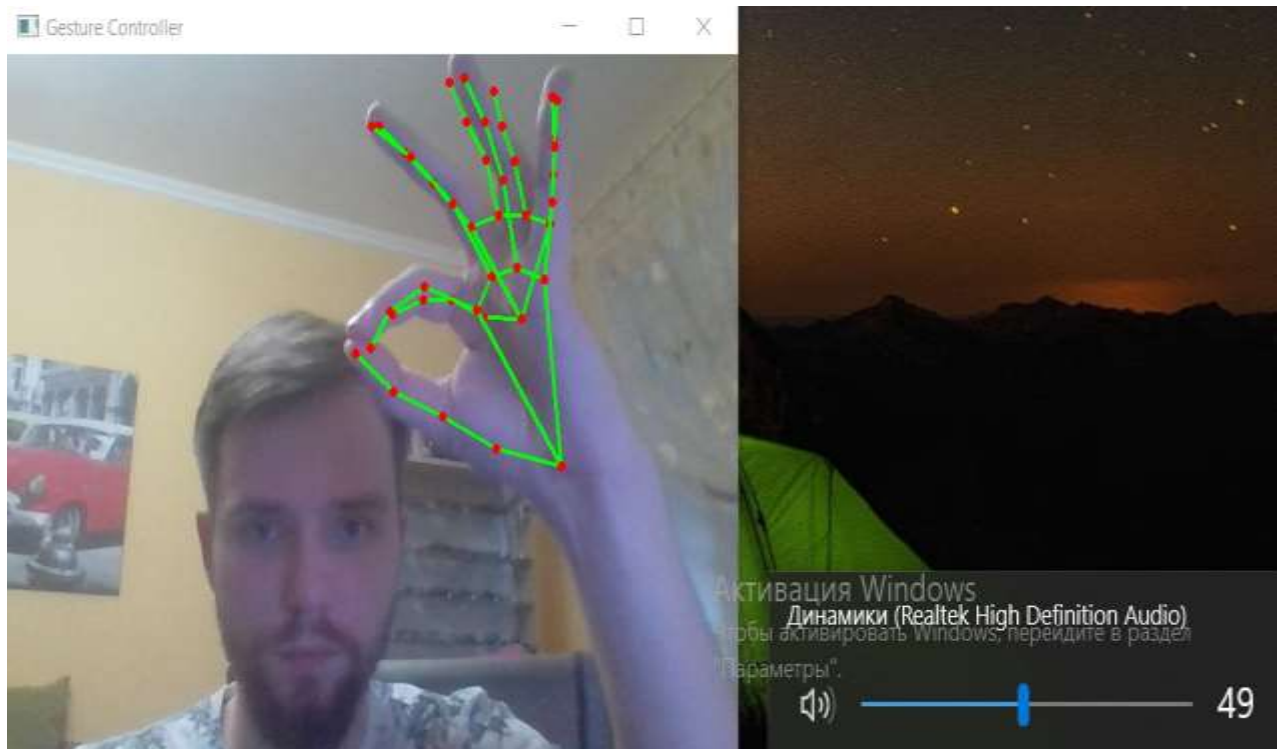


Рисунок 3.13 – Жест «збільшення та зменшення гучності динаміків»

У деяких випадках є складності роботи застосунку, зв'язані з недотриманням ідеальних умов для якісної роботи.

У процесі роботи помітно деякі фантомні детектування жестів програмою та її подальше неправильне виконання. Неправильне світло або погана камера можуть не так детектувати жести та призвести до неправильних дій у роботі [37-41].

### 3.4 Перспективи подальшої роботи

Розроблений застосунок виконує свою задачу, незважаючи на деякі неправильні дії, які потрібно виправляти у майбутньому.

У цій роботі точність та ефективність відіграють велику роль у тому, щоб застосунок був настільки корисний, як і фізична миша, тобто, виключаючи комп'ютерну мишу з життя користувача. Кожен рух фізичної миші повторюється жестами та рухами кінцівок пальців.

Тестування застосунку показало достатні результати (якість обробки та розпізнавання – 92,5%), а також виявило слабкі місця, які у майбутньому потребують більших досліджень та виправлень.

Отримані результати показують, що застосунок є перспективним і потребує розвитку у майбутньому. Детектування жестів та кінцівок рук дає багато можливостей для майбутнього розширення застосунка.

Для того, щоб застосунок був більш зручним та точним при різних умовах, необхідно внести декілька функцій та покращень, а саме:

- розумний та адаптивний рух руками: оскільки поточний процес розпізнавання обмежені в радіусі 25-30 см, необхідні функції адаптивного збільшення та зменшення масштабу, щоб збільшити відстань розпізнавання жестів, де можна автоматично регулювати детектування жестів на основі відстані між користувачем та вебкамерою;

- збільшення точності та продуктивності: час відповіді залежить від апаратного забезпечення комп'ютера, це включає швидкість обробки процесора, розмір оперативної пам'яті та доступні функції вебкамери. Тому програма може мати високу продуктивність, якщо застосувати комп'ютер з якісною вебкамерою, яка працює краще в різних типах освітлення.

## ВИСНОВКИ

У рамках кваліфікаційної роботи був розроблений і реалізований застосунок для керування курсором за допомогою жестів рук.

Виконано всі поставлені задачі, а саме:

- досліджені методи визначення точок на зображенні рук, розібрано поняття детектування кінцівок рук, надано огляд актуальних статей та досліджень по розпізнаванню жестів, розглянуто проблеми з визначення точок на зображенні рук;
- проаналізовані методи визначення кінцівок рук, визначено їх актуальність та недоліки у роботі застосунка;
- розроблено методіку керування курсором за допомогою жестів;
- визначені інструментальні засоби для створення застосунку;
- виконано всі етапи розроблення застосунку для керування комп'ютером за допомогою рук, реалізовано алгоритм роботи застосунку;
- проведено тестування застосунку, проаналізовані результати, визначено особливості реалізованого в застосунку алгоритму, його ефективність та подальший розвиток;
- визначено перспективи подальшої роботи, які виходять з отриманих результатів тестування.

Результати роботи апробовано у вигляді тез доповіді під час XIX Міжнародної науково-практичної конференції «Modern problems in science», 17-20 травня 2022 р., Ванкувер, Канада [42].

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Parada-Loira, F., Gonzalez-Agulla, E., & Alba-Castro, J. L. (2014). Hand gestures to control infotainment equipment in Cars. *2014 IEEE Intelligent Vehicles Symposium Proceedings*. URL: <https://doi.org/10.1109/ivs.2014.6856614> (дата звернення 29.04.2022).
2. Michael Hamilton, Patrick Mead, Megan Kozub, Alexander Felid (2016) Gesture Recognition Model for Robotic Systems of Military Squad Commands.
3. Гороховатський В.О., Творошенко І.С. (2022) Аналіз багатовимірних даних за описом у формі множини компонент: монографія. Харків: ХНУРЕ. 124 с.
4. Чуйков А.В., Вульфін А. М. (2017). Система розпізнавання жестов на основі нейросетевих технологій. *Вестник Уфимського державного авіаційного технічного університету*, 21 (3 (77)), с. 113-122.
5. Preetha Lakshmi, S., Aparna, S., Gokila, V., Rajalakshmi, P. (2021). Hand gesture recognition using CNN. *Smart Innovation, Systems and Technologies*, 371–382. URL: [https://doi.org/10.1007/978-981-16-3675-2\\_28](https://doi.org/10.1007/978-981-16-3675-2_28) (дата звернення 03.05.2022).
6. 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.
7. Tvoroshenko I., and Zarivchatskyi R. (2020) Analysis of existing methods for searching object in the video stream, *Abstracts of VI International Scientific and Practical Conference «About the problems of science and practice, tasks and ways to solve them» (October 26-30, 2020). Milan, Italy*, pp. 500-505.
8. Yu, X., & Yuan, Y. (2019). Hand Gesture Recognition Based on Faster-RCNN Deep Learning. *J. Comput.*, 14(2), 101-110.
9. Творошенко І.С. (2021) Технології прийняття рішень в інформаційних системах: навч. посібник. Харків: ХНУРЕ, 51 с.

10. Mahmoud, Nourelhoda & Fouad, Hassan & Soliman, Ahmed. (2020). Smart healthcare solutions using the internet of medical things for hand gesture recognition system. *Complex & Intelligent Systems*. 7. 1-12.
11. Garg, P. & Aggarwal, Naveen & Sofat, S. (2009). Vision Based Hand Gesture Recognition. World Academy of Science, *Engineering and Technology*. 49. 972-977.
12. Kaur, H.; Rani, J. (2016). A review: Study of various techniques of Hand gesture recognition. In *Proceedings of the 2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), Delhi, India*, 4–6 July 2016; pp. 1–5.
13. Sonkusare, J.S.; Chopade, N.B.; Sor, R.; Tade, S.L.(2015) A review on hand gesture recognition system. In *Proceedings of the 2015 International Conference on Computing Communication Control and Automation, Pune, India*, 26–27 February 2015; pp. 790–794.
14. Murthy, G.R.S.; Jadon, R.S.(2009) A review of vision based hand gestures recognition. *Int. J. Inf. Technol. Knowl. Manag.* 2009, 2, 405–410.
15. Lamberti, Luigi & Camastra, Francesco. (2011). Real-Time Hand Gesture Recognition Using a Color Glove. In *Proceedings of the International Conference on Image Analysis and Processing, Ravenna, Italy*, 14–16 September 2011; pp. 365–373.
16. Wang, Robert & Popovic, Jovan. (2009). Real-time hand-tracking with a color glove. *ACM Trans. Graph.* 28, 1–8
17. Hasan, Mokhtar & Mishra, Pramod. (2012). Hand Gesture Modeling and Recognition using Geometric Features: A Review. *Canadian Journal on Image Processing and Computer Vision*. 3. 12-26.
18. Tvoroshenko, I., & Kukharchuk, V. (2021). Current state of development of applications for recognition of faces in the image and frames of video captures.
19. Prakash, J. & Gautam, U.K.. (2019). Hand gesture recognition. *International Journal of Recent Technology and Engineering*. 7. 54-59.

20. Deng, Weihong & Hu, Jiani & Guo, Jun. (2006). Ada-Boost Algorithm, Classification, Naïve. *Pattern Recognition, International Conference*. 2. 699-702.
21. Marriott, A & Maurer, U. (2022). More Ada in non-Ada systems. *ACM SIGAda Ada Letters*. 41. 71-76.
22. Devineau, G.; Moutarde, F.; Xi, W.; Yang, J. (2018) Deep learning for hand gesture recognition on skeletal data. *In Proceedings of the 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), Xi'an, China, 15–19 May 2018*; pp. 106–113.
23. Ahmad M.A., Gorokhovatskyi V., Tvoroshenko I., Vlasenko N., Mustafa S.K. (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.
24. Saripalli, Joel. (2022). Hand Gestures Recognition using CNN.
25. R. Girshick, J. Donahue, T. Darrell and J. Malik, (2014). "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580-587.
26. Wu, BingZheng & Wu, HuiLing & Du, Yongzhao & Liu, PeiZhong. (2021). Automatic Recognition of Fetal Heart Standard Section Based on Fast-RCNN. 70-73.
27. Tvoroshenko I., and Gorokhovatskyi V. (2022) The Application of Hybrid Intelligence Systems for Dynamic Data Analysis, *International Journal of Engineering and Information Systems*, 6(2), pp. 40-48.
28. Daradkeh Y.I., Gorokhovatskyi V., Tvoroshenko I., and Al-Dhaifallah M. (2022) Classification of Images Based on a System of Hierarchical Features, *Computers, Materials & Continua*, 72(1), pp. 1785-1797.
29. Ubaid, Muhammad & Khalil, Malika & Khan, Muhammad Usman & Saba, Tanzila & Rehman, Amjad. (2022). Beard and Hair Detection, Segmentation and Changing Color Using Mask R-CNN.
30. Кобилін О.А., Творошенко І.С. (2021). Методи цифрової обробки зображень: навч. посібник. Харків: ХНУРЕ.

31. Гороховатський В.О., Творошенко І.С. (2021). Методи інтелектуального аналізу та оброблення даних: навч. посібник.
32. Lyashenko V., Mustafa S.K., Tvoroshenko I., and Ahmad M.A. (2020) Methods of Using Fuzzy Interval Logic During Processing of Space States of Complex Biophysical Objects, *International Journal of Emerging Trends in Engineering Research*, 8(2), pp. 372-377.
33. 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.
34. Tvoroshenko, I. (2019). Development of models of spatial analysis of status of interactive processes of complex systems.
35. 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.
36. 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.
37. 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.
38. Творошенко, І. С., Шевченко, А. Р. (2018) Удосконалення просторової мережі навчальних закладів міста Сєвєродонецька на основі геоінформаційного аналізу. *Системи обробки інформації*, (1), 46-52.
39. Cheng, Fangxiao & Yin, Borui & Chu, Ming. (2022). Research on Target Recognition Algorithm Based on Improved Faster-RCNN.

40. Творошенко, І. С., & Табашник, В. А. (2018). Розробка просторової моделі геоінформаційної підтримки людей з обмеженими можливостями, що пересуваються на інвалідних колясках, у місті Харків. *Збірник наукових праць Харківського національного університету Повітряних Сил*, (1), 122-128.

41. I. Tvoroshenko (2020). Information technologies for decision-making on the conditions of spatially distributed objects, in Abstracts of I International Scientific and Practical Conference. Problems and perspectives of modern science and practice, Austria. pp. 45-50.

42. Тарасов, Д. (2022). Аналіз існуючих програмних застосунків управління курсором за допомогою жестів рук.