


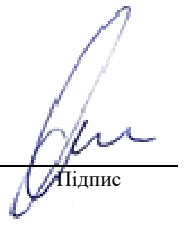


## КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА

на тему Метод визначення точності координації рухів кисті руки з використанням технології захоплення рухів

Галузь знань 12 – Інформаційні технології  
Шифр і назва галузі знань  
Спеціальність 122 – Комп'ютерні науки  
Шифр і назва спеціальності  
Освітня програма Комп'ютерні науки  
Назва освітньої програми

Виконав: студент 2 курсу, група КНм-21-1  О.О. Яницький  
Курс, група виконавця Підпис Ініціали, прізвище  
Керівник: к.т.н., доцент кафедри КН  Р.О. Багрій  
Науковий ступінь, посада Підпис Ініціали, прізвище  
Нормоконтроль: к.т.н., доцент кафедри КН  Р.О. Багрій  
Науковий ступінь, посада Підпис Ініціали, прізвище

До захисту допускаю:  
Зав. кафедри КН, д.т.н., професор  О.В. Бармак  
Підпис Ініціали, прізвище  
10 грудня 2022 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

Факультет інформаційних технологій

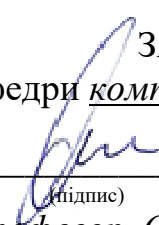
Кафедра комп'ютерних наук

Освітній ступінь магістр

Галузь знань 12 – Інформаційні технології

Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ  
Завідувач кафедри комп'ютерних наук

  
\_\_\_\_\_  
(підпис)  
д.т.н., професор О.В. Бармак  
« 01 » вересня 2022 року

**ЗАВДАННЯ  
НА КВАЛІФІКАЦІЙНУ РОБОТУ МАГІСТРА**

1. Тема кваліфікаційної роботи магістра: «Метод визначення точності координації рухів кисті руки з використанням технології захоплення рухів»

2. Завдання видано студенту Яницькому Олегу Олександровичу  
(прізвище, ім'я, по батькові)

3. Керівник роботи доцент кафедри КН Багрій Руслан Олександрович  
(прізвище, ім'я, по батькові)

4. Затверджені наказом університету від « 21 » липня 2022 р. № 83

5. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета кваліфікаційної роботи магістра – визначення точності координації рухів кисті руки з використанням технології захоплення рухів. Для досягнення поставленої мети визначено наступні задачі: провести аналіз технологій захоплення та відстеження рухів кисті руки; провести аналіз методів виявлення перетину та відстані між 2D об'єктами; розробити метод відстеження шляху пальця користувача та відображення його у 2D просторі; розробити метод визначення точності координації рухів кисті руки; реалізувати розроблений метод для перевірки визначення точності координації рухів за допомогою Leap Motion; провести валідацію розробленого метода.

## Реферат

Кваліфікаційна робота магістра присвячена розробці методу визначення точності координації рухів кисті руки з використанням технології захоплення рухів.

**Актуальність теми.** Тремор людини - одне з найпоширеніших розладів рухів, яке характеризується повторюваними та ритмічними рухами, що виникають незалежно від волі людини [1]. Походження тремору досі не з'ясовано, однак тремор може бути пов'язаний з фізіологічними явищами, такими як старіння, і з неврологічними розладами, наприклад, хворобою Паркінсона. Пацієнти страждають фізично, часто не в змозі писати та займатися мистецтвом, а також соціально, з тремтіннями, що викликають більше соціальної тривоги. На жаль, існує порівняно небагато способів для окремих осіб і лікарів швидко та надійно відстежувати прогресування тремору з часом.

В даний час умови тремору вимірюються візуально в лікарнях або в домашніх умовах без спеціального обладнання. Однак, оскільки кількість людей із хворобою Паркінсона зростає разом зі старінням населення, а також у міру того, як зростають можливості для реабілітації вдома та розвитком технологій, стає дедалі важливішим мати можливість легко отримати симптоми тремору як об'єктивні дані. Клінічна оцінка тремору може бути цінним інструментом для діагностики нервово-м'язових розладів, а також для моніторингу їх прогресування. Кількісну оцінку тремору можна використовувати для контролю введення терапевтичних препаратів і оптимізації глибокої стимуляції мозку.

Отже, незважаючи на зростаючу популярність пристроїв заснованих на розпізнаванні камерою, що слідкують за рухами людини, розробка ефективних методів оцінки точності рухів для цих пристроїв залишається не дослідженою.

**Мета і задачі роботи.** Метою кваліфікаційної роботи магістра є визначення точності координації рухів кисті руки з використанням технології захоплення рухів.

Для досягнення поставленої мети визначено наступні задачі:

- Провести аналіз технологій захоплення та відстеження рухів кисті руки.
- Провести аналіз методів виявлення перетину та відстані між 2D об'єктами.

- Розробити метод відстеження шляху пальця користувача та відображення його у 2D просторі.
- Розробити метод визначення точності координації рухів кисті руки.
- Реалізувати розроблений метод для перевірки визначення точності координації рухів за допомогою Leap Motion.
- Провести валідацію розробленого метода.

**Об'єкт дослідження** – процес відстеження рухів кисті руки в режимі реального часу та визначення точності координації цих рухів.

**Предмет дослідження** – модель кисті руки, обробка даних про положення руки у просторі в режимі реального часу та оцінка точності координації рухів кисті руки.

**Методи дослідження.** Застосовані для вирішення поставлених завдань: для відстеження руху 3D об'єктів – методи тривимірної графіки та візуалізації; для визначення точності координації рухів – математичні алгоритми виявлення зіткнень, а для реалізації інформаційної системи – методології проектування інформаційних систем та веб технології.

**Наукова новизна одержаних результатів.** Вдосконалено метод для відстеження та відображення у 2D просторі переміщення пальця руки користувача, що дозволило отримувати та обробляти дані в режимі реальному часі та обраховувати взаємодію з віртуальними об'єктами.

Вперше розроблено метод для визначення точності координації рухів кисті руки, що дало можливість визначити коефіцієнт відхилення від норми, на основі якого, можна зробити висновок щодо проблем чи захворювань користувача.

**Апробація результатів кваліфікаційної роботи магістра та публікації.** Основні наукові та практичні результати опубліковані у Збірник наукових праць за матеріалами XIV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2022». - Хмельницький, 2022. - С. 261-264. темою роботи: Яницький О.О., Багрій Р.О., Скрипник Т.К. визначення точності координації рухів кисті руки у просторі.

**Структура та обсяг роботи.** Дипломна робота магістра складається з завдання, реферату, змісту, переліку скорочень, вступу, 4 розділів, висновків, переліку посилань із 17 найменувань та 4 додатків. Загальний обсяг дипломної роботи магістра становить 75 сторінок, з них 71 сторінок основного тексту та 30 сторінка додатків. У роботі наведено 47 рисунків.

**Ключові слова:** захоплення рухів, маркерні, координація рухів, безмаркерні, метод, спіраль.

## Зміст

Перелік скорочень .....	4
Вступ .....	5
<b>РОЗДІЛ 1</b>	
Аналіз сучасного стану використання інформаційних технологій для визначення точності координації рухів кисті руки у просторі .....	8
1.1. Аналіз предметної області.....	8
1.1.1 Огляд безконтактних технологій визначення точності координації рухів....	9
1.2 Аналіз технології захоплення та відстеження рухів кисті руки.....	11
1.2.1 Маркерні технології.....	12
1.2.2 Безмаркерні технології .....	15
1.2.3 Leap Motion.....	17
1.3 Аналіз методів виявлення перетину та відстані між 2D об'єктами. ....	18
1.4 Постановка задачі.....	23
Висновки до розділу 1 .....	24
Розробка інформаційної технології для визначення точності координації рухів кисті руки у просторі .....	25
<b>РОЗДІЛ 2</b>	
2.1 Дослідження прикладного програмного інтерфейсу Leap Motion .....	25
2.2 Розробка методу відстеження шляху пальця користувача та відображення його у 2D просторі. ....	29
2.3 Розробка методу визначення точності координації рухів кисті руки. ....	35
Висновки до розділу 2 .....	41
<b>РОЗДІЛ 3</b>	
Проектування інформаційної системи визначення точності координації рухів кисті руки у просторі. ....	42
3.1 Основні вимоги для проектування інформаційної системи .....	42
3.2 Функціональна модель інформаційної системи.....	43

3.3 Об'єктно орієнтована модель інформаційної системи .....	45
3.4 Розробка структури інформаційної системи .....	48
Висновки до розділу 3 .....	53
РОЗДІЛ 4	
Дослідження методу визначення точності координації рухів кисті руки .....	55
4.1 Архітектура інформаційної системи .....	55
4.2 Розробка прикладних компонентів інформаційної системи .....	57
4.3 Прикладне тестування інформаційної системи .....	62
4.4 Функціональне дослідження інформаційної системи .....	67
Висновки до розділу 4 .....	74
Загальні висновки .....	75
Перелік посилань .....	76
Додатки	

### Перелік скорочень

Скорочення, термін, позначення	Пояснення
IS	Інформаційна система
IT	Інформаційні технології
HCI	Людино-комп'ютерний інтерфейс
WIMP	Взаємодія з комп'ютером на базі елементів "windows, icons, menus, pointers"
LMC	Контролер Leap Motion
UI	Користувальницький інтерфейс
EOM	Електронна обчислювальна машина
XML	eXtensible Application Markup Language
TCP	Точка керування інструментом
LeapAPI	Програмний інтерфейс пристрою Leap Motion
API	
GUI	Графічний інтерфейс користувача

## Вступ

**Актуальність теми.** Тремор людини - одне з найпоширеніших розладів рухів, яке характеризується повторюваними та ритмічними рухами, що виникають незалежно від волі людини [1]. Походження тремору досі не з'ясовано, однак тремор може бути пов'язаний з фізіологічними явищами, такими як старіння, і з неврологічними розладами, наприклад, хворобою Паркінсона. Пацієнти страждають фізично, часто не в змозі писати та займатися мистецтвом, а також соціально, з тремтіннями, що викликають більше соціальної тривоги. На жаль, існує порівняно небагато способів для окремих осіб і лікарів швидко та надійно відстежувати прогресування тремору з часом.

В даний час умови тремору вимірюються візуально в лікарнях або в домашніх умовах без спеціального обладнання. Однак, оскільки кількість людей із хворобою Паркінсона зростає разом зі старінням населення, а також у міру того, як зростають можливості для реабілітації вдома та розвитком технологій, стає дедалі важливішим мати можливість легко отримати симптоми тремору як об'єктивні дані. Клінічна оцінка тремору може бути цінним інструментом для діагностики нервово-м'язових розладів, а також для моніторингу їх прогресування. Кількісну оцінку тремору можна використовувати для контролю введення терапевтичних препаратів і оптимізації глибокої стимуляції мозку.

Отже, незважаючи на зростаючу популярність пристроїв заснованих на розпізнаванні камерою, що слідкують за рухами людини, розробка ефективних методів оцінки точності рухів для цих пристроїв залишається не дослідженою.

**Мета і задачі роботи.** Метою кваліфікаційної роботи магістра є визначення точності координації рухів кисті руки з використанням технології захоплення рухів.

Для досягнення поставленої мети визначено наступні задачі:

- Провести аналіз технологій захоплення та відстеження рухів кисті руки.
- Провести аналіз методів виявлення перетину та відстані між 2D об'єктами.
- Розробити метод відстеження шляху пальця користувача та відображення його у 2D просторі.

- Розробити метод визначення точності координації рухів кисті руки.
- Реалізувати розроблений метод для перевірки визначення точності координації рухів за допомогою Leap Motion.
- Провести валідацію розробленого метода.

**Об’єкт дослідження** – процес відстеження рухів кисті руки в режимі реального часу та визначення точності координації цих рухів.

**Предмет дослідження** – модель кисті руки, обробка даних про положення руки у просторі в режимі реального часу та оцінка точності координації рухів кисті руки.

**Методи дослідження.** Застосовані для вирішення поставлених завдань: для відстеження руху 3D об’єктів – методи тривимірної графіки та візуалізації; для визначення точності координації рухів – математичні алгоритми виявлення зіткнень, а для реалізації інформаційної системи – методології проектування інформаційних систем та веб технології.

**Наукова новизна одержаних результатів.** Вдосконалено метод для відстеження та відображення у 2D просторі переміщення пальця руки користувача, що дозволило отримувати та обробляти дані в режимі реальному часі та обраховувати взаємодію з віртуальними об’єктами.

Вперше розроблено метод для визначення точності координації рухів кисті руки, що дало можливість визначити коефіцієнт відхилення від норми, на основі якого, можна зробити висновок щодо проблем чи захворювань користувача.

**Апробація результатів кваліфікаційної роботи магістра та публікації.** Основні наукові та практичні результати опубліковані у Збірник наукових праць за матеріалами XIV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп’ютерних наук АПКН-2022». - Хмельницький, 2022. - С. 261-264. темою роботи: Яницький О.О., Багрій Р.О., Скрипник Т.К. визначення точності координації рухів кисті руки у просторі.

**Структура та обсяг роботи.** Дипломна робота магістра складається з завдання, реферату, змісту, переліку скорочень, вступу, 4 розділів, висновків,

переліку посилань із 17 найменувань та 4 додатків. Загальний обсяг дипломної роботи магістра становить 75 сторінок, з них 71 сторінок основного тексту та 30 сторінка додатків. У роботі наведено 47 рисунків.

## **РОЗДІЛ 1**

### **Аналіз сучасного стану використання інформаційних технологій для визначення точності координації рухів кисті руки у просторі**

#### **1.1. Аналіз предметної області**

Діагноз проблем пов'язаних з розладами моторики рук виявився складним завданням для лікарів. В даний час діагноз, як і раніше, ґрунтується на спостереженні за клінічними ознаками пацієнта, такими як тремор у спокої або ригідність з подальшим спостереженням набору допоміжних критеріїв та відсутністю набору критеріїв виключення. Крім того, тяжкість і прогрес захворювання характеризуються з використанням єдиної шкали оцінювання «товариства рухових розладів» (MDS-UPDRS), яка також заснована на спостереженні клінічних ознак. Ці типи оцінок є суб'єктивними і схильні забувати про невеликі зміни рухової функції, які є важливими для захворювання, що створює проблеми для моніторингу [1]. Автоматизована діагностична платформа за місцем надання медичної допомоги, яку можна легко масштабувати в клінічних та польових умовах, може суттєво розширити можливості моніторингу у всьому світі та виявити хворобу раніше, що призведе до покращення результатів лікування пацієнтів.

В останні роки технології смартфонів та планшетів дозволили розробити високопродуктивні інструменти для моніторингу хвороби Паркінсона. Автоматизація забезпечує кілька явних переваг, коли йдеться про діагностику захворювань, включаючи низьку вартість використання, підвищену швидкість та доступність [2]. Інструмент автоматизованого моніторингу може надати базову інформацію для неврологів та виявити потенційних пацієнтів із проблемами до того, як симптоми приведуть їх до невролога. Це особливо корисно, оскільки в усьому світі зростає нестача неврологів.

Дослідники продемонстрували великий потенціал оцифрованих тестів малювання спіралі Архімеда у виявленні рухових дисфункцій, пов'язаних із тремором людини [3]. Ці тести зосереджені на вимірюваннях кінематики та тиску у

часових рядах. У дослідженнях були розроблені методи для виявлення таких характеристик, як мінливість ширини спіралі, загальна форма, гладкість спіралі, швидкість пера та тиск пера. Досягнення в галузі цифрового спірального малювання, комп'ютеризованого аналізу ознак, проектування завдань малювання та складних методів машинного навчання забезпечили більш широкі діагностичні можливості, досягнувши вражаючої точності класифікації. Продовжувалися дослідження щодо уточнення, доповнення та стандартизації оцінок малювання спіралі Архімеда, вивчення нових інформативних функцій проблем із тремором [4].

Незважаючи на те, що в існуючих дослідженнях проводилася автоматизація аналізу спіральних малюнків у пацієнтів із тремором, не завжди використання спіралі Архімеда було раціоналізовано. Як наслідок, у багатьох роботах нехтують унікальними особливостями спіралі Архімеда, які роблять її потужною для діагностики.

### **1.1.1 Огляд безконтактних технологій визначення точності координації рухів**

Людський тремор є проблемою охорони здоров'я, з якою зіштовхується весь світ. Витрати, пов'язані з медичними та соціальними аспектами, необхідними для діагностики та лікування тремору, постійно зростали протягом останніх десятиліть.

Клінічна оцінка пацієнтів з патологічним тремором зазвичай базується на картинах, отриманих шляхом спостереження за групами аналізу [5]. Для кожного захворювання існує свій стандарт оцінки пацієнтів. Клінічна оцінка не може дати багато відповідей щодо еволюції захворювання, оскільки вона не враховує особливості кожного пацієнта та використовує суб'єктивність експертів під час оцінки та класифікації осіб.

Існує кілька способів вимірювання людського тремору, однак навіть сьогодні найбільш найпоширенішими є методи, які передбачають використання індикатора ступеня тяжкості. У цих методах пацієнтів просять виконати різні малюнки, такі як

спіралі, кола та літери. Згодом неврологи класифікують ці малюнки за цифровою шкалою, яка зазвичай варіюється від 0 (без видимого тремору) до 5 (важкий тремор, що призводить до інвалідності). Потім малюнки, зроблені пацієнтами, порівнюються з прикладами попередніх досліджень, проведених з урахуванням інших пацієнтів, раніше класифікованих. Тому цей тип класифікації полягає у візуальному порівнянні та містить суб'єктивність експерта, відповідального за аналіз. Крім того, цей аналіз запобігає вилученню важливої інформації з активності тремору, такої як частота, амплітуда та швидкість.

Візерунки намальовані від руки зазвичай оцінюють за допомогою візуальних рейтингових шкал. Однак такі шкали забезпечують лише грубі суб'єктивні оцінки амплітуди тремору. Щоб зменшити суб'єктивність і обмеження деяких методів, заснованих на візуальних шкалах, було розроблено кілька стратегій електронного вимірювання тремору, а саме застосування таких приладів як акселерометр та цифрового планшету.

Використання цифрових планшетів є поширеним і забезпечує можливість виявлення активності тремору в кінетичних умовах. Звичайною функцією планшета є можливість аналізу креслень безпосередньо на комп'ютері. Вимірювання тремору за допомогою цифрових планшетів є альтернативою виявлення тремору, що не потребують введення в організм інструментів, яка поєднує в собі простоту з точністю та універсальністю обчислювальних методів. Планшет здатний повідомляти про положення кінчика пера на його поверхні. Використовуючи цю властивість, цей пристрій може виявляти рух предмета за стандартними малюнками, розміщеними на ньому.

Проаналізувавши наведені роботи варто відзначити, проблема з використанням таких методів полягає в тому, що форма руки суб'єкта обмежена через необхідність тримати або носити вимірювальний пристрій, окрім того потрібен експерт для оцінювання отриманих даних.

Отже, виникає необхідність в використанні пристрою який розпізнавав би рух на оброблених зображеннях камери та не обмежував рухи користувача, а також

використання методу, який не потребував оцінки експерта, або ж його присутність була мінімальною.

У наступному розділі дається короткий огляд технології захоплення та відстеження рухів, що надають дані, потрібні для застосування метода в оцінці тремору користувача

## **1.2 Аналіз технології захоплення та відстеження рухів кисті руки.**

Жести та вирази обличчя повсякденно використовуються людьми для взаємодії між собою, тоді як взаємодія людини з комп'ютером все ще потребує розуміння та аналізу сигналів для інтерпретації бажаної команди, що зробило взаємодію складною та неприродною. Останнім часом розробка спеціальних пристроїв введення свідчить про велику увагу в цій галузі для полегшення взаємодії між людьми та комп'ютерами, а також для здійснення більш складної взаємодії з комп'ютером. Поєднання традиційних пристроїв, миші та клавіатури, з новими розробленими пристроями взаємодії, такими як розпізнавання жестів і обличчя, тактильні датчики та пристрої відстеження забезпечує гнучкість дистанційного керування, розпізнавання жестів, віртуальної реальності та відеоігор [6]. Жест розглядається як природний спосіб спілкування між людьми, особливо з вадами слуху, це фізичний рух тіла, рук або пальців, який передає повідомлення, і система що розпізнає жести, використовується для інтерпретації та пояснення цього руху як команди яка зрозуміла комп'ютеру.

З останнім прогресом споживчих технологій віртуальної реальності і доповненої реальності можливість взаємодії з цифровим світом за допомогою рук набуває більшого значення. Рука є одним із найскладніших і найкрасивіших елементів природної інженерії в людському тілі що дозволяє нам маніпулювати маленькими предметами з великою точністю, але в порівнянні з іншими частинами тіла, відстежування рук та пальців є дуже складним, руки набагато менші, мають більше можливостей для згину та перекривають один одного, що призводить до

збоїв у відстеженні. Існує дві основні технології захоплення рухів: безмаркерна та маркерна система захоплення рухів.

### 1.2.1 Маркерні технології

Маркерна система використовує спеціальний костюм із датчиками, кожен рух людини відстежується камерами, які отримують дані з датчиків, після чого ця інформація надходить до комп'ютера, де в свою чергу все зводиться в єдину тривимірну модель, детально відтворюючи кожні рухи, на основі яких пізніше або в режимі реального часу створюється анімація 3D моделі (рисунок 1.1). Також цим методом відтворюється міміка актора, у цьому випадку на його обличчі розташовані маркери, які дозволяють фіксувати основні мімічні рухи. Маркерна технологія існує набагато довше, саме тому на теперішній час існує дуже велика кількість похідних систем захоплення рухів, розрізняються вони лише у принципі передачі рухів [7].

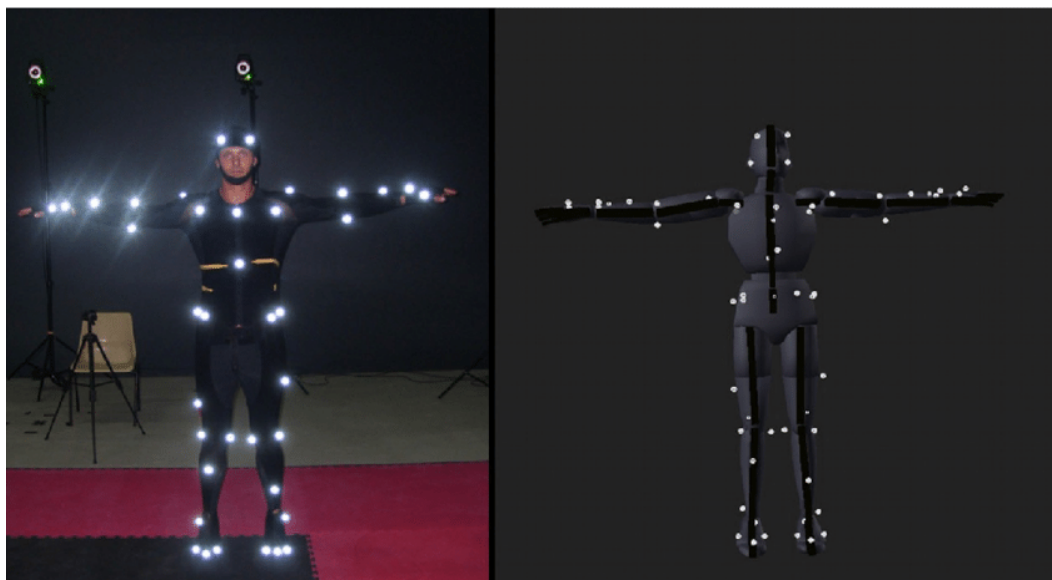


Рисунок 1.1 – Спеціальний костюм для відстеження рухів на базі маркерної технології

Оптично пасивні системи використовують дані, отримані від датчиків зображення для триангуляції 3D - положення суб'єкта між двома або більше

камерами, відкаліброваних, щоб забезпечити перекриття виступів. Збір даних зазвичай реалізується з допомогою спеціальних маркерів, прикріплених до актора. Відстеження великої кількості виконавців або розширення зони захоплення досягається за рахунок додавання більшої кількості камер. Мінус такого методу в тому, що багато часу займає розміщення маркерів на людині і при швидкому русі або ж близькому розташування система може їх плутати.

При оптично активній технології на актора одягають не маркери, які відбивають світло, а світлодіоди, які, навпаки, самі його створюють. Кожен світлодіод має свій ідентифікатор, що дозволяє розпізнавати їхнє положення. Мінусами є те, що таким способом неможливо захоплювати міміку людини, світлодіоди є досить крихкими і, крім того, потрібно підключати додатковий контролер для їх зчитування, що в деяких місцях сковує рухи актора

В магнітній системи замість маркерів використовують магніти, які створюють магнітні потоки (рисунок 1.2). При русі актора з магнітами ці потоки спотворюються, що дозволяє ресиверам, у вигляді камер вирахувати місце розташування магнітів, цим створивши анімацію. До мінусів можна віднести, що різні електроприлади можуть створювати магнітні та електричні перешкоди. У ресиверів більша чутливість до положення магнітів, через це потрібна більша робоча зона, а також відсутня можливість захоплення рухів обличчя.



Рисунок 1.2 – Рукавиці на магнітному трекінгу

Механічна система для роботи якої на суглоби актора надягається механічний скелет, який повністю копіює всі його рухи (рисунок 1.3). Після чого в комп'ютер передаються дані про всі згини суглобів. Мінуси полягають у тому, що механічний скелет найбільше обмежує рухи актора, крім згинів дана система не може захоплювати будь-які інші рухи. Також незважаючи на металевий каркас при необережному використанні, можна пошкодити його.



Рисунок 1.3 – Механічний скелет

Інерційні системи для збору даних про рух використовують гіроскопи та інерційні рецептори (рисунок 1.4). Ця система відрізняється з інших тим, що визначає як становище гіроскопа у просторі, так і його кут нахилу. Мінуси полягають у тому, що система не може захоплювати міміку, так само потрібно прикріплювати додатковий контролер.

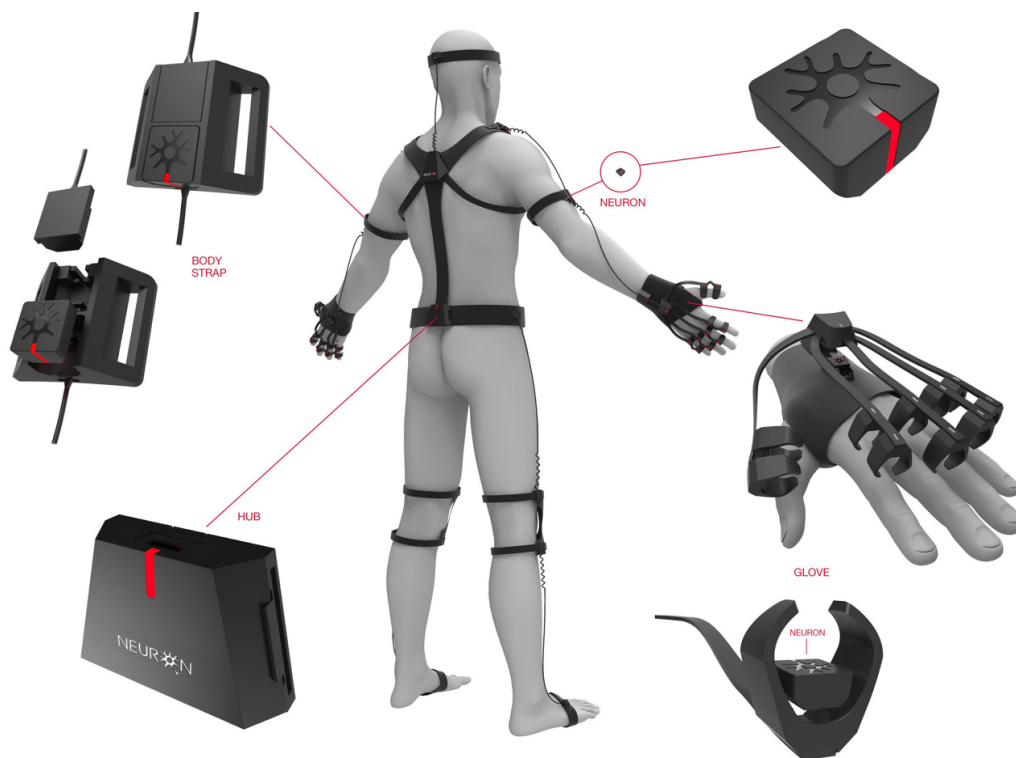


Рисунок 1.4 – Механічний скелет

### 1.2.2 Безмаркерні технології

Маркери є найбільш оптимальним рішенням для вирішення проблеми розрахунку пози і не вимагають потужних пристроїв. Його простота, надійність і ефективність мають велику перевагу. З іншого боку, безмаркерна технологія виявляє об'єкти або характерні точки сцени без попереднього знання навколишнього середовища. Цей механізм складніше реалізувати, оскільки він реалізує алгоритми, які є дорогими з точки зору часу обчислень. Поява та вдосконалення нових пристроїв дозволили використати цю технологію та відмовитися від присутності маркера в сцені та перейти до безмаркерних методів.

Нові методи та дослідження у галузі комп'ютерного зору призводять до швидкого розвитку безмаркерного підходу до захоплення руху. Ці системи не вимагають носити спеціальне обладнання для відстеження. Спеціальні алгоритми комп'ютера розроблені таким чином, щоб дозволити системі аналізувати кілька потоків оптичного входу та ідентифікувати людські форми, розбиваючи їх на

складові для відстеження. Вважають, що безмаркерні системи з технічної точки зору прогресивніші, оскільки технології, що лежать в їх основі, складніші і наукомісткі. Так як комп'ютеру необхідно відрізнити праву сторону від лівого персонажа, а блискучі поверхні може збити безмаркерну систему, вони вимагають повноцінного машинного зору. Останнім часом найбільш популярними для захоплення та відстеження рук стали прилади які використовують безмаркерну технологію з додатковими датчиками, такі як Leap Motion, датчики Time-of-Flight, і датчики глибини Kinect та інші.

Kinect - сенсорний безконтактний ігровий контролер (рисунок 1.5), спочатку створений компанією Microsoft для консолі Xbox 360, а пізніше і для персональних комп'ютерів під ОС Windows.



Рисунок 1.5 – Прилад Kinect

Kinect дозволяє користувачеві взаємодіяти з ігровою приставкою без застосування контактного ігрового контролера за допомогою розпізнавання усних команд, поз тіла і об'єктів або малюнків, що показуються. Контролер включає два датчики глибини, кольорову відеокамеру і мікрофонні ґрати. Спеціалізоване ПЗ здійснює повне тривимірне розпізнавання рухів тіла, міміки обличчя та голосу. Інфрачервоний проектор, об'єднаний з монохромною КМОП-матрицею, є датчиком

глибини, це дає Kinect можливість отримувати 3D-зображення при будь-якому природному освітленні. Сенсор автоматично калібрується, враховуючи навколишні умови, діапазон глибини та програма проєкту дозволяє це зробити.

### 1.2.3 Leap Motion

Leap motion - пристрій безмаркерного захоплення рухів рук (рисунок 1.6), що складається з пари інфрачервоних камер та світлодіодів, що працюють в інфрачервоному діапазоні та підсвічують робочу зону. Камери відстежують рух рук у робочій ділянці простору (область тривимірної взаємодії умовно маючи форму віртуального куба з стороною в 1м). Дані передаються на ПК, до якого підключено контролер із інтерфейсом USB. Ухвалена інформація обробляється за допомогою спеціалізованого ПЗ, результат виробленої дії відображається на екрані монітора [8].

LMC розпізнає зап'ястя, долоню та пальці двох рук окремо. Ця технологія дозволяє займатися творчістю, організовувати фізику взаємодії аватару рук з предметами у віртуальному оточенні.



Рисунок 1.6 – Прилад Leap motion

Контролер відстежує рухи пальців з точністю до 0,01 мм. LMC має перевагу у мобільності за рахунок мінімальних габаритів перед Kinect та іншими приладами. Пристрій здатний розпізнавати рух пальців більш точно порівняно з Kinect, що може бути використане у застосуванні до діагностики для оцінки тремору рук хворих з порушеннями рівноваги.

В ідеї технології LMC закладено можливість переходу на принципово новий рівень управління комп'ютером та роботи у різних програмах та іграх. Ця розробка рано чи пізно приведе людство до повної відмови спочатку від вказівних пристроїв таких як мишки і джойстиків, а згодом, можливо, і сенсорних екранів. У першому випадку ми отримуємо новий і більш природний для нас спосіб взаємодії з об'єктами та відмовляємось від зайвих периферійних пристроїв. У другому випадку, коли ми можемо відмовитися від сенсорних пристроїв, це пов'язано з тим, що приладу не має значення, чи знаходяться ваші руки в рукавичках, його не лякає бруд, волога, жир та інше.

Можливо, на практиці LMC працює поки не так ідеально, як вона описується, але головний крок вже зроблено. Ця технологія вже є однією з найкращих для захоплення та відстеження рук, навіть якщо навіть вона в майбутньому не стане широко використовуваною, то на її базі буде створено багато таких пристроїв нового покоління. У разі активне впровадження технологій захоплення рухів у комп'ютерні ігри, а й у наше повсякденне життя стане неминучим.

Після розгляду технологій та приладів для захоплення рухів, було обрано пристрій на основі безмаркерної технології Leap Motion за рахунок його простоти, компактності, а також гарантії знаходження потрібного рішення.

### **1.3 Аналіз методів виявлення перетину та відстані між 2D об'єктами.**

Завдання виявлення зіткнень виникає у системах автоматизованого проектування, у робототехніці, у системах віртуальної реальності та відеоіграх

(рисунок 1.7). У найпростішому випадку потрібно визначити, чи перетинаються задані об'єкти у просторі. Часто потрібна інформація про те, коли і де сталося зіткнення. Наприклад, при моделюванні механічної системи може знадобитися список точок контакту, нормалі до поверхонь та глибина взаємопроникнення в точках контакту.

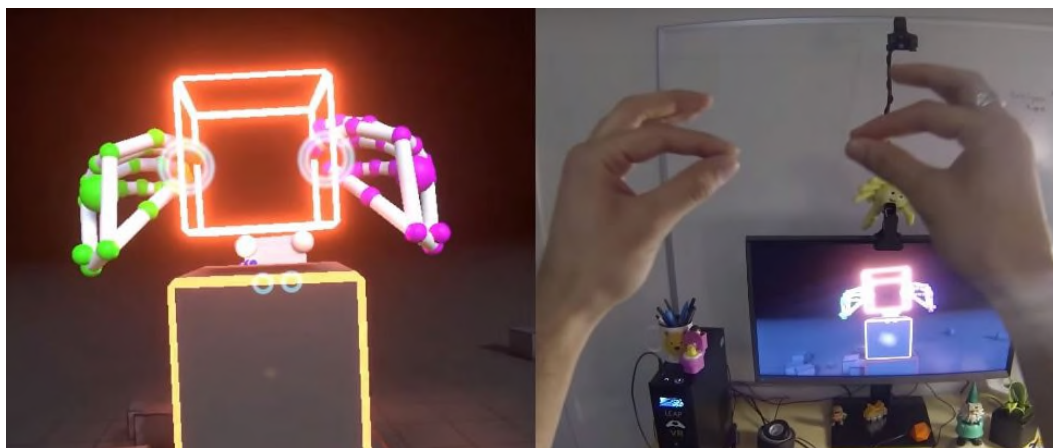


Рисунок 1.7 – Виявлення зіткнень для Leap Motion у віртуальній реальності

Виявлення зіткнень є одним із найчастіше використовуваних термінів для виявлення чи розпізнавання ударів чи зіткнень об'єктів на екрані. Це велика тема, і немає єдиної техніки чи теореми включення чи візуалізації всіх типів зіткнень. Термін виявлення зіткнень широко використовується в програмуванні ігор, тоді як тестування влучення – це термінологія, яка використовується в програмуванні інтерфейсу користувача.

В даний час відомо безліч алгоритмів виявлення зіткнень, більшість з яких обмежуються роботою з опуклими об'єктами, а нескладні неопуклі об'єкти зазвичай розбиваються на опуклі примітиви. Зараз найбільшого поширення набули геометричні методи виявлення зіткнень, засновані на використанні обмежувальних об'ємів [9].

При виявленні зіткнень двома важливими аспектами є оперативність та точність. Вимоги до виявлення зіткнень:

- перевірка на наявність зіткнення;

- позиція виявленого зіткнення;
- відстань між об'єктами;
- прогнозування зіткнення наступного разу.

Алгоритми виявлення зіткнень можна класифікувати за типом подання вхідних даних та за типом розпізнавання зіткнень. Найчастіше зустрічаються вхідні дані наступних типів:

- сітки з трикутників;
- конструктивна блочна геометрія;
- неявно задана геометрія.

Використання сітки з трикутників обумовлено тим, що трикутник є базовим геометричним примітивом для відображення на екрані складних об'єктів. Конструктивна блокова геометрія часто використовується в САПР (Система автоматизованого проектування). Комбінуючи прості базові тривимірні об'єкти, такі як сфери, прямокутні паралелепіпеди, конуси, на виході отримуємо складний об'єкт. Неявне завдання об'єкта передбачає завдання форми об'єкта як математичного рівняння. Очевидно, що цей спосіб занадто громіздкий і не підходить для складних об'єктів. Якщо ж не зазначено явно, передбачається, що для вхідних даних для алгоритму виявлення зіткнень використовуються полігональні сітки з трикутників.

Перевірка на зіткнення двох об'єктів може бути складним завданням, що включає перевірки між безліччю граней кожного об'єкта. Для швидкого виконання завдання часто використовуються спрощені форми для представлення кожного об'єкта, що дозволяє проводити швидкі тести на зіткнення. Ці тести, звичайно, не мають достатньої точності, і часто вони використовуються як швидкий тест, щоб визначити, чи потрібна подальша перевірка. Розглянемо три такі обмежувальні об'єми (рисунки 1.8):

- коло, що обмежує;
- обмежуючий прямокутник, вирівняний по координатних осях;
- орієнтовні обмежувальні рамки;

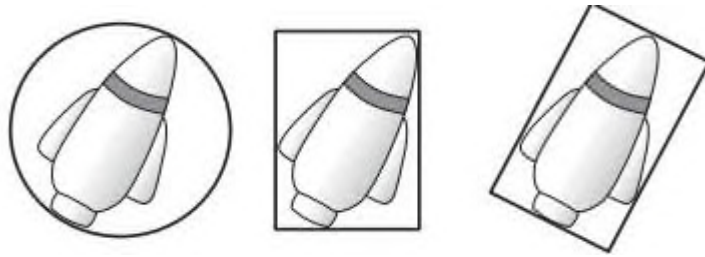


Рисунок 1.8 – Види обмежувальних об'ємів

Обмежуюче коло використовується як один із типів обмежуючого об'єму при визначенні зіткнень [10]. При застосуванні даного методу об'єкт повністю знаходиться всередині даного кола, і зіткнення розраховуються від поверхні кола, а не від поверхні об'єкта, що міститься в ньому (рисунок 1.9). Використання обмежуючого кола для виявлення зіткнень є найпростішим, швидким і грубим методом.

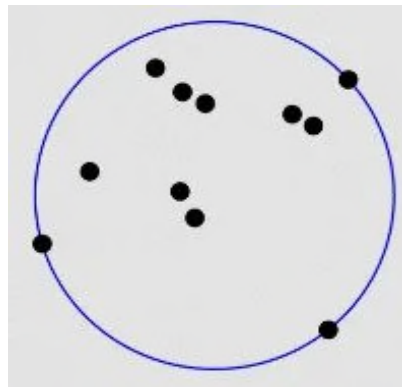


Рисунок 1.9 – Обмежуюче коло

Обмежуючий прямокутник, вирівняний по координатних осях (Axis Aligned Bounding Box, AABB) – це прямокутник, чотири сторони якого вирівняні щодо системи координат, де він знаходиться (рисунок 1.10). Це означає, що прямокутник не може обертатися і завжди знаходиться під кутом 90 градусів (зазвичай вирівняний щодо екрана). AABB - очевидний і простий метод для реалізації та корисний в іграх, де мало об'єктів, які можуть зіткнутися. Однак, якщо потрібно реалізувати цей метод зіткнень у грі, в якій присутня величезна кількість об'єктів, що зіштовхуються, то він стане занадто затратним у обчислювальному відношенні, і тому доведеться шукати способи оптимізації.

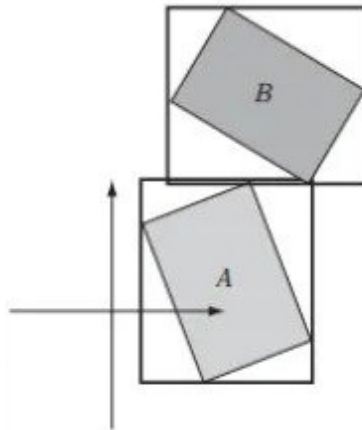


Рисунок 1.10 – Обмежуючий прямокутник вирівняний за координатними осями

Щоб відрізнити загальний випадок від AABB, довільний обмежуючий прямокутник, іноді називають орієнтованою обмежувальною рамкою (Oriented Bounding Box, OBB), де використовується локальна система координат існуючого об'єкта. AABB набагато простіше перевірити на перетин, ніж OBB, але має недолік – AABB не обертається, OBB можна обертати (рисунок 1.11). Основна ідея методу полягає у використанні простої геометрії замість складних геометричних новинок.

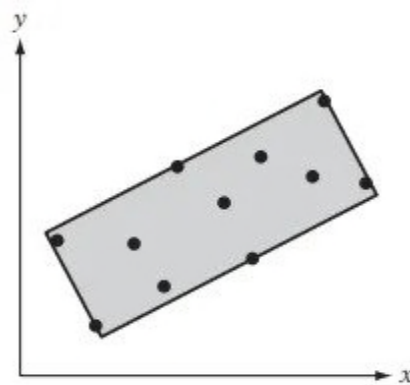


Рисунок. 1.11 – Орієнтовні обмежувальні рамки

Обмежувальні об'єми - це прості геометричні форми, всередину яких вписуються один або кілька об'єктів більшої геометричної складності. Найчастіше

використовуються сфери і прямокутники. Якщо потрібна справді щільна посадка, можна використовувати об'ємні плити або опуклі корпуси. При використанні обмежувальних об'ємів з більш щільною посадкою ймовірність передчасного відхилення збільшується, але водночас тест стає більш затратним, а вимоги до пам'яті зростають. Зазвичай всі обчислення виконуються на етапі попередньої обробки і, при необхідності, перетворюються за допомогою обмежених об'єктів під час виконання, щоб відповідати рухам об'єктів.

Отже, зважаючи на всі плюси та мінуси методів виявлення перетину та відстані, обрано метод обмежувального кола, так як він є найшвидшим, що дозволить застосовувати його в реальному часі при відстежуванні рухів без затримок.

#### **1.4 Постановка задачі**

Метою кваліфікаційної роботи магістра є створення методу відстеження рухів користувача та в подальшому оцінки точності координації цих рухів за допомогою пристрою Leap motion. Тому для виконання поставленого завдання потрібно:

- Провести аналіз технології захоплення та відстеження рухів кисті руки.
- Провести аналіз методів виявлення перетину та відстані між 2D об'єктами.
- Розробити метод відстеження шлях пальця користувача та відображення його у 2D просторі.
- Розробити метод визначення точності координації рухів кисті руки.
- Реалізувати веб-додаток для перевірки точності координації рухів за допомогою Leap Motion. Провести експериментальне тестування.

Вдалим виконанням завдання буде висока ефективність метода при визначенні точності координації рухів у користувача.

## Висновки до розділу 1

Після опрацювання першого розділу було сформовано мету роботи, завдання дослідження та доведено актуальність роботи. Проведено аналіз предметної області, де визначено проблемні моменти під час оцінки координації, та один із основних методів для тестування, а саме спіраль Архімеда. Розглянуто існуючі статі та публікації, з яких виділено плюси, мінуси та основні вимоги до пристроїв та методів для визначення точності рухів.

Також був виконаний аналіз існуючих технологій захоплення та відстеження рухів, та прилади на базі цих технологій. Проаналізувавши їх, обрано найбільш відповідний для виконання поставлених задач пристрій Leap motion.

Розглянуто методи виявлення перетину та відстані між 2D об'єктами, та обрано метод обмежувального кола, який дозволяє в реальному часі без затримок виявляти перетини між об'єктами під час відстежування рухів.

## **Розробка інформаційної технології для визначення точності координації рухів кисті руки у просторі**

### **РОЗДІЛ 2**

#### **2.1 Дослідження прикладного програмного інтерфейсу Leap Motion**

Розробка методу відстеження шляху пальця користувача, а також подальше відображення його у 2D просторі передбачає безконтактну взаємодію користувача з комп'ютером.

Щоб забезпечити такі можливості запропоновано використати тривимірний сенсор Leap Motion, що потребує детального розгляду методів відстеження рук в реальному часі та відображення їх у двовимірному просторі на комп'ютері.

LMC у поєднанні з поточним API надає позиції в декартовому просторі попередньо визначених об'єктів, таких як кінчики пальців, кінчики пера тощо. Надані позиції відносяться до центральної точки приладу, яка розташована на положення другого, центрованого інфрачервоного випромінювача. Як показано на рисунку 2.1, контролер складається з трьох випромінювачів інфрачервоного світла і двох інфрачервоних камер.

Отже, LMC можна класифікувати як систему оптичного відстеження на основі Stereo Vision. Через відсутність точок сцени та попередньо визначених об'єктів, які можна виявити, традиційні методи калібрування не підходять для Leap Motion. Тим не менш, необхідна точна система відліку, щоб оцінити точність і повторюваність контролера. Промислові роботи підтримують можливість кріплення різних інструментів до свого TCP і демонструють високу точність у певному діапазоні. Отже, промислові роботи можуть виступати в якості високоточних еталонних систем під час оцінки LMC.

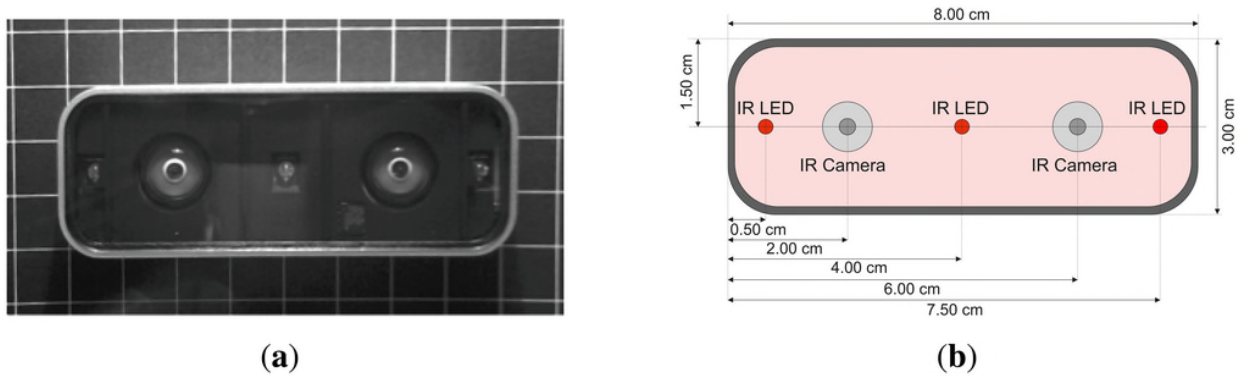


Рисунок 2.1 – Візуалізація (а) реального та (б) схематичного вигляду Leap Motion Controller.

Пристрій комплексу для розробки програмного забезпечення LMC (SDK) (рис. 1) безпосередньо забезпечує деякі з найбільш релевантних точок рухів руки, уникаючи потреби в складних обчисленнях для отримання даних про глибину та колір [11].

Leap motion API дозволяє отримувати дані з пристрою. Розробник може отримати дані рук, рук, пальців, кісток, інструментів, жестів і зображень. Оскільки LMC відстежує руки та пальці в полі зору, він надає оновлення як набір або кадр даних. Кожен об'єкт Frame, який представляє рамку, містить будь-які відстежувані руки з детальним описом їхніх властивостей в один момент часу. Об'єкт Frame по суті є коренем моделі даних (рисунок 2.2). Окремо розглянемо, що представляє об'єкт рука.

Дані руки (рисунок 2.3):

- Тип руки: права або ліва
- Розташування долоні: центр долоні
- Нормальний вектор: вектор, нормальний до площини, що складається з пальців.
- Вектор напрямку: вектор, паралельний напрямку чотирьох пальців.
- Фактори руху: відносні масштаби, обертання, коефіцієнти трансляції для руху
- Швидкість долоні: швидкість руху вашої руки (одиниця: мм/с, вектор)

– Сила захоплення

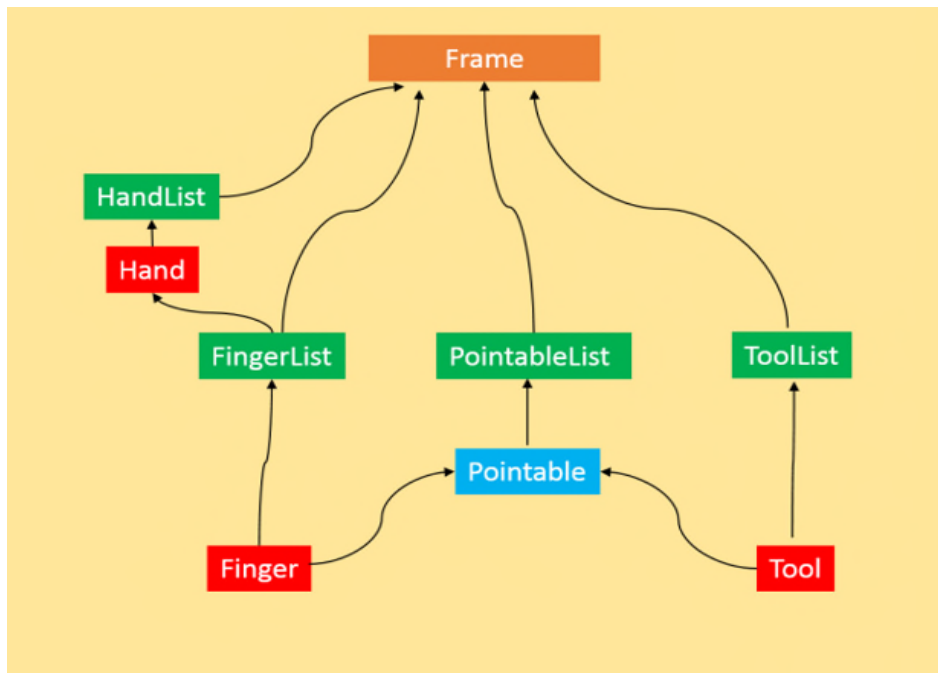


Рисунок 2.2 – Модель даних Leap motion API

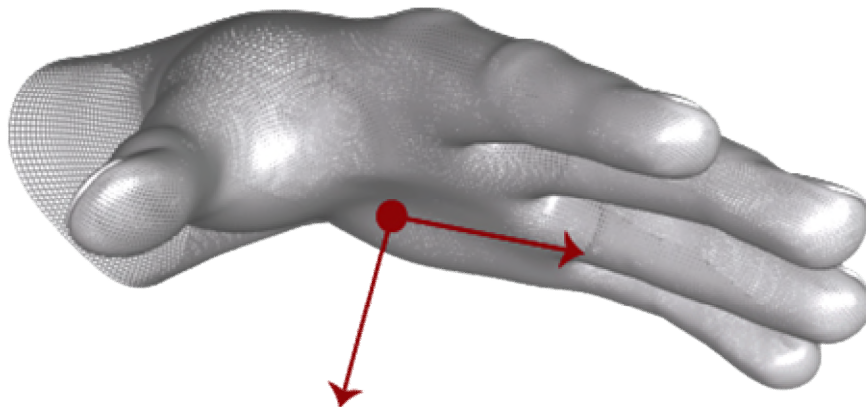


Рисунок 2.3 – Модель руки Leap motion API

Finger належить до вказівних об'єктів, він має ті ж характеристики, що й Tool. Властивість, яка відрізняє інструмент від пальця, полягає в тому, що він довший, тонший і прямиший для прикладу: тонка лінійка, ручка, олівець тощо, позначаються як «інструменти».

Дані пальця (рисунок 2.4):

- Ідентифікатор пальця
- Положення кінчиків пальців
- Вектор кінчика пальця
- Тип пальця, включаючи великий, вказівний, середній, безіменний і мізинець



Рисунок 2.4 – Модель пальців рук Leap motion API

Також можна отримати дані про кістки користувача. Leap Motion може розпізнати чотири кістки кожного пальця. Анатомічно кістки поділяються на чотири типи[12] (рисунок 2.5):

- П'ясткова кістка всередині кисті, що з'єднує палець із зап'ястком (крім великого пальця).
- Проксимальна фаланга – кістка в основі пальця, з'єднана з долонею
- Проміжна фаланга - середня кістка пальця, розташована між кінчиком і основою
- Дистальна фаланга – кінцева кістка на кінці пальця

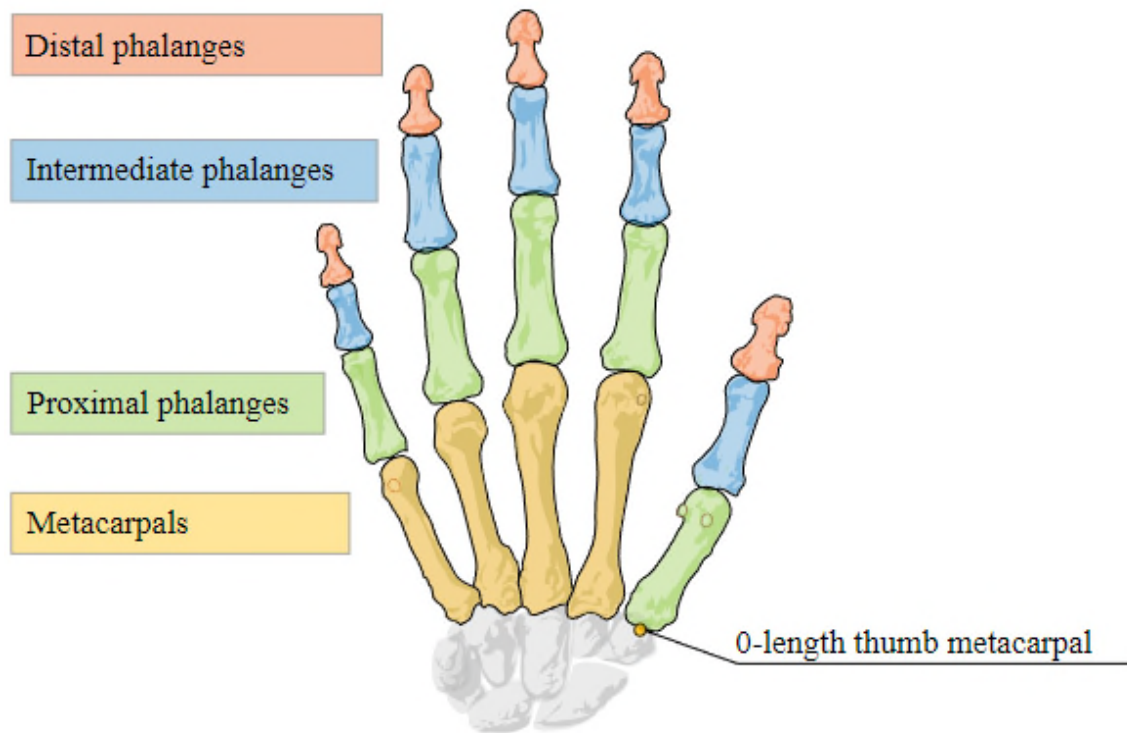


Рисунок 2.5 – Модель кісток руки Leap motion API

## 2.2 Розробка методу відстеження шляху пальця користувача та відображення його у 2D просторі.

Основним завданням під час використання контролера Leap Motion у програмі є відображення значень координат, отриманих від контролера, у відповідну систему координат, визначену програмою.

LMC надає координати в реальних міліметрах у системі відліку самого приладу. Тобто, якщо положення кінчика пальця задано як  $(x, y, z) = [100, 100, -100]$ , ці числа є міліметрами – або  $x = +10$  см,  $y = 10$  см,  $z = -10$  см.

Саме апаратне забезпечення LMC є центром цієї системи відліку. Початкова точка розташована вгорі, по центру обладнання. Тобто, якщо користувач торкнетесь середини контролера, то координати кінчика пальця будуть  $[0, 0, 0]$  (рисунок 2.6):

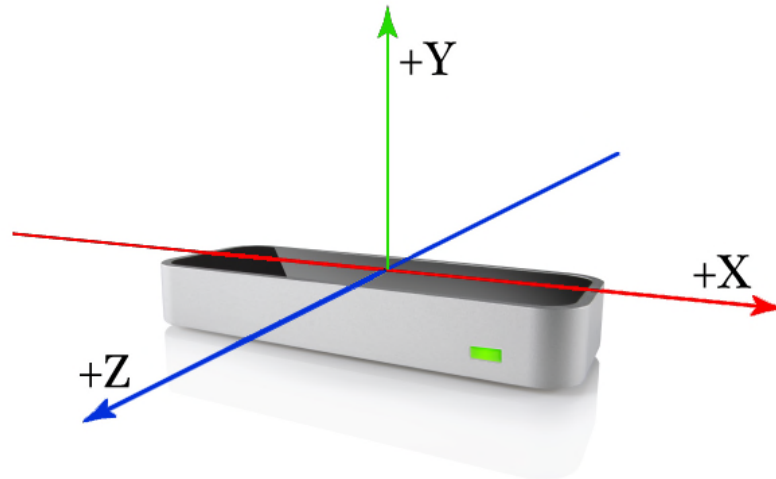


Рисунок 2.6 – Початок координат пристрою Leap motion

У звичайному положенні, тобто на столі з користувачем з одного боку та монітором комп'ютера з іншого, користувач знаходиться попереду контролера, а екран монітора позаду. Якщо користувач увімкне автоматичну орієнтацію, програмне забезпечення налаштує систему координат, якщо контролер перевернутий (зелений світлодіод спрямований від користувача). Однак, якщо користувач розташує свій контролер в іншому положенні (догори ногами або набік), програмне забезпечення не зможе це виявити або відкоригувати, що призведе до некоректної роботи застосунка.

При розробці потрібно пам'ятати про ці моменти, та попереджувати користувача не наближатися надто близько до контролера за допомогою візуальних підказок; наведення рук прямо на контролер збільшує ймовірність того, що пальці та руки будуть закривати один одного від видимості.

Щоб використовувати інформацію з пристрою для програми, потрібно інтерпретувати ці дані так, щоб вони мали сенс для неї. Наприклад, щоб зіставити координати LMC із координатами програми, потрібно вирішити, які осі використовувати, яку частину поля зору приладу використовувати та чи використовувати абсолютне відображення чи відносне відображення.

Для 3D-додатків зазвичай має сенс використовувати всі три осі пристрою. Для 2D-додатків можна відкинути одну вісь, як правило, вісь z, і відобразити вхідні дані з двох решти осей на координати 2D-додатку.

Окрім цього, також потрібно вирішити, яку частину діапазону Leap Motion і поля зору використовувати у програмі. Поле зору являє собою перевернуту піраміду. Доступний діапазон по осях x і z набагато менший біля пристрою, ніж у верхній частині діапазону. Якщо використовувати занадто широкий діапазон або дозволяти занадто низьку висоту, тоді користувач не зможе дістатися до нижніх кутів програми.

Для коректного відображення потрібно використовувати формули перетворення:

$$x_{app} = (x_{leap} - Leap_{start}) \frac{App_{range}}{Leap_{range}} + App_{start}$$

Значення  $x_{leap}$  зміщує центр шкали x вліво або вправо. Оскільки кожна рука природно лежить праворуч або ліворуч від контролера Leap Motion, користувачеві може бути легше взаємодіяти з повноекранною програмою, якщо змістити відображення координат на основі того, яка рука використовується.

$$y_{app} = (y_{leap} - Leap_{start}) \frac{App_{range}}{Leap_{range}} + App_{start}$$

$y_{leap}$  встановлює значення, яке відноситься до нижньої частини області відображення. Більше значення означає, що користувач повинен тримати руку вище над контролером, що також означає, що він, швидше за все, зможе дістатися до нижніх кутів.

$$Leap_{range} = Leap_{end} - Leap_{start}$$

$$App_{range} = App_{end} - App_{start}$$

Змінюючи початкову  $Leap_{start}$  та кінцеву  $Leap_{end}$  точки, змінюються відображення координат, щоб рухи охоплювали більшу чи меншу область у програмі. Зменшивши значення діапазону, менший діапазон фізичного руху буде зіставлено з шириною області відображення, що полегшить переміщення з одного боку на інший.

Використовуючи цей метод, можна повністю налаштувати, взаємодію користувача з додатком, в залежності від вимог. Більш загальним методом перетворення координат включає в себе використання InteractionBox – це стандартний функціонал Leap Motion [13], який визначає прямолінійну область у полі зору (рисунок 2.7).

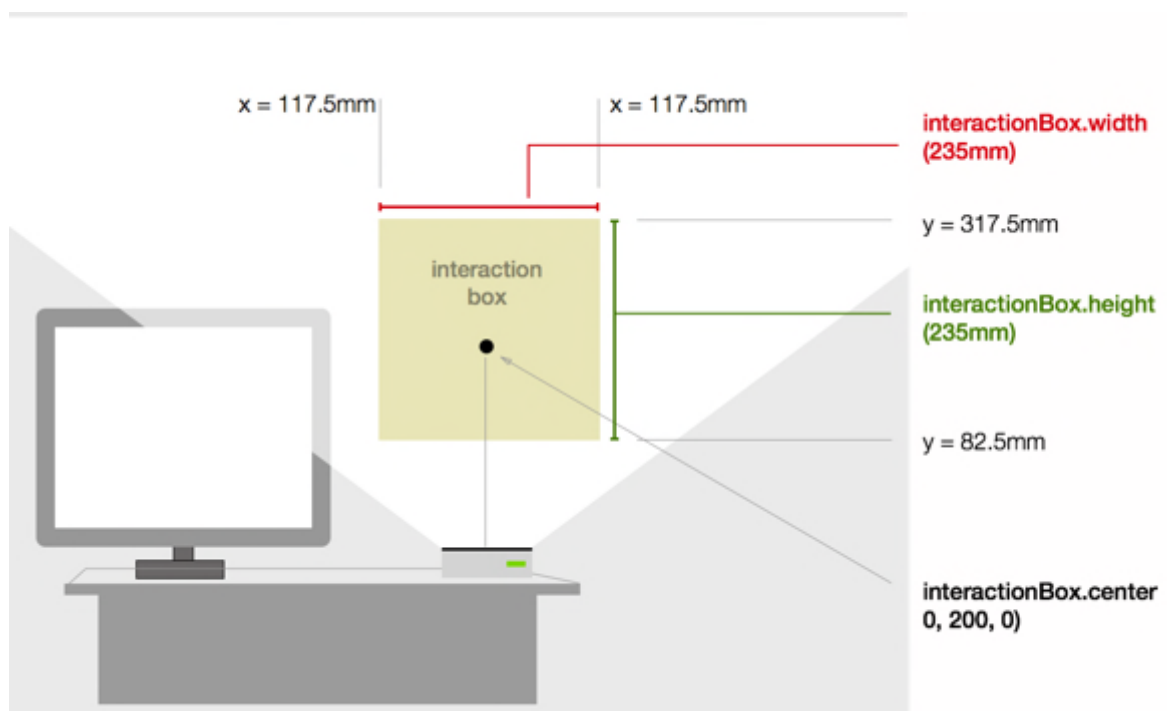


Рисунок 2.7 – Зображення InteractionBox

Поки рука або палець користувача залишаються в цьому полі, вони гарантовано залишаються в полі зору Leap Motion. Використовувати цю гарантію,

зіставляючи область взаємодії програми з областю, визначеною `InteractionBox`, замість зіставлення з усім полем огляду `Leap Motion`. Клас `InteractionBox` надає метод `normalizePoint`, щоб допомогти зіставити координати `Leap Motion` з координатами програми, спочатку нормалізувавши їх до діапазону  $[0..1]$ .

Розмір `InteractionBox` визначається полем огляду `Leap Motion` і налаштуванням висоти взаємодії користувача. Програмне забезпечення контролера регулює розмір коробки залежно від висоти, щоб утримувати нижні кути в полі зору. Якщо встановити висоту взаємодії вище, то коробка стає більшою. Користувачі можуть встановити висоту взаємодії на основі висоти, яку вони вважають за краще тримати руками під час використання контролера. Деякі користувачі люблять тримати руки набагато вище над пристроєм, ніж інші. Використовуючи `InteractionBox` для відображення системи координат `Leap Motion` із системою координат програми, можна пристосуватись до обох типів користувачів. Користувач також може встановити автоматичне налаштування висоти взаємодії. Якщо користувач переміщує свої руки нижче поточної підлоги вікна взаємодії, програмне забезпечення контролера знижує висоту, аналогічно, якщо користувач переміщується над полем взаємодії, контролер піднімає поле.

Оскільки поле взаємодії може змінюватися з часом, нормалізовані координати точки, виміряні в одному кадрі, можуть не збігатися з нормалізованими координатами тієї самої точки реального світу, нормалізовані за допомогою поля взаємодії, наданого іншим кадром. Таким чином, прямі лінії або ідеальні кола, «намальовані в повітрі» користувачем, можуть не бути прямими або гладкими, якщо вони нормалізовані між кадрами.

Якщо використовувати покадрову нормалізацію, і при цьому «`InteractionBox`» буде зміщуватися, може виникнути ефект що спотворить жест, який робить користувач (рисунок 2.8).

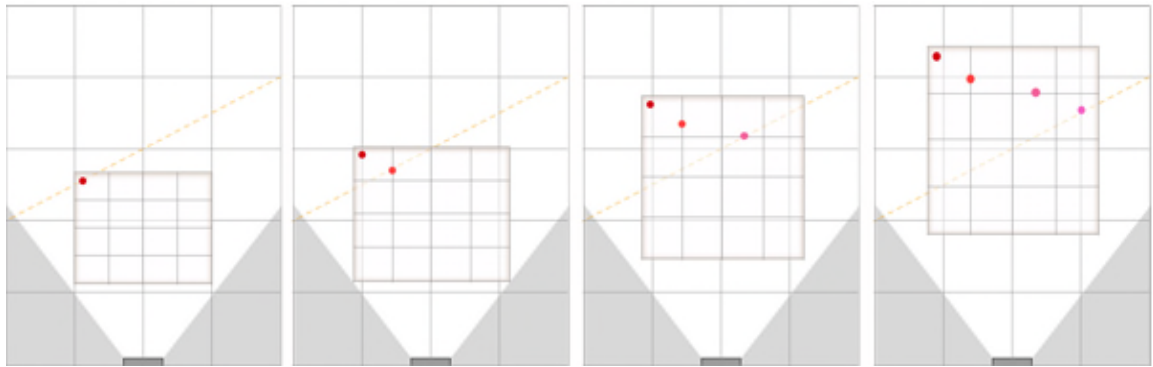


Рисунок 2.8 – Спотворення жест при розпізнаванні

Для більшості програм це не є суттєвою проблемою, але у випадку з перевіркою точності рухів, потрібно гарантувати послідовну нормалізацію набору точок, які відстежуються між кадрами, для цього потрібно зберегти один об'єкт InteractionBox – в ідеалі об'єкт із найбільшою висотою, шириною та глибиною – і використовувати його для нормалізації всіх точок.

Щоб використовувати вікно взаємодії, спочатку нормалізується точка, яку потрібно відобразити, а потім перетворюється нормалізовані координати у систему координат програми. Нормалізація перетворює точки в межах області взаємодії на діапазон  $[0..1]$  і переміщує початок координат у нижній лівий кут. Після чого помножити ці нормалізовані координати на максимальний діапазон для кожної осі системи координат програми, зсуваючи та інвертуючи вісь за потреби.

У випадку з 2D додатком, початок системи координат зазвичай розміщують у верхньому лівому куті вікна – і, природно, не використовують вісь  $z$ . Для кращого розуміння всього процесу методу відстеження шляху пальця користувача та відображення його у 2D просторі створено діаграму діяльності, в якій продемонстровано покрокове виконання методу (рисунок 2.9).

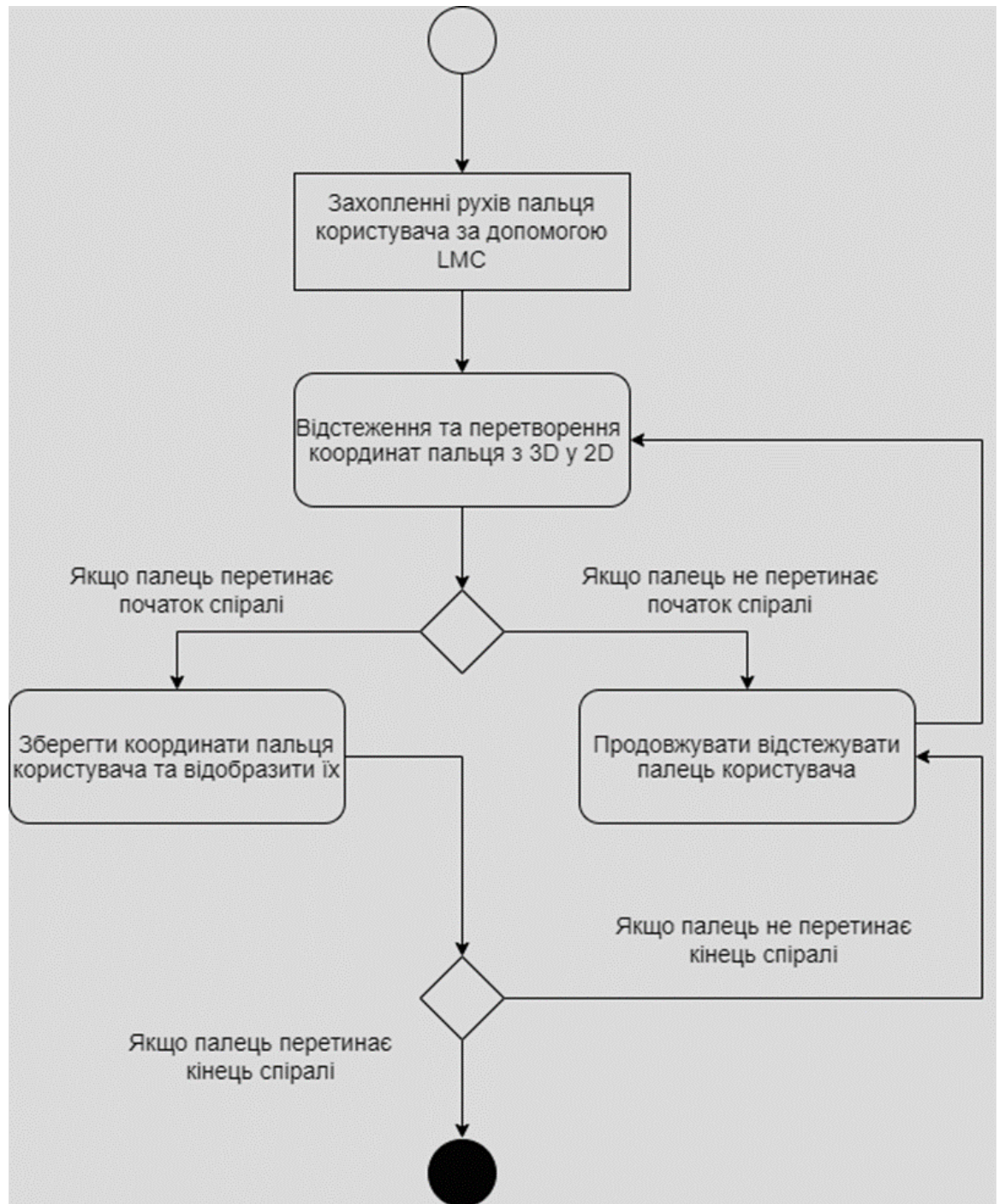


Рисунок 2.9 – Діаграма діяльності методу відстеження та відображення пальця в 2D

### 2.3 Розробка методу визначення точності координації рухів кисті руки.

Архімед досліджував конкретне рівняння, яке створювало спіраль, радіус якої змінювався безпосередньо залежно від кута від початку спіралі (рисунок 2.10) [14].

Спіральне рівняння в полярній системі координат:

$$r(\theta) = a\theta$$

де  $a$  визначає швидкість відхилення спіралі від початку координат.

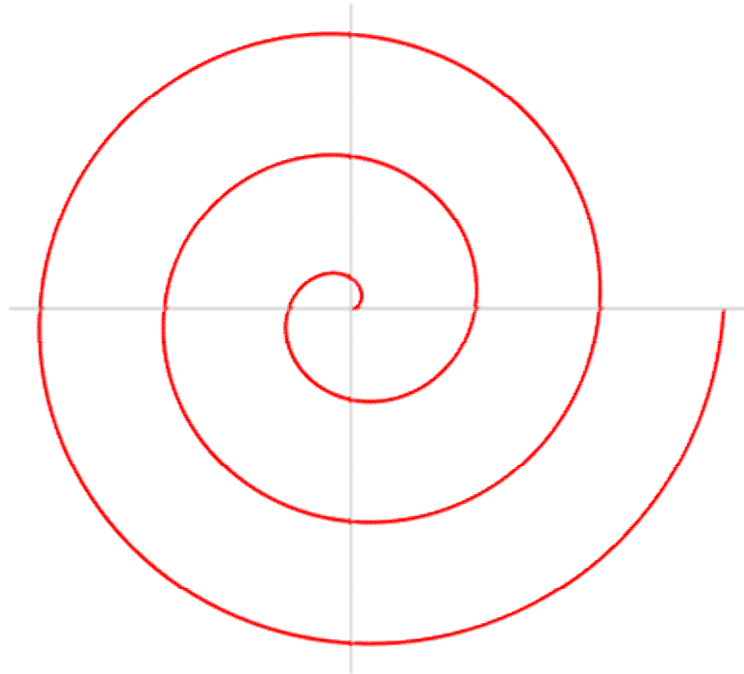


Рисунок 2.10 – Спіраль Архімеда

Спіраль використовується 2200 років тому як тестовий стимул, який був підтверджений як метод кількісної оцінки есенціального тремору та хвороби Паркінсона [15]. Спіральний малюнок Архімеда фіксує частоту, амплітуду та напрямок тремтіння без необхідності враховувати стилістичні відмінності почерку (Рисунок 2.11). Оскільки вона вимагає одного безперервного руху пера без коротких перерв між написанням слів, спіраль підкреслює аномальні рухи дистонії, гіпокінезії та тремору. Було показано, що аналіз спіралей, намальованих на папері або графічному планшеті, є чутливим і дійсним методом кількісної оцінки тремору в пацієнтів з есенціальним тремором [16].

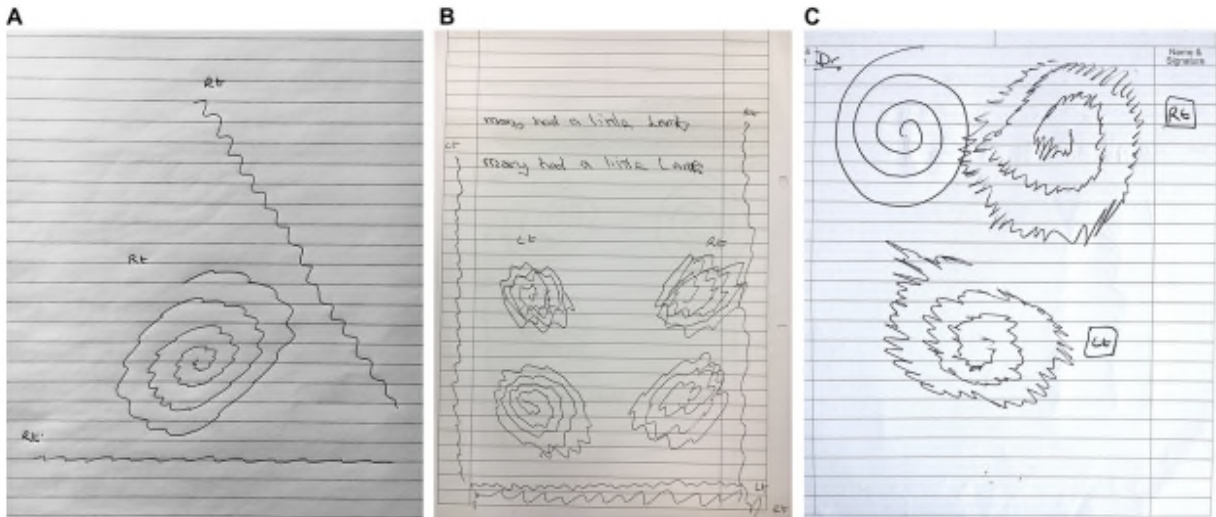


Рисунок 2.11 – Малюнки пацієнтів з різними випадками тремору.

Використовуючи спіраль Архімеда для тестування, в якому користувачеві потрібно повторити фігуру спіралі, за допомогою пальця в просторі. В процесі всього тесту, пересування пальця буде відстежуватись, після додавання всіх точок даних до шляху користувача, потрібно проаналізувати його. Якби шлях був намальований ідеально, кожен рух по спіралі збільшував би радіус із лінійною швидкістю. Перетворюючи кожен точку  $(x, y)$  на шляху користувача в радіус (від початку спіралі), можна порівняти, наскільки радіуси, що збільшуються, підходять до лінії, використовуючи лінійну регресію найменших квадратів і встановлюючи значення  $R^2$  як відсоток точності користувача, для цього процесу створено діаграму діяльності, в якій відображено весь процес методу (рисунок 2.12).

У більшості випадків точки даних не лежать на прямій лінії, що дає можливість відобразити зв'язок між двома змінними за допомогою кількох різних ліній. Вибір кожної лінії може призвести до ситуації, коли лінія буде ближче до одних точок і далі від інших точок. Та не можливо вирішити, яка лінія найкраще відповідає даним. У таких випадках для визначення лінії найкращого підходу можна використовувати метод найменших квадратів. Він визначає лінію, яка найкраще підходить для заданих спостережених даних, мінімізуючи суму квадратів - відхилень від кожної точки даних до лінії.

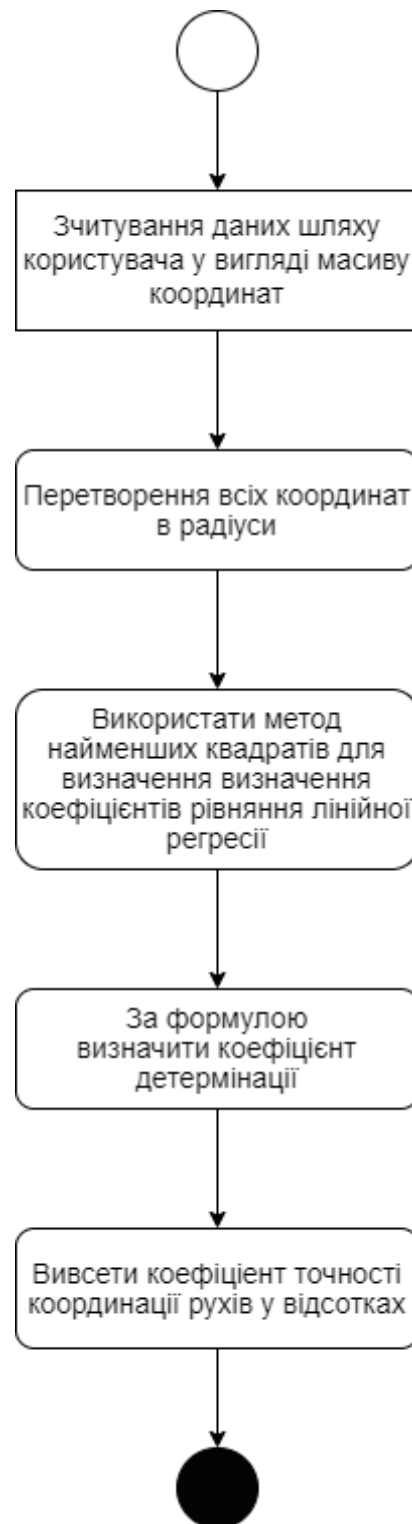


Рисунок 2.12 – Діаграма методу визначення точності координації рухів кисті руки

Математична процедура для знаходження кривої, яка найкраще підходить для заданого набору точок, шляхом мінімізації суми квадратів зсувів («залишків») точок від кривої (рисунок 2.13). Сума квадратів зсувів використовується замість

абсолютних значень зсуву, оскільки це дозволяє розглядати залишки як неперервну диференційовану величину. Однак, оскільки використовуються квадрати зміщень, віддалені точки можуть мати непропорційний вплив на підгонку, властивість, яка може бути або не бажаною залежно від поточної проблеми.

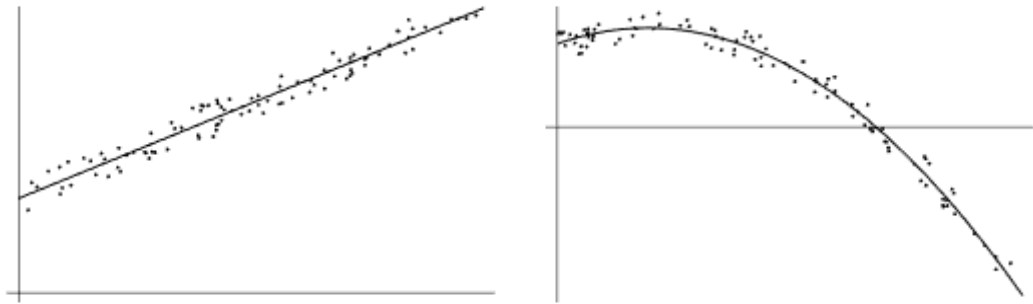


Рисунок 2.13 – Крива, яка найкраще підходить для заданого набору точок

Метод найменших квадратів [17] полягає в тому, що сума квадратів відхилень значень  $y$  від отриманого рівня регресії – мінімальна. Рівняння лінійної регресії має вигляд:

$$y = ax + b$$

$a$ ,  $b$  – коефіцієнти лінійного рівняння регресії;

$x$  – незалежна змінна, в даному випадку це буде номер послідовності, так як чим далі малює користувач, тим більший має бути радіус;

$y$  – залежна змінна, відповідно радіус послідовності.

Для перетворення, координат користувача в радіус використовуватиметься формула рівняння кола:

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} = d$$

Вже з неї можна отримати радіус:

$$(x - h)^2 + (y - k)^2 = r^2$$

$h, k$  - координати початку спіралі Архімеда;

$x, y$  – поточні координати пальця користувача.

Отримавши всі дані, подальше обчислення відбувається за допомогою формул визначення коефіцієнтів рівняння лінійної регресії:

$$a = \frac{n \sum_{i=1}^n x_i y_i - (\sum_{i=1}^n x_i)(\sum_{i=1}^n y_i)}{n * \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2}$$

$$b = \frac{\sum_{i=1}^n y_i - a \sum_{i=1}^n x_i}{n}$$

Формула коефіцієнта детермінації  $R^2$ :

$$R^2 = \frac{\sum(y_x - \bar{y})^2}{\sum(y_i - \bar{y})^2} = 1 - \frac{\sum(y_i - y_x)^2}{\sum(y_i - \bar{y})^2}$$

$R$ -квадрат - це статистичний показник того, наскільки близько дані до підігнаної лінії регресії [18]. Він також відомий як коефіцієнт детермінації або коефіцієнт множинної детермінації для множинної регресії. Визначення  $R$ -квадрату є досить простим; це відсоток варіації змінної відповіді, який пояснюється лінійною моделлю.  $R$ -квадрат завжди становить від 0 до 100%. 0% вказує на те, що модель не пояснює жодної мінливості даних відповіді навколо її середнього значення. 100% означає, що модель пояснює всю мінливість даних відповіді навколо її середнього значення.

Нанесення підігнаних значень за спостережуваними значеннями графічно ілюструє різні значення  $R$ -квадрат для регресійних моделей (рисунок 2.14). Регресійна модель ліворуч становить 38,0% дисперсії, тоді як модель справа – 87,4%. Чим більшу дисперсію враховує регресійна модель, тим ближче точки даних будуть знижуватись до підігнаної лінії регресії. Теоретично, якби модель могла пояснити 100% дисперсії, підігнані значення завжди дорівнювали б спостережуваним значенням, і, отже, усі точки даних потрапляли б на лінію регресії.

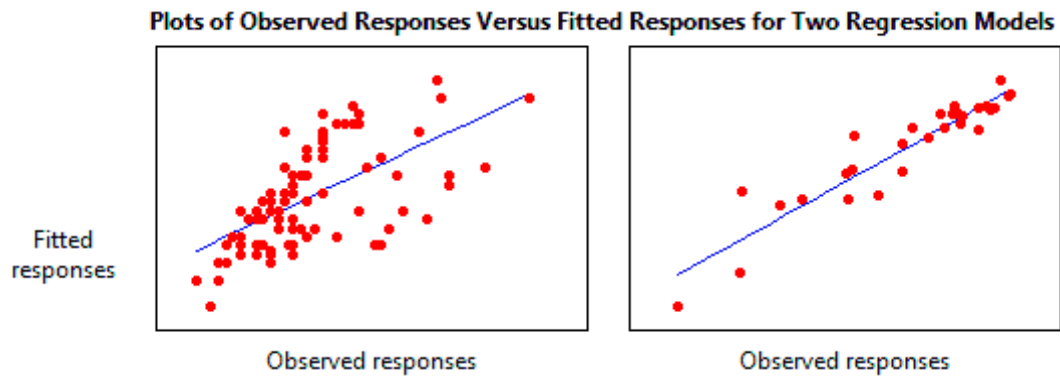


Рисунок 2.14 – Приклад регресійних моделей

Загалом, що вищий R-квадрат, то краще модель відповідає даним. Цей коефіцієнт і буде визначати точність координації рухів користувача. Тобто, якщо намалювати найбільш наближену форму до спіралі Архімеда, тоді цей коефіцієнт буде достатньо високим, та навпаки у випадку відхилення, нижчий коефіцієнт означитиме проблеми з координацією.

## Висновки до розділу 2

В другому розділі розроблено інформаційну технологію для визначення точності координації рухів кисті руки у просторі. В процесі чого досліджувалось прикладний програмний інтерфейс для пристрою Leap Motion, який використовується для методу відстеження шляху пальця користувача.

Розроблено алгоритм для методу відстеження шляху пальця користувача та відображення його у 2D просторі та описані формули перетворення для нього.

При розробці методу визначення точності координації рухів кисті руки, описано основний математичний метод найменших квадратів, та розписані формули для визначення коефіцієнту точно рухів.

## РОЗДІЛ 3

### **Проектування інформаційної системи визначення точності координації рухів кисті руки у просторі.**

Мета роботи полягає у розробці інформаційної системи, яка б дала можливість визначення точності координації рухів кисті руки у просторі. з використанням зовнішніх пристроїв. Так як в теперішній час веб-застосунки більш популярні за десктопі, тому, що вони однаково працюють з будь-яким пристроєм, чи то смартфон, планшет, ноутбук, чи стаціонарний комп'ютер - адже вони практично не залежить від операційної системи. Також вони доступні з будь-якої точки світу, з будь-якого пристрою, а файли користувача завжди будуть під рукою. Тобто основною їхньою перевагою це доступність, що є дуже важливим для інформаційної системи визначення точності координації рухів кисті руки у просторі, тому було вирішено реалізувати її саме таким способом.

Для досягнення поставленої мети необхідно розробити наступні моделі та компоненти:

- функціональну модель, яка задовольняє основні вимогами;
- об'єктно орієнтована модель
- структура інформаційної системи
- модель взаємодії користувача з інтерфейсом веб-додатка;

Також, потрібно описати функціональні блоки моделі, з яких складається система, їх залежності та взаємодію.

### **3.1 Основні вимоги для проектування інформаційної системи**

При реалізації інформаційної системи визначення точності координації рухів, важливим є створення функціональної моделі, яка б визначала її основні

можливості. Для даної інформаційної системи було визначено наступні функціональні можливості:

- робота з даними отримані від LMC;
- використання спіралі Архімеда для тестування ;
- відстежування траєкторії рухів користувача, та їх відображення в зрозумілому вигляді;
- відображення показників після проходження тесту.

Також було визначено, деякі не функціональні вимоги, які покращують розробку веб-додатку та взаємодію його з користувачем.

– Користувальницький досвід. При розробці інтерфейсу особливо у випадку веб-сайту, який має багато сторінок, слід пам'ятати про користувачів, щоб при взаємодії в них не виникало труднощів та все було зрозуміло.

– Інструкція. Оскільки, використовуються Leap Motion, потрібно використовувати інструкції та підказки, які допоможу користувачеві при взаємодії з веб-додатком за допомогою цього пристрою

– Формалізація. Цей принцип полягає у розробці певних правил, інструкцій, нормативів, які створюють формальну модель для визначають поведінки та структуру взаємодію системи. Використовуючи таку функціональну модель надає можливість реалізації на різних платформах.

Окрім, того важливим аспектом, є використання додаткових приладів, таких як LMC, для якого потрібно додати додаткові інструкції та вказівки.

### **3.2 Функціональна модель інформаційної системи**

Для даної інформаційної системи було запропоновано функціональну модель, яка забезпечує основні функціональні вимоги та складається з декількох блоків (рисунок 3.1).

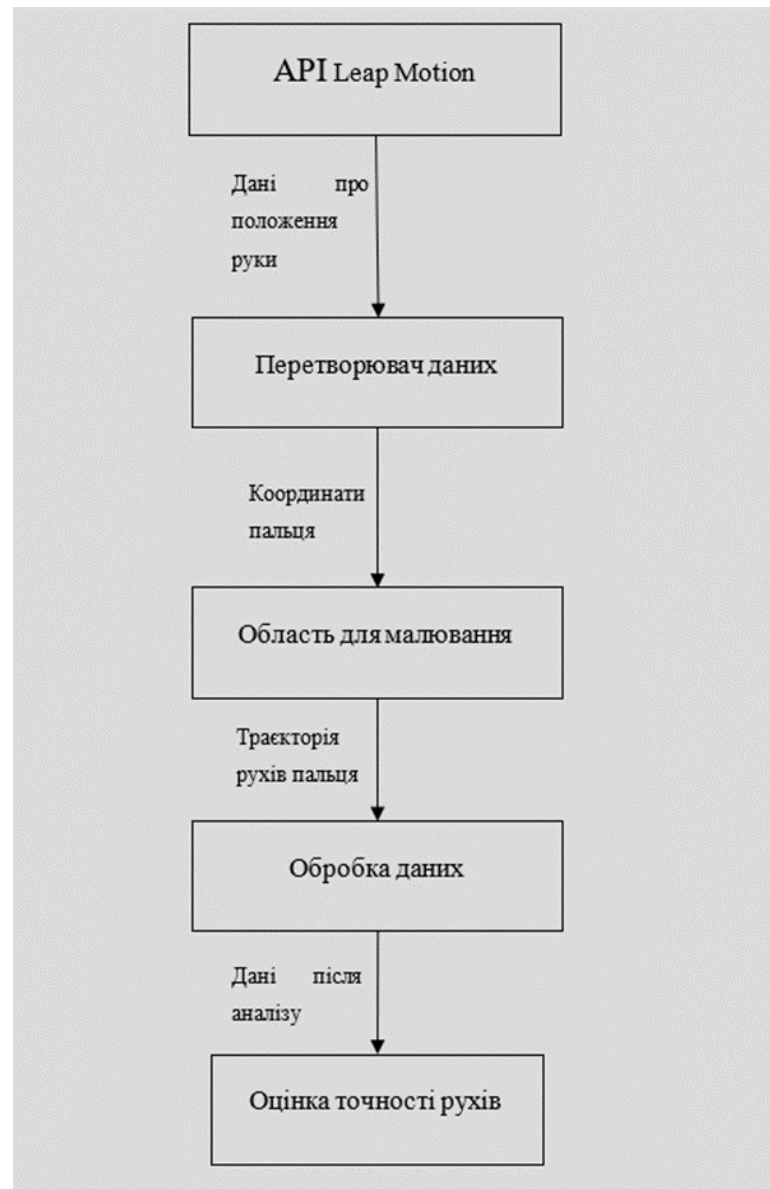


Рисунок 3.1 – Функціональна модель інформаційної системи

*Блок API Leap Motion* готова бібліотека, яка використовується для взаємодії з Leap Motion. Безперервно обробляє дані отриманих з камер та датчиків пристрою, після чого обраховує основні характеристики положення рук користувача в просторі, та передає їх далі в разі запиту.

*Блок перетворення даних* виконує запит до API Leap Motion для отримання даних, та в подальшому перетворює їх в форму зрозумілу для веб-додатку. Використовується метод для перетворення тривимірних координат у двовимірні. За допомогою цього блоку користувач може взаємодіяти з програмою через LMC.

*Блок області для малювання* де відбувається основна взаємодія користувача та програми, для проходження тесту у вигляді спіралі Архімеда на координацію рухів. При взаємодії користувача з цією областю промальовується траєкторія всього пройденого шляху. Для початку та завершення тесту використовується метод обмежувального кола, для перевірки зіткнення пальця користувача з віртуальними об'єктами.

*Блок обробки даних* аналізує отримання дані, після проходження користувачем тесту, за допомогою методу найменших квадратів. Отримання результати зберігаються в базі даних на сервері веб-сайту.

*Блок оцінки точності рухів* визначає коефіцієнт детермінації з даних які були отримані з усіх тестів пройдених користувачем з бази даних, та виводить цей показник користувачу, як коефіцієнт точності рухів. Окрім того виводиться графік оцінювання, в якому більш детально можна оглянути результати аналізу, а також і дані всіх попередніх результатів, для відслідковування всієї історії та прогресу.

Отримана функціональна модель описує процес обробки інформації, отриманні дані на вході, та дані які вийшли на виході, що дозволяє реалізувати інформаційну систему визначення точності координації рухів.

### **3.3 Об'єктно орієнтована модель інформаційної системи**

Для об'єктно орієнтованої моделі потрібно визначити основні сутності предметної області інформаційної системи, а також визначити їх властивості та функції. Для інформаційної системи визначення точності координації рухів було визначено наступні сутності:

- Тест – основний об'єкт, який керує всім процесом проходження тесту користувача від початку до кінці. Починаючи від захоплення рухів користувача контролером, в подальшому відображення траєкторії, обробки даних, розрахунок коефіцієнта точності, та закінчуючи відображенням результату проходження тесту;

- контролер – представляє сам контролер Leap Motion, відповідає за захоплення рухів користувача, налаштування пристрою, обробки даних з камер та датчиків. Є основною сутністю за допомогою якого і отримується вся інформація для проведення тестів;

- фрейм – об'єкт для обробки кожного кадру захоплення рухів, потрібен для конфігурації даних які будуть отримані з API Leap Motion, їх конвертацію для інформаційної системи, а саме перетворення з тривимірних координат у двовимірні. Контролер безперервно захоплює рухи користувача, що відповідає одному фрейму в секунду, кожен цей об'єкт зберігає всі властивості про захопленні рухи;

- рука – кожен фрейм містить певні дані про відстежуванні рухи, в основному це дві руки на один фрейм, але LMC здатний відстежувати і більше, при цьому знижується якість отримуваної інформації. Даний об'єкт відповідає за конкретні дані про руку користувача, в основному використовується для отримання даних про пальці, так як за допомогою них буде отримуватись оцінка, але оскільки це також важлива інформація, яка може знадобиться в подальшому для отримання додаткових показників, було вирішено використовувати його. Має такі основні властивості: Тип руки, розташування долоні; нормальний вектор; вектор, нормальний до площини; вектор напрямку; фактори руху; швидкість долоні; сила захоплення;

- пальці – основний об'єкт, який використовується для визначення точності рухів, представляє власне палець користувача, відповідно, об'єкт рука може містити, 5 таких об'єктів, і про кожен зберігається стан його властивостей, хоча в тестування використовується лише вказівний палець. Його властивості: ідентифікатор пальця, положення кінчиків пальців, вектор кінчика пальця, тип пальця;

- тремор – допоміжний об'єкт, але не менш важливий, який слугує для визначення точності координації рухів, за допомогою методу найменших квадратів, також має доступ до додаткових математичних функцій та формул. Дозволяє розрахувати коефіцієнт координації користувача та інші дані в процесі обрахунку;

- спіраль – в даній інформаційній системі для тесту використовується спіраль Архімеда, і даний об'єкт представляє його. Надає функціонал для побудови спіралі використовуючи математичні формули та розрахунки. Основна властивість це швидкість відхилення спіралі від початку координат, що може вплинути на сам тест;

- траєкторія – об'єкт, який працює разом із фреймом, після початку тесту, відображає траєкторію пройдену користувачем від початкової точки спіралі до кінцевої, на основі захоплених рухів.

- користувач – відображає об'єкт самого користувача, потрібен для доступу у використання основного функціоналу інформаційної системи, за допомогою якого користувач матиме змогу проходити тести, переглядати результат, та всю історію тестів, які він проходив раніше. Основна властивість, якраз буде коефіцієнт точності координації рухів.

На основі описаних вище сутностях було побудовано ER-діаграму, для схематичного представлення взаємодії всіх об'єктів, зважаючи на їхні функції та властивості. Тобто, починається все з об'єкта користувача, який розпочинає тест, який в свою чергу є головним об'єктом і містить всі інші сутності. Проміжна сутність інтерфейс потрібна для відображення об'єктів які слугують, для взаємодії з користувачем, а саме об'єкт спіраль яка малює спіраль Архімеда, траєкторія для малювання шляху який пройшов користувач, та тремор для результату. Також об'єкт контролер, що захоплює всі рухи, в подальшому які обробляється фреймом, який передає дані для руки та пальця.

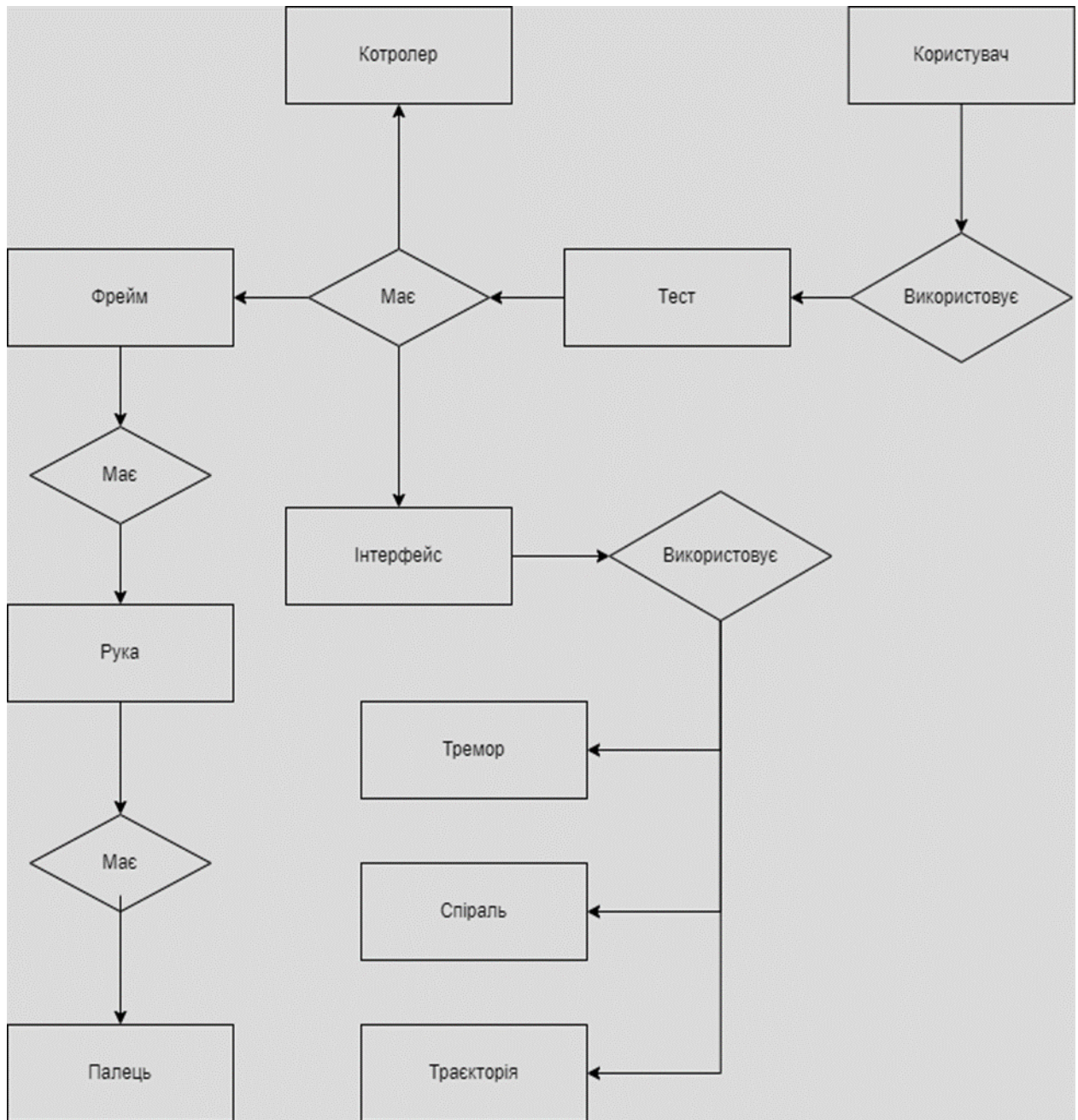


Рисунок 3.2 – ER-діаграма інформаційної системи

### 3.4 Розробка структури інформаційної системи

Для розробки структури бази даних інформаційної системи визначення точності координати рухів необхідно зберігати, зчитувати, та редагувати дані про сутності предметної області, а саме: тест, контролер, фрейм, рука, пальці, користувач, не враховуючи сутності які відносяться до інтерфейсу для взаємодії з користувачем: тремор, спіраль, траєкторія. Оскільки було використано об'єктно

орієнтований підхід при проектуванні системи, буде використано реляційну бази даних. Це тип бази даних, яка зберігає та забезпечує доступ до точок даних, пов'язаних одна з одною. Реляційні бази основані на реляційній моделі, інтуїтивно зрозумілому, простому способі представлення даних у таблицях, на рисунку 3.3 представлено схему таких таблиць.

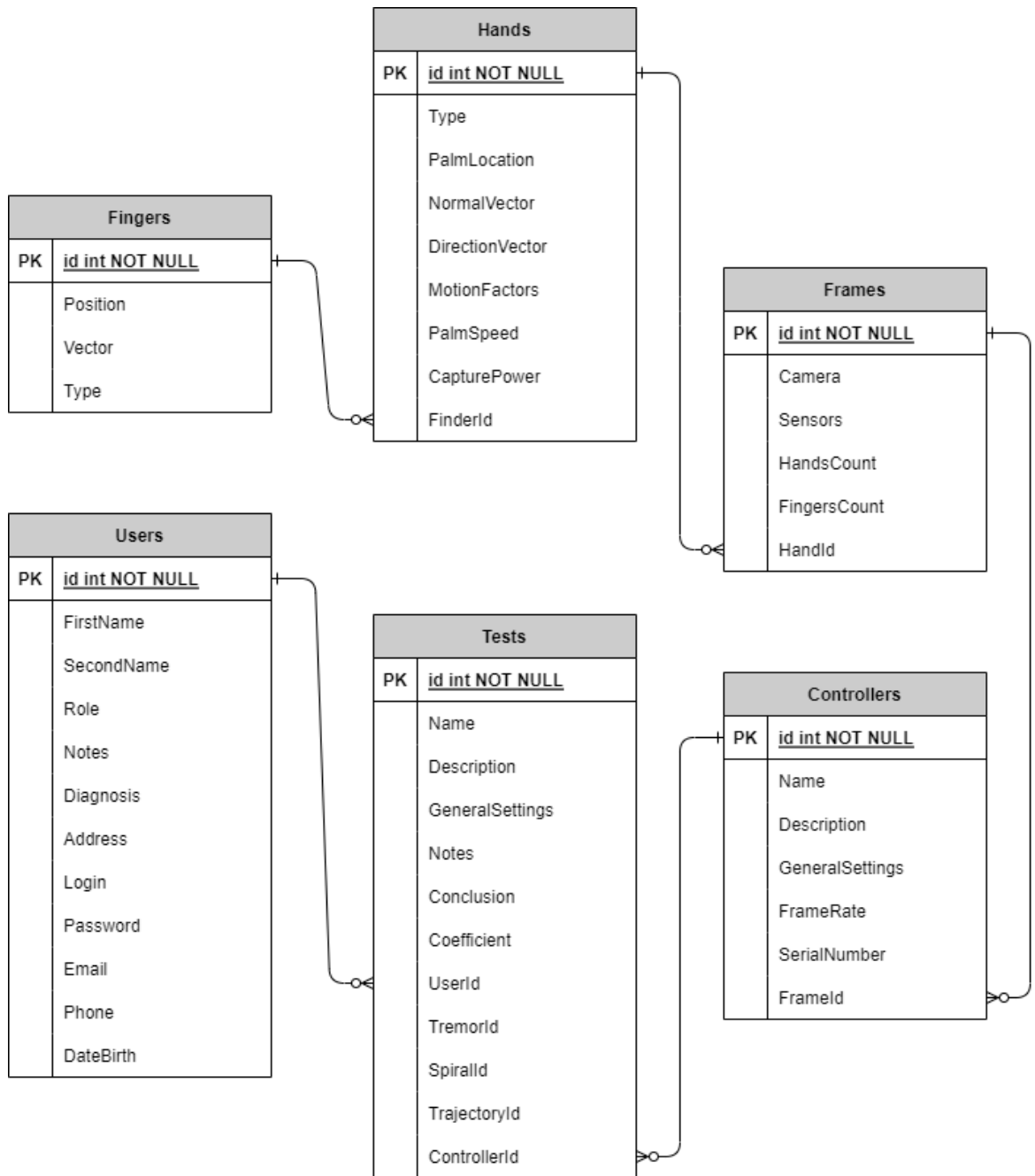


Рисунок 3.3 – Інфологічна модель бази даних

Таблиця «Tests» призначена для збереження даних про карти (хоча для першого альфа запуску працюватиме лише одна мапа) й містить наступні атрибути: ID, Name, Notes, Conclusion, Coefficient, GeneralSettings (таблиця 3.1).

Таблиця 3.1 – Атрибути таблиці «Tests»

№ п/п	Назва	Тип даних	Опис
1	ID	INT	Первинний ключ Унікальний ідентифікатор
2	Name	NVARCHAR	Назва тесту
3	Description	NVARCHAR	Короткий опис від користувача, використовується в якості відгуку
4	GeneralSettings	NVARCHAR	Налаштування тесту
5	Notes	NVARCHAR	Нотатки від лікаря
6	Conclusion	NVARCHAR	Висновок тесту від лікаря
7	Coefficient	DECIMAL	Результат тесту, коефіцієнт точності координат рухів
8	UserId	INT	Вторинний ключ
9	TremorId	INT	Вторинний ключ
10	SpiralId	INT	Вторинний ключ
11	TrajectoryId	INT	Вторинний ключ
12	ControllerId	INT	Вторинний ключ

Таблиця «Controllers» призначена для збереження даних про пристрій Leap Motion й містить наступні атрибути: ID, Name, Description, GeneralSettings, FrameRate, SerialNumber (таблиця 3.2).

Таблиця 3.2 – Атрибути таблиці «Controllers»

№ п/п	Назва	Тип даних	Опис
1	ID	INT	Первинний ключ Унікальний ідентифікатор
2	Name	NVARCHAR	Серійна назва приладу

3	Description	NVARCHAR	Короткий опис про прилад
4	GeneralSettings	NVARCHAR	Налаштування приладу
5	FrameRate	DECIMAL	З якою частотою обробляти фрейми
6	SerialNumber	DECIMAL	Серійний номер виробництва
7	FrameId	INT	Вторинний ключ

Таблиця «Frames» призначена для збереження даних про фрейми, оскільки пристрій обробляє фрейми кожної секунди, і якщо зберігати їх всіх, в найближчий час це призведе до накопичення великої кількості даних, тому було вирішено зберігати лише загальну інформацію для всіх фреймів за один тест. Містить наступні атрибути: ID, Camera, Sensors, HandsCount, FingersCount (таблиця 3.3).

Таблиця 3.3 – Атрибути таблиці «Frames»

№ п/п	Назва	Тип даних	Опис
1	ID	INT	Первинний ключ Унікальний ідентифікатор
2	Camera	NVARCHAR	Інформація отримана з камер
3	Sensors	NVARCHAR	Інформація отримана з сенсорів
4	HandsCount	INT	Інформація про кількість рук на даному фреймі, які захопив прилад
5	FingersCount	INT	Інформація про кількість пальців на даному фреймі, які захопив прилад
6	HandId	INT	Вторинний ключ

Таблиця «Hands» призначена для збереження даних про руки користувача, які були захоплені приладом. Містить наступні атрибути: ID, PalmLocation, DirectionVector, MotionFactors, PalmSpeed, CapturePower (таблиця 3.4).

Таблиця 3.4 – Атрибути таблиці «Hands»

№ п/п	Назва	Тип даних	Опис
1	ID	INT	Первинний ключ

			Унікальний ідентифікатор
2	Type	NVARCHAR	Тип руки: права або ліва
3	PalmLocation	NVARCHAR	Розташування центру долоні
4	NormalVector	NVARCHAR	Нормальний вектор до площини, що складається з пальців.
5	DirectionVector	NVARCHAR	Вектор напрямку, паралельний напрямку чотирьох пальців.
6	MotionFactors	NVARCHAR	Фактори руху: відносні масштаби, обертання, коефіцієнти трансляції для руху
7	PalmSpeed	NVARCHAR	Швидкість долоні: швидкість руху руки користувача(одиниця: мм/с, вектор)
8	CapturePower	NVARCHAR	Сила захоплення
9	FinderId	INT	Вторинний ключ

Таблиця «Fingers» призначена для збереження даних про руки користувача, які були захоплені приладом. Містить наступні атрибути: ID, Name, Notes, Conclusion, Coefficient, GeneralSettings (таблиця 3.5).

Таблиця 3.5 – Атрибути таблиці «Fingers»

№ п/п	Назва	Тип даних	Опис
1	ID	INT	Первинний ключ Унікальний ідентифікатор
2	Position	NVARCHAR	Положення кінчиків пальців
3	Vector	NVARCHAR	Вектор кінчика пальця
4	Type	NVARCHAR	Тип пальця, включаючи великий, вказівний, середній, безіменний і мізинець

Таблиця «Users» призначена для збереження даних про користувача. Містить наступні атрибути: ID, FirstName, SecondName, Role, Notes, Diagnosis, Address, Login, Password, Email, Phone, DateBirth (таблиця 3.6).

Таблиця 3.6 – Атрибути таблиці «Users»

№ п/п	Назва	Тип даних	Опис
1	ID	INT	Первинний ключ Унікальний ідентифікатор
2	FirstName	NVARCHAR	Ім'я користувача
3	SecondName	NVARCHAR	Прізвище користувача
4	Role	NVARCHAR	Роль користувача, може бути: пацієнт, лікар, та адміністратор
5	Notes	NVARCHAR	Нотатки від лікаря, якщо роль користувача пацієнт
6	Diagnosis	NVARCHAR	Діагноз хвороби, у випадку якщо роль користувача пацієнт
7	Address	NVARCHAR	Поле для адреси
8	Login	NVARCHAR	Поле для логіну користувача
9	Password	NVARCHAR	Поле для пароля для користувача
10	Email	NVARCHAR	Поле для пошти користувача
11	Phone	NVARCHAR	Поле для номера телефону користувача
12	DateBirth	Date	Поле для запису дати народження користувача

### Висновки до розділу 3

В даному розділі описано проектування інформаційної системи визначення точності координації рухів кисті руки у просторі. При виконанні дотримано основні вимоги, а саме розроблено функціональну модель, в якій описано функціональні блоки з яких складається система, як відбувається процес обробки інформації, отриманні дані на вході, та дані які вийшли на виході.

Додатково описано об'єктно орієнтовану модель інформаційної системи, для виділення основних сутностей предметної області та визначення їх властивостей та

функцій. Також для даної моделі побудовано ER-діаграму та розроблено структури інформаційної системи.

## РОЗДІЛ 4

### Дослідження методу визначення точності координації рухів кисті руки

#### 4.1 Архітектура інформаційної системи

Для тестування методу визначення точності координації рухів кисті руки, як раніше було зазначено, буде реалізовано інформаційну систему у вигляді веб-додатку, за допомогою якого можна дослідити його ефективність.

Було вирішено розділити структуру програми на декілька основних модулів, кожен з яких буде відповідати за певну функцію, які були визначено раніше. Модуль для роботи з пристроєм Leap Motion, модуль визначення рухів, модуль проведення тестування, модуль аналізу тестування, а також не функціональний модуль інтерфейсу системи, який відповідає за взаємодію з користувачем, враховуючи особливості при використанні пристрою Leap Motion. Для даних модулів, сформовано наступну діаграму класів (рисунок 4.1), які будуть використовуватись при розробці веб-додатку.

На даній схемі є клас «User» з якого і починається весь процес, звертаючись до класу «Test». Клас «Test» являється основним, в якому і відбувається застосування методу визначення точності координації рухів. В ньому проводиться ініціалізація всіх компонентів та класів системи, відбувається безперервне зчитування рухів користувача, обробка цих даних. Також керує класом який відповідає за інтерфейс користувача - «Canvas», малювання основного полотна де відбувається тест та спіралі Архімеда - «Spiral» який використовується як основа перевірки точності рухів, відображення траєкторії - «Tracker» пройдені користувачем від початку до кінця тесту, та показ результату у вигляді коефіцієнта точності координації та графіка.

Клас «Analysis» відбувається всі розрахунки потрібні для методів, що використовуються в інформаційній системі: перетворення координат користувача в радіуси, застосування методу найменших квадратів, знаходження координат для

спіралі Архімеда за формулою, перевірка на зіткнення використовуючи метод обмежувального кола,

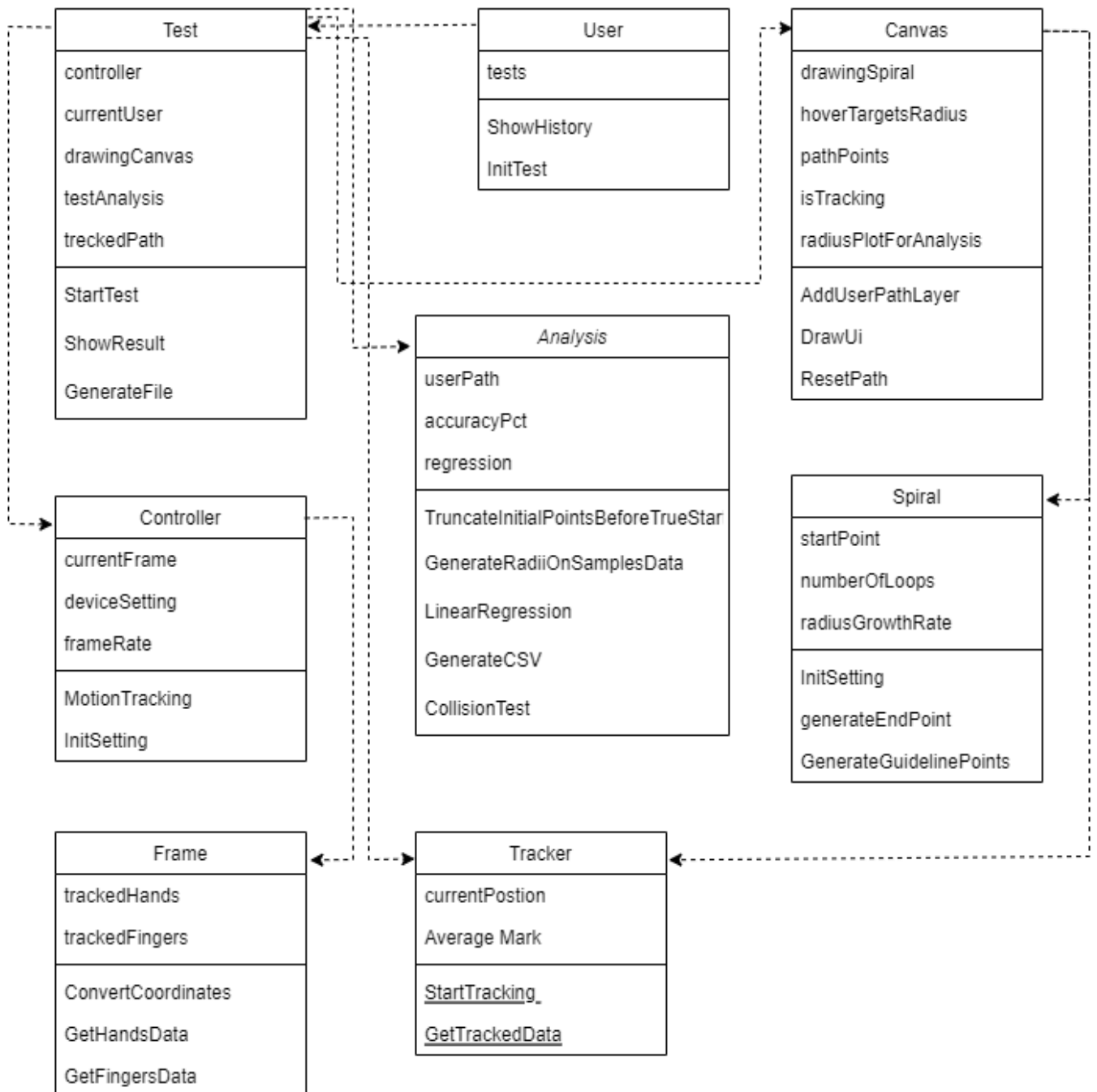


Рисунок 4.1 – Діаграма класів інформаційної системи

## 4.2 Розробка прикладних компонентів інформаційної системи

Для даної інформаційною системи визначення точності координації рухів кисті руки у просторі, основним являється саме захоплення рухів користувача, яке відбувається за допомогою пристрою Leap Motion, саме тому для цієї взаємодію потрібно для початку підключити API Leap Motion, це відбувається за допомогою простого коду, який потрібно додати до проекту, було обрано варіант для розробки:

- For production (minified):

```
<script src="//js.leapmotion.com/leap-0.6.3.min.js"></script>
```

- For development:

```
<script src="//js.leapmotion.com/leap-0.6.3.js"></script>
```

Після чого, з'являється доступ до даних з пристрою Leap Motion Controller, для цього потрібно створити цикл від API Leap Motion, щоб зчитування відбувалось безперервно:

```
// Setup Leap loop with frame callback function
var controllerOptions = {};

Leap.loop(controllerOptions, function(frame) {
  // Body of callback function
})
```

Від тепер можна отримати об'єкт «frame» який являється коренем ієрархії всіх об'єктів та зберігає дані про захоплені рухи здійсненні користувачем. Приклад використання отриманих даних у вигляді 3D моделі рук, які були захоплені пристроєм зображено на рисунку 4.2.

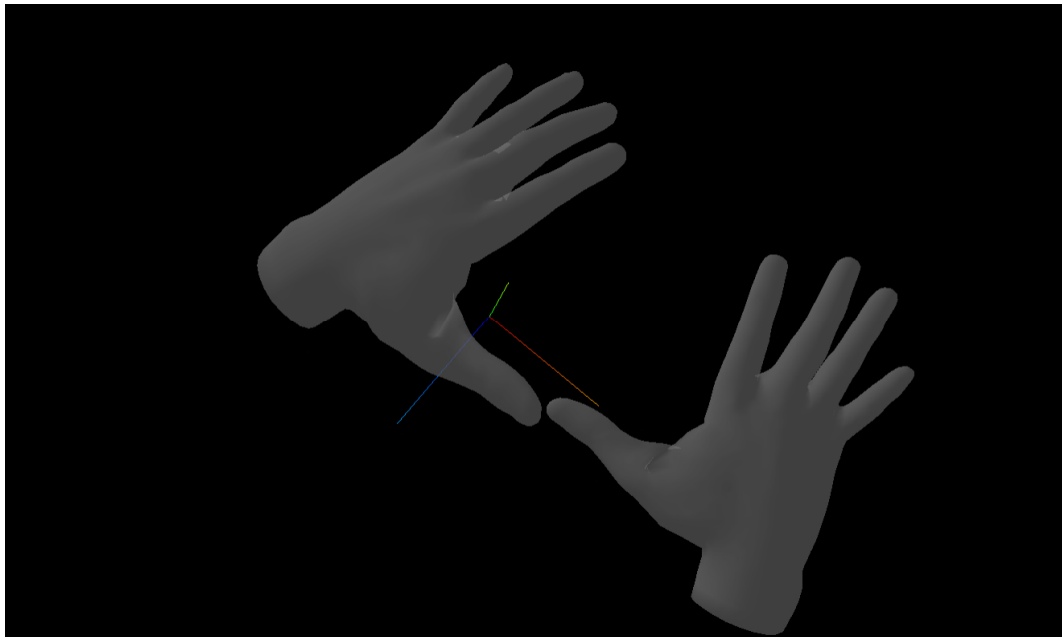


Рисунок 4.2 – Відображення 3D моделі рук за допомогою LMC

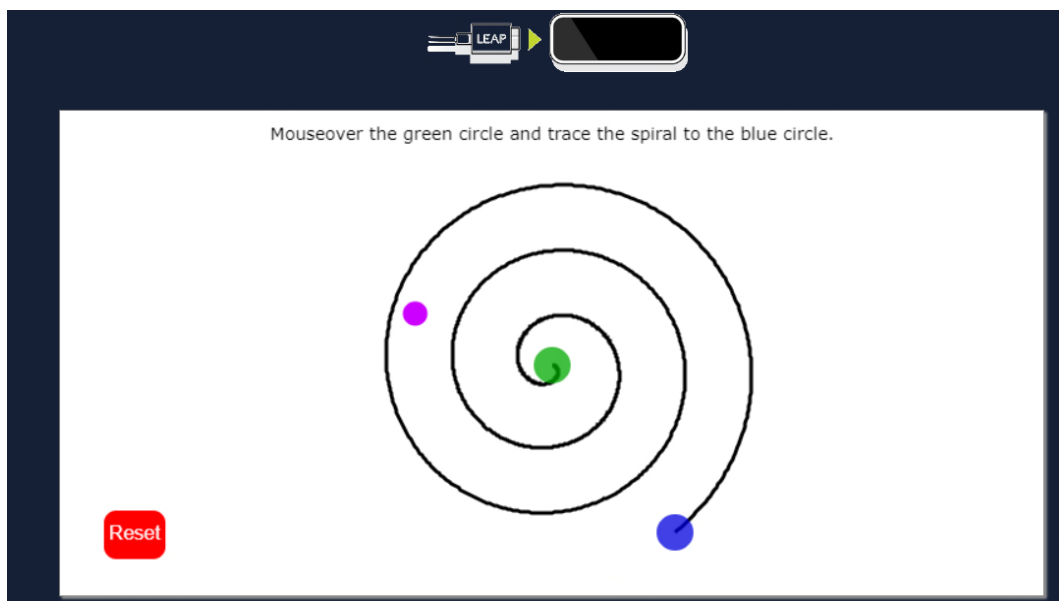


Рисунок 4.3 – Приклад використання «Canvas» в проєкті

Окрім того, також буде використовуватись бібліотека Canvas, яка підключається за схожим принципом, та дозволяє малювати форми, контури, зображення та інші малюнки на пустому елементі, який називається полотном. Ця

бібліотека буде використовуватися для малювання спіралі Архімеда, та графіків в проекті.

На рисунку 4.3 зображено полотно на якому намальовано додаткові елементи які потрібні для тесту, та саму спіраль, та перш ніж це зробити потрібно, знайти її координати та властивості, для цього створено об'єкт, в якому відбувається ініціалізація таких значення як початкова точка, кількість кругів, та швидкість зростання радіуса, отримавши ці дані, обраховується за рівнянням координати по яким буде будуватися спіраль. При цьому, так як рівняння спіралі Архімеда в полярній системі координат, ще потрібно перетворити знайдені точки в декартову систему координат, за все це відповідає наступний код:

```

this.startPoint = params.startPoint;
this.numberOfLoops = params.numberOfLoops;
this.radiusGrowthRate = params.radiusGrowthRate;

this.xyForPolar = function(coords){
    return {x: Math.round(coords.r * Math.cos(coords.angle)),
            y: Math.round(coords.r * Math.sin(coords.angle))};
};

this.generateGuidelinePoints = function(){
    var allPoints = {};

    var maxAngleInDegrees = this.numberOfLoops * 360;

    for(var currentDegrees = 0; currentDegrees <= maxAngleInDegrees; currentDegrees++){

        var currInRadians = currentDegrees * Math.PI / 180.0;
        var point = this.xyForPolar({r: this.radiusGrowthRate * currentDegrees,
                                     angle: currInRadians});
        allPoints['x'+(currentDegrees+1)] = point.x + this.startPoint.x;
        allPoints['y'+(currentDegrees+1)] = point.y + this.startPoint.y;
    }
    return allPoints;
};

```

Для початку тестування, користувачеві потрібно перетнути пальцем початок спіралі, так як вони представлення колами(рисунок 4.5), зелений для початку спіралі та фіолетовий для пальця, достатньо використати метод обмежувального кола для того, щоб перевірити чи відбувався перетин. Аналогічна перевірка виконується і для закінчення тесту. Реалізація методу:

```
function collisionTest(obj1,obj2){
  var sumOfRadii = obj1.radius + obj2.radius;
  var diffInX = obj2.x - obj1.x;
  var diffInY = obj2.y - obj1.y;
  var vectorMagnitude = Math.sqrt(diffInX*diffInX + diffInY*diffInY);

  return vectorMagnitude < sumOfRadii;
}
```

Підготувивши полотно для тесту, наступним буде реалізація основного функціоналу, в якому буде відбуватись безперервне зчитування рухів користувача за допомогою пристрою, з усіх даних обиратиметься лише координати пальця, які конвертуються в координати для відображення на полотні, тобто перетворення 3D координат в 2D, в процесі цього вони будуть зберігатись для подальшого аналізу:

```
////////////////////////////////////
// SETUP AND OBTAIN DATA FROM LEAP MOTION
////////////////////////////////////
Leap.loop({}, function(frame) {
  if(frame.pointables.length > 0){
    var pointable = frame.pointables[0];
    var interactionBox = frame.interactionBox;
    var normalizedPosition = interactionBox.normalizePoint(pointable.tipPosition, true);

    var pointerOnCanvas = {x: $('canvas').width() * normalizedPosition[0],
                          y: $('canvas').height() * (1 - normalizedPosition[1])};

    var leapCursorLayer = $('canvas').getLayer('leapCursor');

    if( ((Math.round(pointerOnCanvas.x) != Math.round(leapCursorLayer.x)) ||
        (Math.round(pointerOnCanvas.y) != Math.round(leapCursorLayer.y))) &&
        isTracking == false) {

      isTracking = collisionTest(leapCursorLayer, $('canvas').getLayer('startCircle'));

    }else if(isTracking == true && collisionTest(leapCursorLayer, $('canvas').getLayer('targetCircle')) == true){

      isTracking = false;
      var analysis = new Analysis(radiusPlotForAnalysis);
      analysis.printResults();
    }
  }
}
```

На етапі аналізу, якраз застосовується метод визначення точності координації рухів кисті руки, а саме метод найменших квадратів, після розрахунку якого в результаті буде отримано коефіцієнт детермінації – це значення і буде оцінкою точності координації рухів. В кодї нижче реалізовано цей метод:

```

this.leastSquares = function(data){
  var lr = {};
  var n = data.length;
  var sum_x = 0;
  var sum_y = 0;
  var sum_xy = 0;
  var sum_xx = 0;
  var sum_yy = 0;

  for (var i = 0; i < data.length; i++) {
    sum_x += data[i].sample;
    sum_y += data[i].radius;
    sum_xy += (data[i].sample * data[i].radius);
    sum_xx += (data[i].sample * data[i].sample);
    sum_yy += (data[i].radius * data[i].radius);
  }

  lr.slope = (n * sum_xy - sum_x * sum_y) / (n*sum_xx - sum_x * sum_x);
  lr.intercept = (sum_y - lr.slope * sum_x)/n;
  lr.r2 = Math.pow((n*sum_xy - sum_x*sum_y)/Math.sqrt((n*sum_xx-sum_x*sum_x)*(n*sum_yy-sum_y*sum_y)),2);
  lr.fn = function (x) { return this.slope * x + this.intercept; };

  return lr;
};

```

Після завершення тесту, виводяться дані результатів, а саме відображається коефіцієнт детермінації у вигляді відсотків, створення csv файлу в якого заносяться дані, та графік:

```

this.printResults = function(){
  var accuracyPct = Math.round(this.regression.r2 * 100); // rounds R^2
  $('#results').html("Accuracy: " + accuracyPct + "%");
  this.showChart( this.xDots, this.yDots, this.chartData);
  this.generateCSV(this.radiiData);
};

```

Після реалізації всіх компонентів описаних вище, можна повноцінно проводити тестування на визначення точності координації рухів кисті руки, підключивши пристрій Leap Motion, результат інтерфейсу можна переглянути на рисунку 4.4. Приклад отриманого csv файлу, а також графік на якому можна оцінити проходження всього тесту показано на рисунку 4.5.

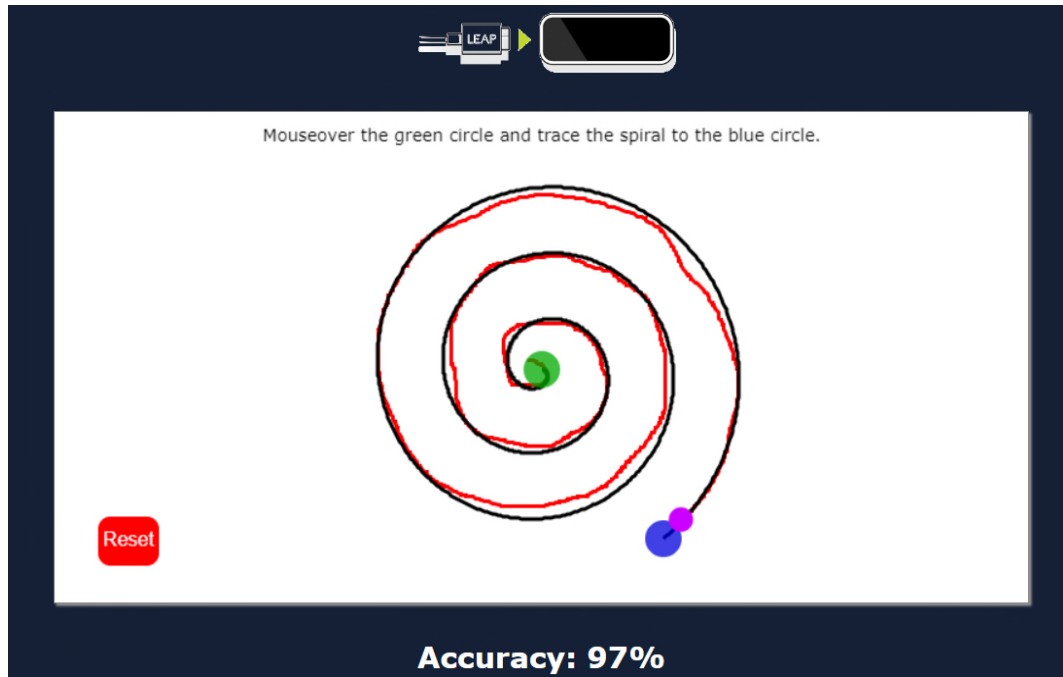


Рисунок 4.4 – Приклад успішно пройденого тесту

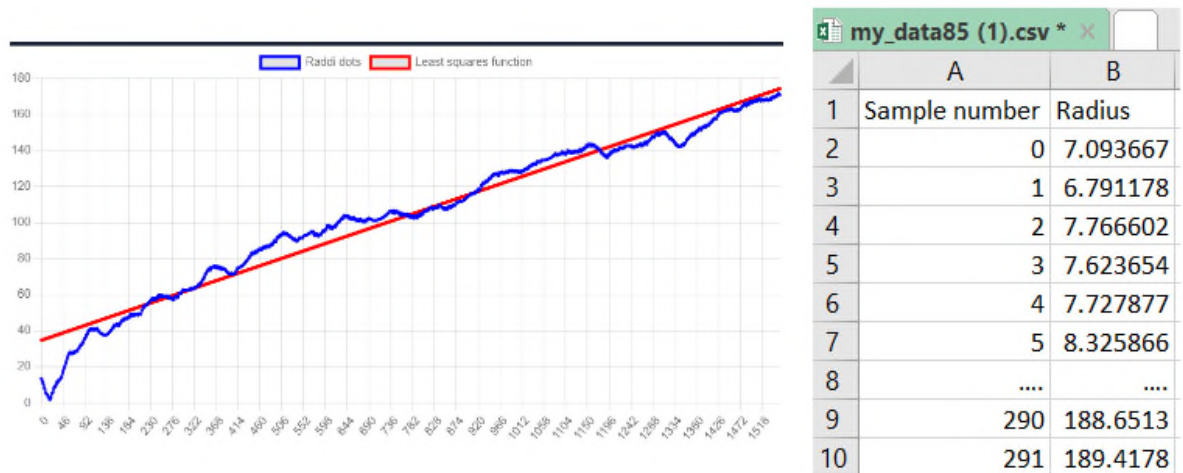


Рисунок 4.5 – Графік, та вміст csv файлу, що демонструють результати отримані під час тесту

### 4.3 Прикладне тестування інформаційної системи

Щоб провести дослідження на справність роботи інформаційної системи визначення точності координації рухів кисті руки у просторі, було створено тест-кейси, для перевірки базового функціоналу. Для початку слід протестувати чи

правильно працює веб-додаток, та користувач може з ним взаємодіяти. Після запуску веб-застосунку потрібно перейти за посиланням в браузері, у випадку справної роботи, буде загружено головну сторінку, в даному випадку панель приладів, інакше браузер покаже повідомлення про помилку (рисунк 4.6-4.7).

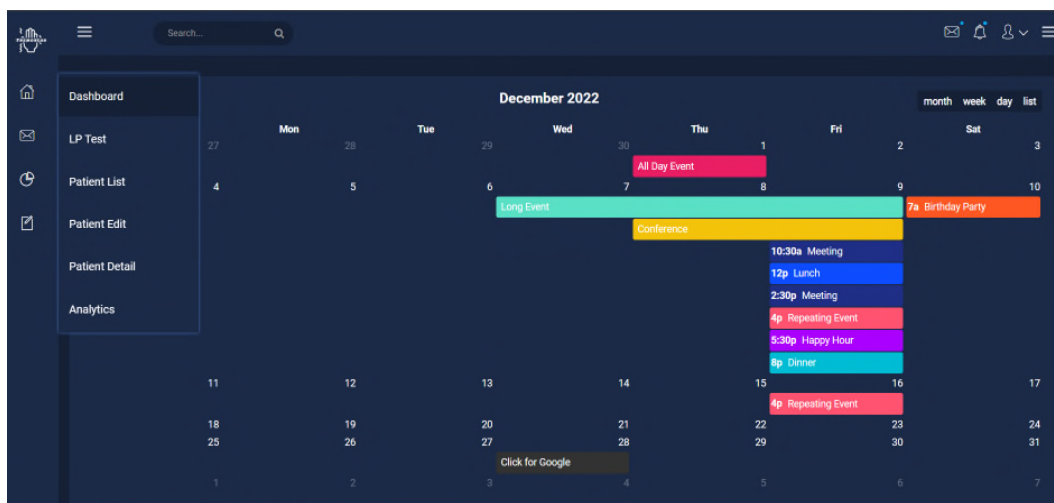


Рисунок 4.6 – Головна сторінка веб-застосунку

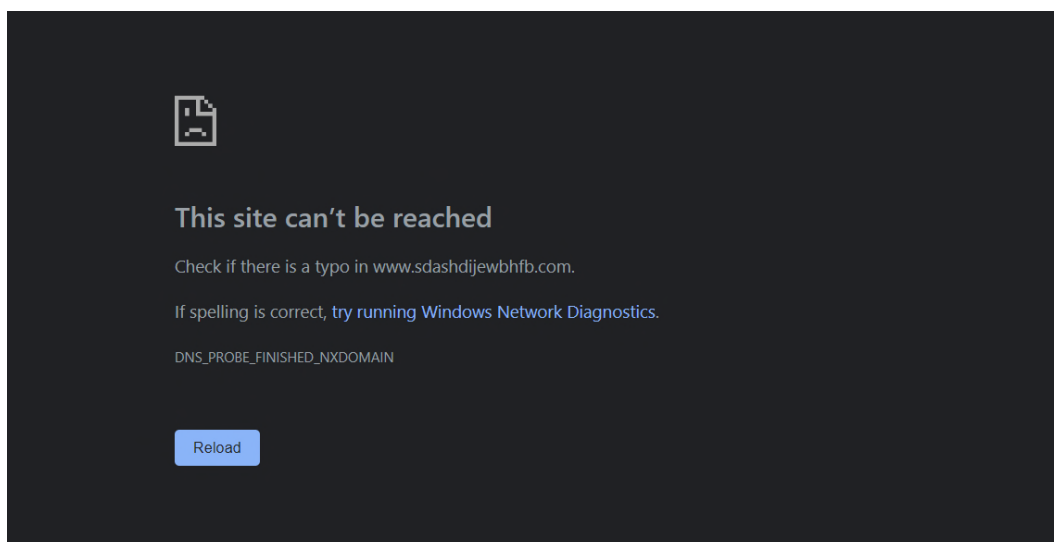


Рисунок 4.7 – Помилка при загрузці веб-застосунку

Таблиця 4.1 – Тест-кейс 01

Тест-кейс №: 1	Пріоритет: 1	Створено: 03.12.2022, Яницький О. О.
Назва: Перевірка на запуск веб-додатка		

Кроки	Очікуваний результат
<ol style="list-style-type: none"> <li>1. Запустити веб-додаток</li> <li>2. Перейти за посиланням на головну сторінку</li> <li>3. Порівняння отриманого результату з очікуваним</li> </ol>	Головна сторінка веб-додатку загрузено
Результат проходження тест-кейсу: успішний	

Оскільки для справної роботи з системою користувачу потрібно використовувати Leap Motion, важливо протестувати справність самого пристрою та чи він підключений до комп'ютера, а також чи працює API для нього. Після запуску веб-застосунку, в разі успішного підключення пристрою, та справної його роботи на полотні де зображено спіраль, палець користувача буде відображено у вигляді фіолетового кола, а також подальші інструкції для тестування (рисунок 4.3), в іншому випадку буде відображено повідомлення про помилку, та несправність всього функціоналу для проведення тесту(рисунок 4.8).

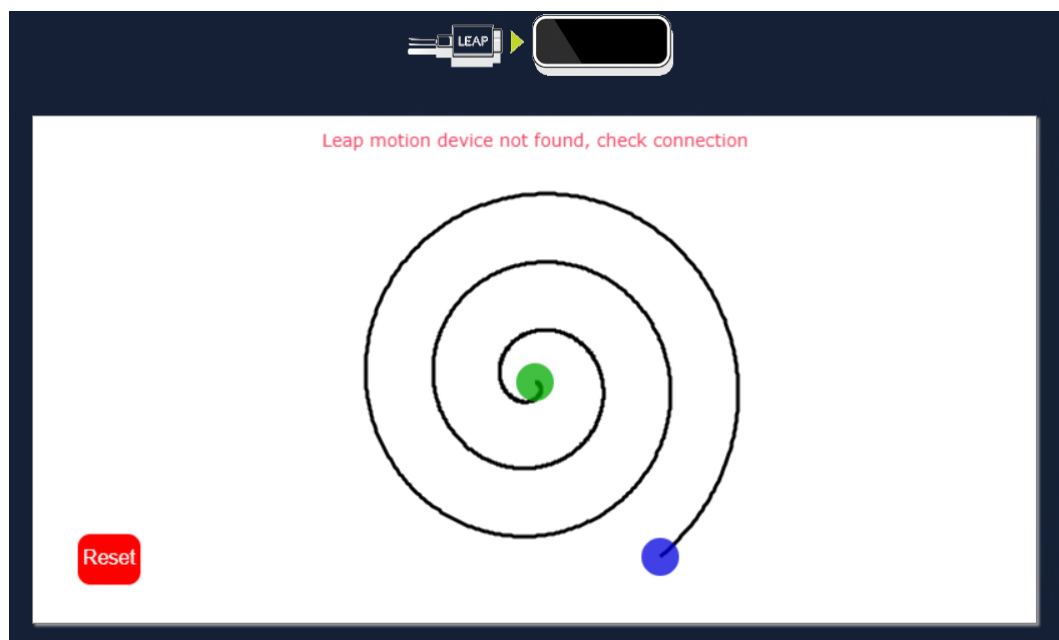


Рисунок 4.8 – Повідомлення про несправне підключення пристрою

Таблиця 4.2 – Тест-кейс 02

Тест-кейс №: 2	Пріоритет: 1	Створено: 03.12.2022, Яницький О. О.
Назва: Перевірка підключення пристрою Leap Motion		
Кроки		Очікуваний результат
1. Запустити веб-додаток 2. Перейти на сторінку з тестуванням, та зачекати її загрузки та самого полотна де відбудуватиметься тест 3. Порівняння отриманого результату з очікуваним		Пристрій підключено правильно, на полотні відображено фіолетове коло у випадку рухів користувача пальцем.
Результат проходження тест-кейсу: успішний		

Дальше потрібно перевірити запуск самого тесту, його виконання, та його закінчення. Коректна робота вважається, якщо палець відображено на полотні у вигляді фіолетового кола, та при перетині зеленого кола, що є початком спіралі, запуститься тестування, і всі подальші пересування пальця будуть залишати слід червоною лінією до моменту завершення тесту, а саме перетину фіолетового кола – кінця спіралі, успішну роботу вже було продемонстровано раніше на рисунку 4.4.

Таблиця 4.3 – Тест-кейс 03

Тест-кейс №: 3	Пріоритет: 1	Створено: 03.12.2022, Яницький О. О.
Назва: Перевірка відпрацювання тесту		
Кроки		Очікуваний результат
1. Запустити веб-додаток 2. Перейти на сторінку з тестуванням 3. На сторінці тесту, за допомогою рухів пальця перетнути зелене коло 4. Провести палець далі та перетнути фіолетове коло		При наведенні пальця за допомогою пристрою на полотно, появляється фіолетове коло, при подальшому русі та перетину зеленого кола, розпочинається тестування, та залишається червоний слід. Після перетину фіолетового кола, тест завершуються.

5. Порівняння отриманого результату з очікуваним	
Результат проходження тест-кейсу: успішний	

Основні кейси для взаємодії користувача з інтерфейсом, які потрібні для роботи методу були описані, тепер буде написано тест-кейс для перевірки роботи самого методу для перевірки точності координації рухів. При його перевірці також буде покрито інші тест-кейси які мають менший пріоритет, та не були описані окремо, а саме коректне відображення результату, графіків, генерування csv файлу. Для перевірки точності використовується спіраль Архімеда, а саме потрібно відтворити її точну форму у просторі за допомогою пальця, чим більша точність, тим більший результат, та на графіку зображено менше відхилення. На рисунку 4.4 можна переглянути високий результат при правильному відтворенні форми спіралі, та невелике відхилення на графіку (рисунок 4.5). Коли ж форма не відповідає спіралі (рисунок 4.9), результат точності рухів, низький, також можна спостерігати велике відхилення на графіку (рисунок 4.10),

Place your index finger on the green circle and spiral to the blue circle

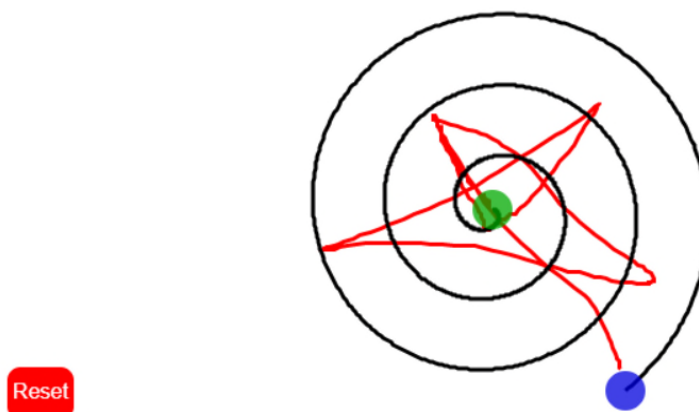


Рисунок 4.9 – Невдале проходження тестування

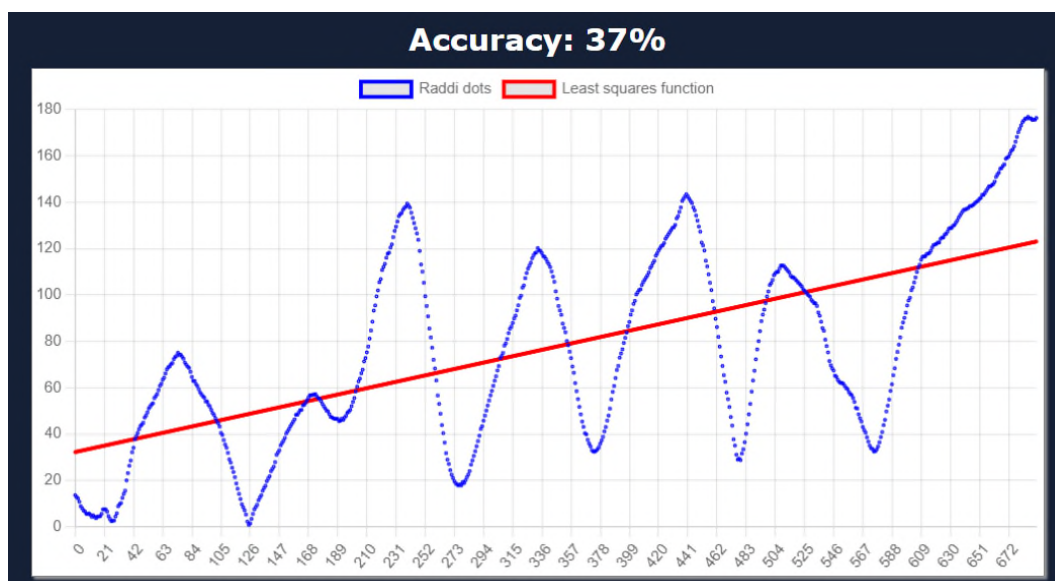


Рисунок 4.10 – Результати невдалого тестування

Таблиця 4.4 – Тест-кейс 04

Тест-кейс №: 4	Пріоритет: 1	Створено: 03.12.2022, Яницький О. О.
Назва: Перевірка роботи методу		
Кроки	Очікуваний результат	
<ol style="list-style-type: none"> <li>1. Запустити веб-додаток</li> <li>2. Перейти на сторінку з тестуванням</li> <li>3. Розпочати тестування</li> <li>4. Повторити форму спіралі</li> <li>5. Завершити тестування</li> <li>6. Перевірити результати</li> </ol>	Після проходження тестування, якщо форма яку повторив користувач, відповідає формі намальованої спіралі, або схожа, то отриманий результат повинен бути більше 80%, в іншому випадку менше 50%	
Результат проходження тест-кейсу: успішний		

#### 4.4 Функціональне дослідження інформаційної системи

Для дослідження ефективності застосування методу інформаційної системи, а саме метод визначення точності координації рухів кисті руки, було проведено тестування на різних користувачах та проаналізовано їхні результати. Обрано три приклади, для демонстрації роботи метода, а саме нормальний результат людини без проблем, користувача з порушенням координації рухів, і останній з сильним

тремором рук. Для порівняння і оцінювання цих результатів, використовувалось, коефіцієнт детермінації, значення, яке відображається в кінці тестування, та відповідає за точність рухів; графіки, які будуються на основі даних під час тестування; табличні дані, які були занесені до csv файлу.

Спочатку розглянемо приклад, тестування користувача з порушенням координації, на рисунку 4.11 зображено його результат точності рухів, неозброєним оком можна замітити явну проблему, при обведенні спіралі, відповідно отриманий результат 68%, що є близькою до реальності, але ще потрібно порівняти з іншими випадками.

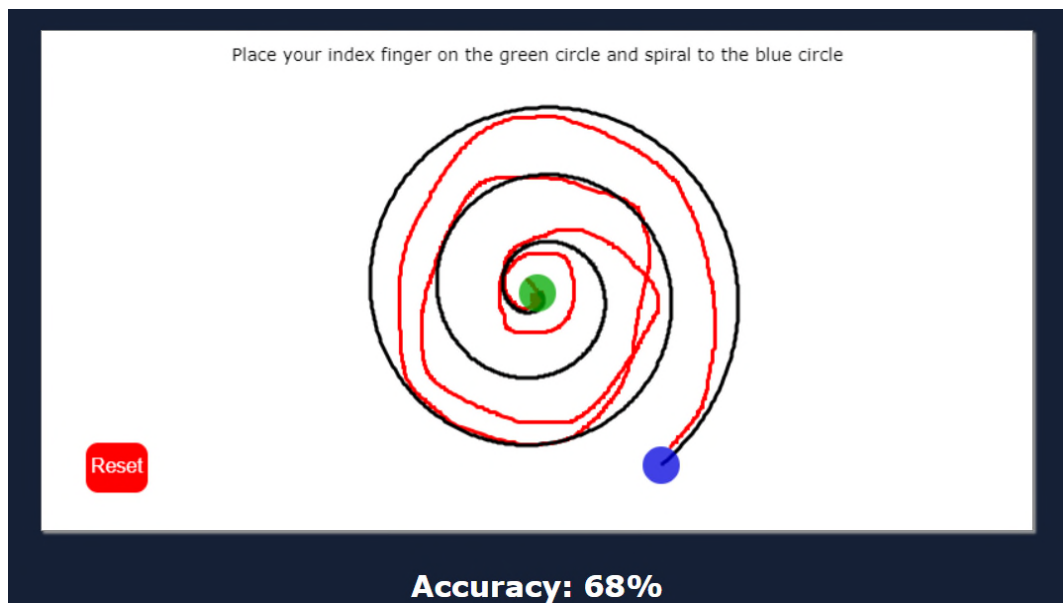


Рисунок 4.11 – Результати користувача з порушенням координації рухів

Ще розглянемо результати отриманих графіків, які показують додаткові дані оцінювання. На рисунку 4.12 зображено графік підігнаної лінії регресії використовуючи дані отримані після застосування методу найменших квадратів. Це лінія, яка найкраще підходить для заданих спостережених даних, дані в системі, це радіуси, які були отримані з шляху пройденим користувачем. Ці дані виведені на наступному графіку (рисунок 4.13).

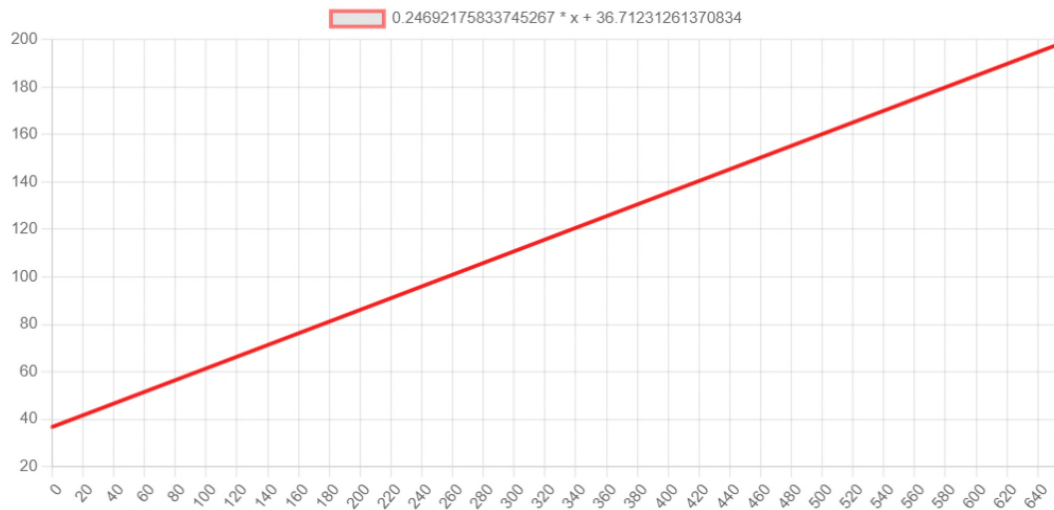


Рисунок 4.12 – Графік підігнаної лінії регресії

Об'єднанні графіки можна спостерігати на рисунку 4.14, на якому можна замітити, як лінія проходить через точки даних і чим ближче точки даних будуть знижуватись до підігнаної лінії регресії тим більший буде коефіцієнт детермінації. На даному графіку, лінія проходить крізь малу кількість точок, що і вказує на малий відсоток точності координації.

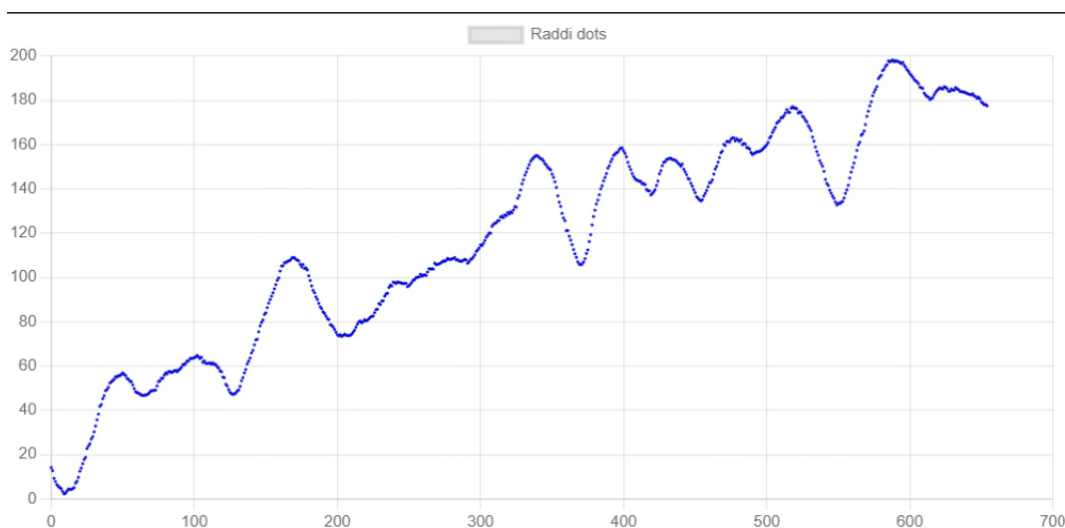


Рисунок 4.13 – Графік спостережених даних (радіусів)

Також можна замітити на графіку спостережних даних, точки будують криву, яку можна розцінювати як відхилення від намальованої спіралі, та визначити наскільки коливалась рука користувача під час тесту.

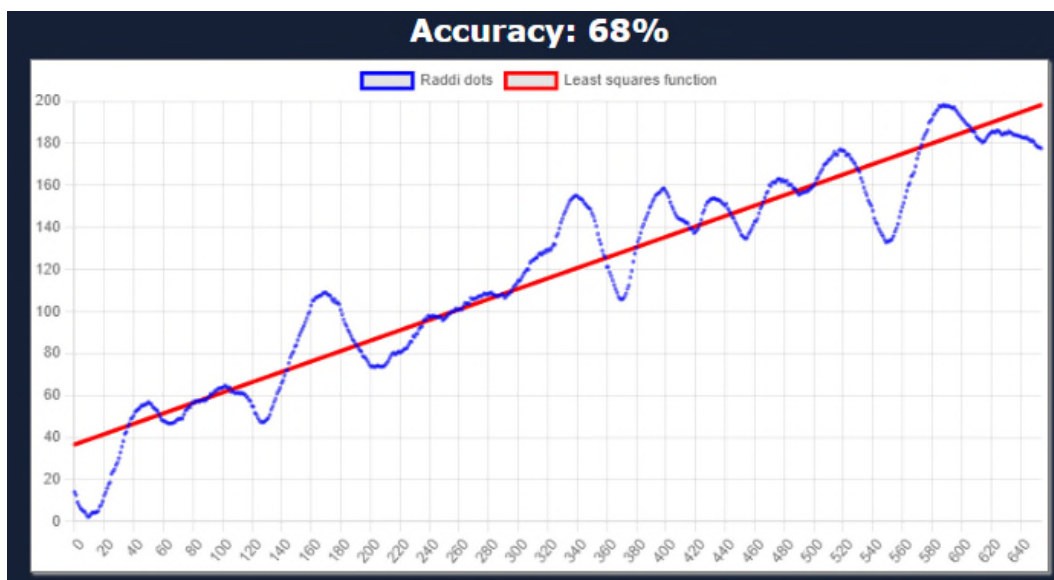


Рисунок 4.14 – Об'єднанні графіки

Наступні результати отримані від користувача, в якого немає проблем з координацією (рисунок 4.15), на рисунку 4.16 його пройдений шлях, що підтверджує попередні результати та їх оцінка.

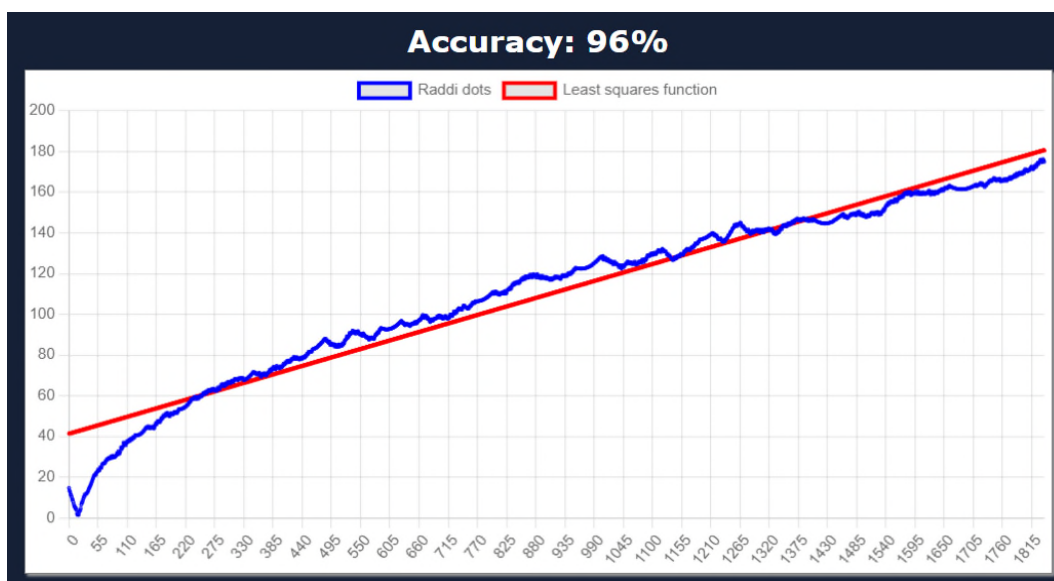


Рисунок 4.15 – Результати користувача з нормальною координацією

Чим менше відхилення від спіралі, тим більше точок даних пронизує лінія регресії та відповідно високий коефіцієнт точності. В теорії якщо користувача відтворить ідеальну спіраль, то на графіку лінія регресії проходила б крізь всі точки, що означитиме 100% коефіцієнт точності (рисунок 4.17), штучно створений графік. Останнє що варто підмітити з даного результату, те що коливання руки користувача були не значними, що видно по лінії спостережних даних.

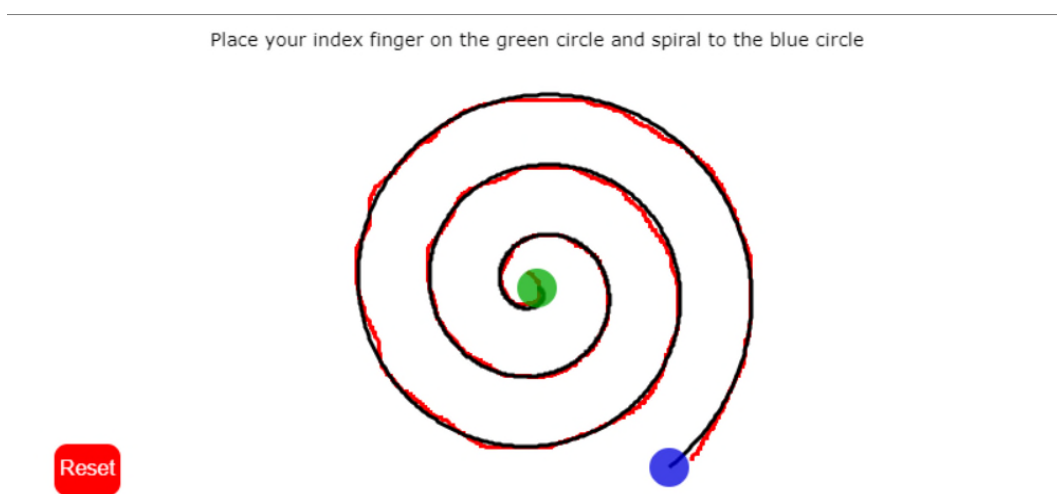


Рисунок 4.16 – Шлях користувача з нормальною координацією

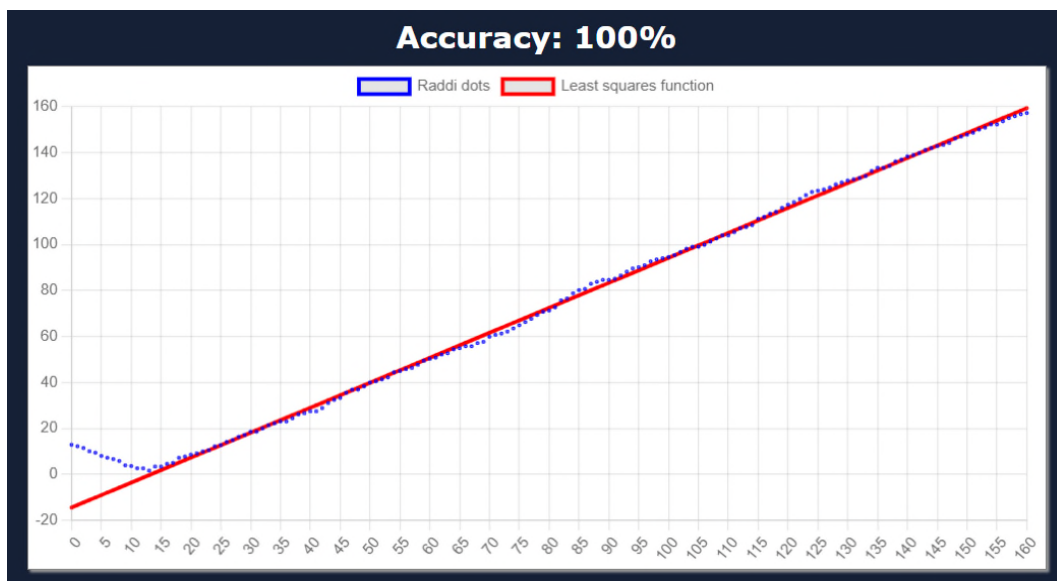


Рисунок 4.17 – Штучно створений графік з відмінним результатом

Останнім прикладом буде користувач з тремором рук, дещо унікальний випадок, так як на результатах видно (рисунок 4.18) високу оцінку точності координації – 94% лише на 2% менше, ніж у користувача у якого немає проблем з координацією, проте якщо переглянути його намальований шлях (рисунок 4.19), можна спостерігати скривлену спіраль, в наслідок сильного тремору рук під час тесту.

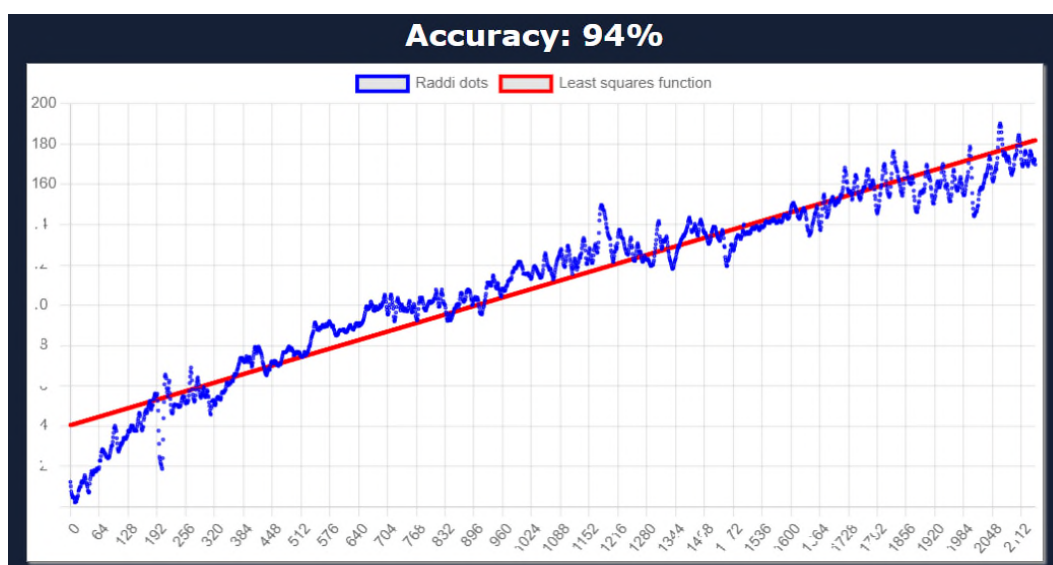


Рисунок 4.18 – Результати користувача з тремором рук

Це наслідок того що використовуваний метод, визначає точність лише по одній ознаці, це збільшення радіусів з пройденим шляхом, і з графіку видно, що лінія регресії проходить крізь велику кількість точок, відповідно і оцінка точності висока. Но з графіку, як і з попередніх результатів з лінії спостережних даних можна замітити частоту коливань рухів при проходженні тесту, і при сильному треморі частота дуже висока.

Тобто якщо визначити цю характеристику, та додати її при оцінці, можна покращити метод, автоматизувавши визначення такої проблеми з координацією як тремор, щоб не було необхідності у спеціалісті.

Place your index finger on the green circle and spiral to the blue circle

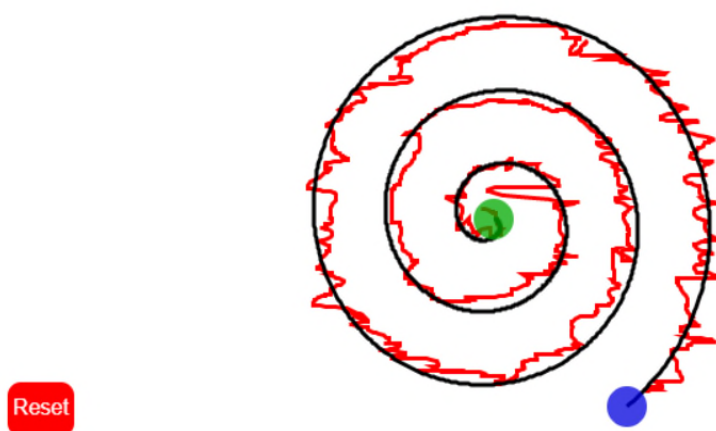


Рисунок 4.19 – Шлях користувача з тремором рук

Отже, в дослідженні показано, що метод працює правильно при певних випадках з порушенням координації, хоча є недоліки які можна покращити, метод виконує поставлене завдання, проте все ще потребує наявності спеціаліста в деяких кейсах. Додатково додано табличні дані які записані у csv файли під час трьох тестувань.

Таблиця 4.1 – Табличні дані з csv файлів

Результати користувача з порушенням координації рухів		Результати користувача з тремором рук		Результати користувача з нормальною координацією	
Номер вибірки	Радіус під час вибірки	Номер вибірки	Радіус під час вибірки	Номер вибірки	Радіус під час вибірки
1	0.721114064	1	1.456014432	1	1.523151415
2	1.523154922	2	1.979890356	2	2.952967092
3	2.473862635	3	2.77848045	3	3.862645059
4	3.452534107	4	2.433097988	4	5.772352317
5	4.440719219	5	2.473858934	5	6.746855822
6	5.433229482	6	2.884439328	6	7.727876531
...	...	...	...	...	...
1460	163.9820663	839	169.5656719	296	181.016897
1461	163.2982489	840	168.8630128	297	180.1480419
1462	162.6791877	841	169.6010527	298	179.2805539
1463	163.4206783	842	169.6364261	299	178.414453
1464	163.5002083	843	169.0050801	300	177.5497593
1465	163.5919256	844	169.8131829	301	176.6864937
1466	162.9371601	845	169.235685	302	175.8246771

## Висновки до розділу 4

В четвертому розділі виконано дослідження методу визначення точності координації рухів кисті руки, який використовується в інформаційній системі. В процесі його виконання, розроблено та описано особливості архітектури самої інформаційної системи у вигляді діаграма класів.

Описано прикладні компоненти інформаційної системи та їх функціонал, також продемонстровано реалізацію цих компонентів у вигляді програмного коду. Окремо розглянуто особливості при роботі з LMC, та використання API Leap Motion для обробки даних з пристрою

Проведено прикладне тестування інформаційної системи визначення точності координації рухів кисті руки у просторі, в результаті чого створено ряд тест-кейсів для перевірки справності інтерфейсу з яким взаємодіє користувач та функціональної частини веб-додатку. Також створено тест-кейс для перевірки самого методу, в результаті успішно підтверджено справність всіх компонентів

В кінці описано функціональне тестування, яке проводилось у вигляді дослідженні результатів отриманих під час тестування, детально описано для трьох випадків з різним станом здоров'я користувача, і хоча в результаті виявлено недоліки які можна покращити, основну роботу яку повинен виконувати метод було підтверджено.

## Загальні висновки

В результаті кваліфікаційної роботи магістра розроблено інформаційну систему в якій використовується метод, що відстежує шлях пальця за допомогою пристрою Leap Motion визначає точність координації рухів кисті руки у просторі.

В процесі розробки інформаційної системи виконано наступні задачі:

– Досліджено технології захоплення та відстеження рухів кисті руки. При розробці інформаційної системи для захоплення та відстежування рухів обрано пристрій, який працює на принципі безмаркерних технологій Leap Motion.

– Проведено аналіз методів виявлення перетину та відстані між 2D об'єктами, в результаті чого обрано, метод обмежувального кола, який найкраще підходить для системи, через те що являється більш простішим, але при цьому найшвидшим для обрахунку.

– Вдосконалено метод відстеження шляху пальця користувача та відображення його у 2D просторі, що дозволило взаємодіяти з віртуальними об'єктами в режимі реального часу.

– Вперше розроблено метод визначення точності координації рухів кисті руки, за допомогою спіралі Архімеда як виду тестування та методу найменших квадратів, для розрахунків.

– Розроблено веб-додаток в якому реалізовано весь функціонал визначений для інформаційної системи, та реалізовано основний метод для визначення точності координації рухів за допомогою Leap Motion.

– Проведено валідацію розробленого метода у вигляді тестування створеної інформаційної системи різними користувача, продемонстровано три приклади, на основі їх результатів, зроблено висновки, що метод працює правильно, та задовольняє поставленні цілі. Також можливе удосконалення методу, шляхом врахування додаткових показників при розрахуванні коефіцієнта точності, що покращить результати, та дозволить визначати рідкісні випадки проблем з координацією рухів.

## Перелік посилань

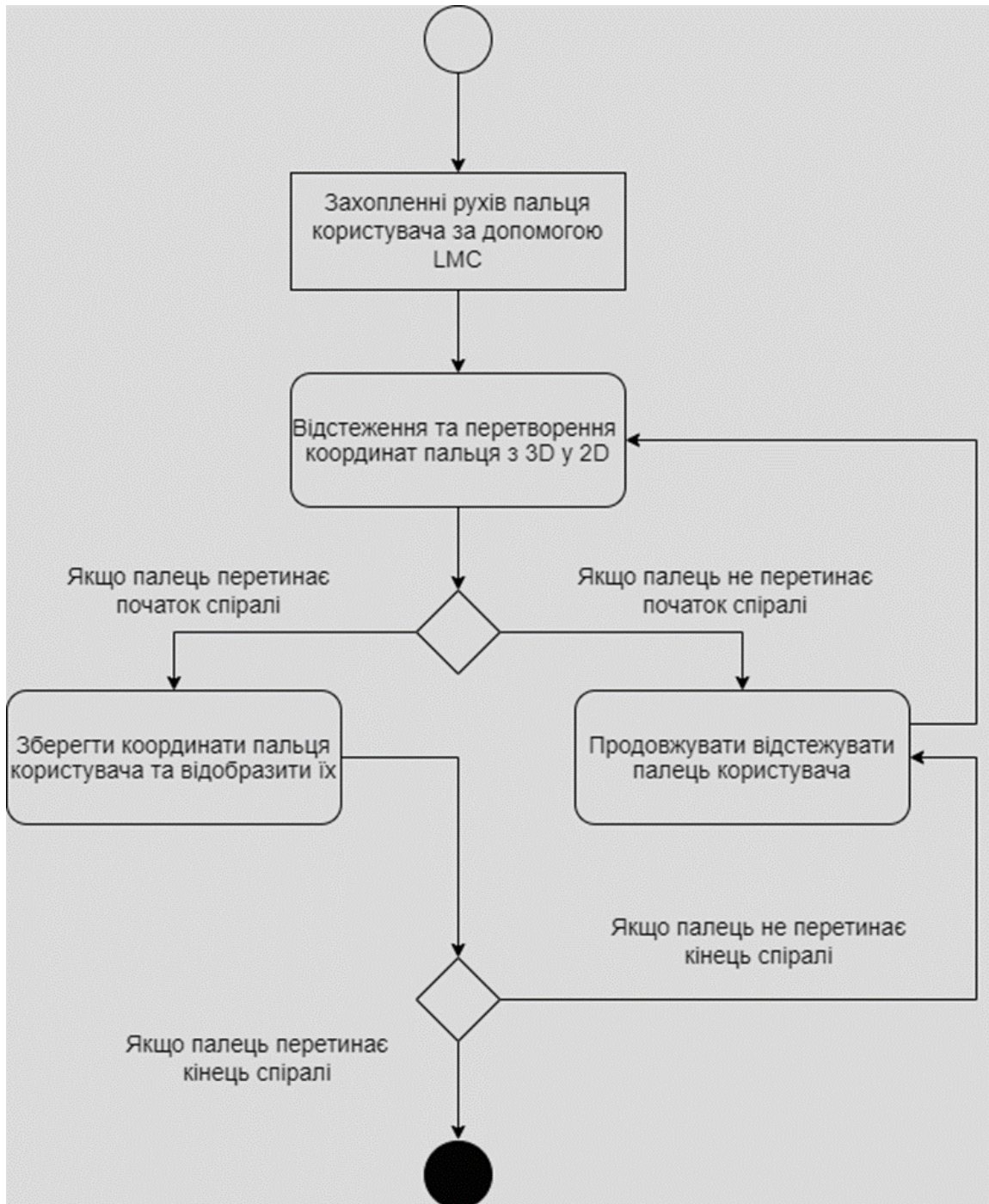
1. Human Tremor: Origins, Detection and Quantification URL: <https://www.intechopen.com/chapters/41790>
2. Technology in Parkinson's disease URL: <https://movementdisorders.onlinelibrary.wiley.com/doi/abs/10.1002/mds.26642>
3. Digitalized spiral drawing in Parkinson's disease: A tool for evaluating beyond the written trace URL: <https://www.sciencedirect.com/science/article/abs/pii/S0167945718300083>
4. Distinguishing Different Stages of Parkinson's Disease URL: <https://www.frontiersin.org/articles/10.3389/fneur.2017.00435/full?ref=https://githubhelp.com>
5. Diagnosis and Management of Tremor URL: <https://jamanetwork.com/journals/jamainternalmedicine/fullarticle/485435>
6. Human-computer interaction URL: [https://en.wikipedia.org/wiki/Human%E2%80%93computer\\_interaction](https://en.wikipedia.org/wiki/Human%E2%80%93computer_interaction)
7. Comparison of Markerless and Marker-Based Motion Capture Technologies URL: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0087640>
8. Leap Motion URL: <https://www.ultraleap.com/product/leap-motion-controller/>
9. 2D collision detection URL: [https://developer.mozilla.org/en-US/docs/Games/Techniques/2D\\_collision\\_detection](https://developer.mozilla.org/en-US/docs/Games/Techniques/2D_collision_detection)
10. Collision Detection URL: <https://www.jeffreythompson.org/collision-detection/circle-circle.php>
11. Review of Three-Dimensional Human-Computer Interaction with Focus on the Leap Motion Controller URL: <https://www.mdpi.com/1424-8220/18/7/2194>
12. API Overview URL: [https://developer-archive.leapmotion.com/documentation/csharp/devguide/Leap\\_Overview.html#motion-tracking-data](https://developer-archive.leapmotion.com/documentation/csharp/devguide/Leap_Overview.html#motion-tracking-data)

13. Coordinate Systems URL: [https://developer-archive.leapmotion.com/documentation/javascript/devguide/Leap\\_Coordinate\\_Mapping.html](https://developer-archive.leapmotion.com/documentation/javascript/devguide/Leap_Coordinate_Mapping.html)
14. Archimedean spiral URL: [https://en.wikipedia.org/wiki/Archimedean\\_spiral](https://en.wikipedia.org/wiki/Archimedean_spiral)
15. Spiral Analysis in Subjects with Parkinson's Disease before and after Levodopa Treatment URL: <https://journals.sagepub.com/doi/full/10.5301/jabfm.2012.9265>
16. Improved Spiral Test Using Digitized Graphics Tablet for Monitoring Parkinson's Disease URL: <https://doi.org/10.13140/RG.2.1.1898.6005>
17. Least Square Method URL: <https://www.cuemath.com/data/least-squares/>
18. Coefficient of Determination URL: <https://www.scribbr.com/statistics/coefficient-of-determination/>
19. Яницький О.О., Багрій Р.О., Скрипник Т.К. визначення точності координації рухів кисті руки у просторі. Збірник наукових праць за матеріалами XIV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2022». - Хмельницький, 2022. - С. 261-264

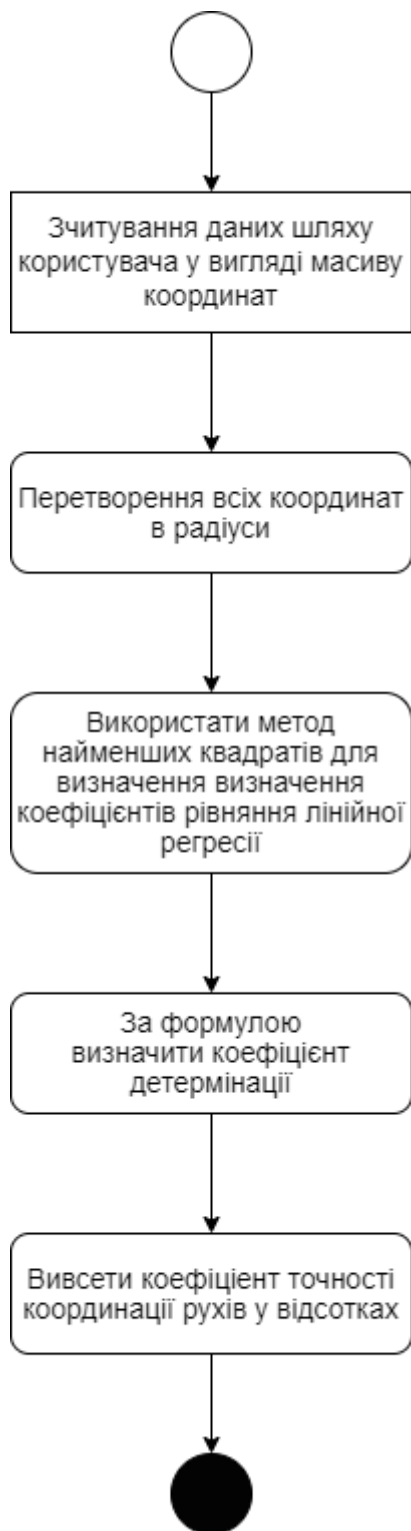
# ДОДАТКИ

## Додаток А

Схема роботи методу визначення точності координації рухів кисті руки.

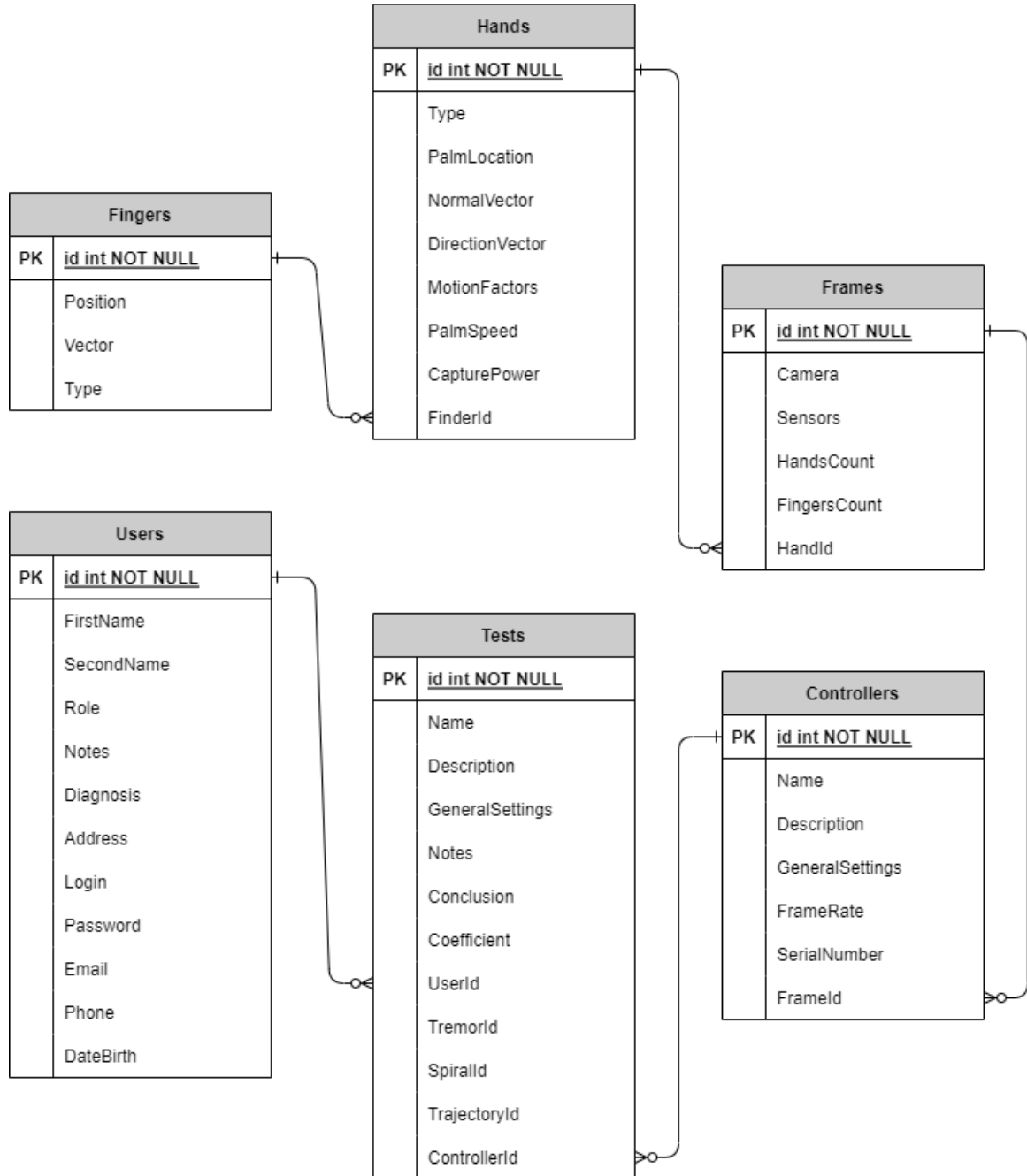


**Схема роботи методу визначення точності координації рухів кисті руки.**



# Додаток Б

## Інфологічна модель бази даних



## **Додаток В**

### **Світлини наукових публікацій, виконаних при роботі над кваліфікаційною роботою магістра**

*(ксерокопії титульної сторінки, сторінки змісту та всіх сторінок із публікацією)*

#### **Перелік наукових публікацій:**

Яницький О.О., Багрій Р.О., Скрипник Т.К. визначення точності координації рухів кисті руки у просторі. Збірник наукових праць за матеріалами XIV Всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2022». - Хмельницький, 2022. - С. 261-264

Міністерство освіти і науки України  
Хмельницький національний університет



**ЗБІРНИК НАУКОВИХ ПРАЦЬ**  
за матеріалами XIV Всеукраїнської науково-практичної конференції  
«Актуальні проблеми комп'ютерних наук АПКН-2022»

*18-19 листопада 2022*

Хмельницький 2022

УДК 004.912

Яницький О.О., Багрій Р.О., Скрипник Т.К.

*Хмельницький національний університет*

## **ВИЗНАЧЕННЯ ТОЧНОСТІ КООРДИНАЦІЇ РУХІВ КИСТІ РУКИ У ПРОСТОРИ**

*Розглянуто методи визначення точності координації рухів кисті руки з використанням технології захоплення рухів, проведено аналіз технології захоплення та відстеження рухів кисті руки, а також аналіз методів виявлення перетину та відстані між 2D об'єктами.*

*Investigated methods of determining the accuracy of the coordination of hand movements using motion capture technology was considered, an analysis of the technology of capturing and tracking hand movements was performed, as well as an analysis of methods for detecting the intersection and distance between 2D objects.*

**Вступ** Людський тремор є проблемою охорони здоров'я, з якою зіштовхується весь світ. Витрати, пов'язані з медичними та соціальними аспектами, необхідними для діагностики та лікування тремору, постійно зростають протягом останніх десятиліть.

Існує кілька способів вимірювання людського тремору [1]. Однак навіть сьогодні найбільш найпоширенішими є методи, які передбачають використання індикатора ступеня тяжкості. У цих методах пацієнтів просять виконати різні малюнки, такі як спіралі, кола та літери. Згодом неврологи класифікують ці малюнки за допомогою візуальних рейтингових шкал. Однак такі шкали забезпечують лише грубі суб'єктивні оцінки амплітуди тремору. Щоб зменшити суб'єктивність і обмеження деяких методів, заснованих на візуальних шкалах, було розроблено кілька стратегій електронного вимірювання тремору, а саме застосування таких приладів як акселерометр та цифрового планшету [2].

Використання цифрових планшетів є поширеним і забезпечує можливість виявлення активності тремору в кінетичних умовах [3]. Звичайною функцією планшета є можливість аналізу креслень безпосередньо на комп'ютері. Вимірювання тремору за допомогою цифрових планшетів є альтернативою виявлення тремору, що не потребують введення в організм інструментів, яка поєднує в собі простоту з точністю та універсальністю обчислювальних методів.

Однак проблема з використанням таких методів полягає в тому, що форма руки суб'єкта обмежена через необхідність тримати або носити вимірювальний пристрій. Навпаки, форма руки була б значно менш обмеженою, якби була можливість вимірювати симптоми тремору за допомогою датчика руху, який міг би

розпізнавати рух на оброблених зображеннях камери. Безмаркерне захоплення рухів і є такою технологією, за допомогою якої можна відстежувати рухи людини в реальному часі без додаткових обладнань, які б обмежували руки та впливали на оцінку.

**Основна частина** Технологія захоплення рухів працює як за допомогою маркерів, так і без них [4]. Маркери – це датчики, що розміщуються на спеціальному костюмі (зазвичай чорного кольору, зробленому з лайкри або спандексу) або в реперних точках безпосередньо на тілі людини (як правило, у її найбільш рухомих частинах, таких як суглоби). Безмаркерна технологія захоплення рухів функціонує завдяки комп'ютерному баченню, зокрема, системи розпізнання образів. Програмне забезпечення безмаркерної системи умовно розділяє тіло людини на частини й ідентифікує рух кожної з них, що надає змогу відзняти технічно складні сцени, такі як боротьба, без ризику пошкодження маркерів, а також уникнути плутанини між сигналами, які надходять від датчиків, що трапляється, скажімо, під час використання оптично пасивних систем [5].

Маркери є найбільш оптимальним рішенням для вирішення проблеми розрахунку пози і не вимагають потужних пристроїв. Його простота, надійність і ефективність мають велику перевагу. З іншого боку, безмаркерна технологія виявляє об'єкти або характерні точки сцени без попереднього знання навколишнього середовища. Цей механізм складніше реалізувати, оскільки він реалізує алгоритми, які є дорогими з точки зору часу обчислень. Поява та вдосконалення нових пристроїв дозволили використати цю технологію та відмовитися від присутності маркера в сцені та перейти до безмаркерних методів.

Нові методи та дослідження у галузі комп'ютерного зору призводять до швидкого розвитку безмаркерного підходу до захоплення руху. Ці системи не вимагають носити спеціальне обладнання для відстеження. Спеціальні алгоритми комп'ютера розроблені таким чином, щоб дозволити системі аналізувати кілька потоків оптичного входу та ідентифікувати людські форми, розбиваючи їх на складові для відстеження. Вважають, що безмаркерні системи з технічної точки зору прогресивніші, оскільки технології, що лежать в їх основі, складніші і наукомісткі. Так як комп'ютеру необхідно відрізнити праву сторону від лівого персонажа, а блискучі поверхні може збити безмаркерну систему, вони вимагають повноцінного машинного зору. Останнім часом найбільш популярними для захоплення та відстеження рук стали прилади які використовують безмаркерну технологію з додатковими датчиками, такі як Leap Motion, датчики Time-of-Flight, і датчики глибини Kinect та інші.

Leap motion (рисунок 1) - пристрій безмаркерного захоплення рухів рук, що складається з пари інфрачервоних камер та світлодіодів, що працюють в інфрачервоному діапазоні та підсвічують робочу зону [6]. Камери відстежують рух рук у робочій ділянці простору (область тривимірної взаємодії умовно маючи форму віртуального куба з стороною в 1м). Дані передаються на ПК, до якого

підключено контролер із інтерфейсом USB. Ухвалена інформація обробляється за допомогою спеціалізованого ПЗ, результат виробленої дії відображається на екрані монітора.



Рисунок 1 – Прилад Leap motion

Leap Motion розпізнає зап'ястя, долоню та пальці двох рук окремо. Ця технологія дозволяє займатися творчістю, організовувати фізику взаємодії аватарів рук з предметами у віртуальному оточенні. Контролер відстежує рухи пальців з точністю до 0,01 мм. Leap Motion Controller має перевагу у мобільності за рахунок мінімальних габаритів перед Kinect та іншими приладами. Leap Motion здатний розпізнавати рух пальців більш точно порівняно з Kinect, що може бути використане у застосуванні до діагностики для оцінки тремору рук хворих з порушеннями рівноваги.

В ідеї технології Leap Motion закладено можливість переходу на принципово новий рівень управління комп'ютером та роботи у різних програмах та іграх. Ця розробка рано чи пізно приведе людство до повної відмови спочатку від вказівних пристроїв таких як мишки і джойстиків, а згодом, можливо, і сенсорних екранів. У першому випадку ми отримуємо новий і більш природний для нас спосіб взаємодії з об'єктами та відмовляємось від зайвих периферійних пристроїв. У другому випадку, коли ми можемо відмовитися від сенсорних пристроїв, це пов'язано з тим, що Leap Motion не має значення, чи знаходяться ваші руки в рукавичках, його не лякає бруд, волога, жир і т.д.

Для того щоб відслідковувати координацію рухів потрібно відслідковувати зіткнення рук з віртуальними об'єктами, для цього потрібно використовувати один із методів виявлення перетину та відстані між 2D об'єктами.

Завдання виявлення зіткнень виникає у системах автоматизованого проектування, у робототехніці, у системах віртуальної реальності та відеоіграх (рисунок 2). У найпростішому випадку потрібно визначити, чи перетинаються задані об'єкти у просторі. Часто потрібна інформація про те, коли і де сталося

зіткнення. Наприклад, при моделюванні механічної системи може знадобитися список точок контакту, нормалі до поверхонь та глибина взаємопроникнення в точках контакту. Виявлення зіткнень є одним із найчастіше використовуваних термінів для виявлення чи розпізнавання ударів чи зіткнень об'єктів на екрані. Це велика тема, і немає єдиної техніки чи теореми включення чи візуалізації всіх типів зіткнень. Термін виявлення зіткнень широко використовується в програмуванні ігор, тоді як тестування влучення - це термінологія, яка використовується в програмуванні інтерфейсу користувача.

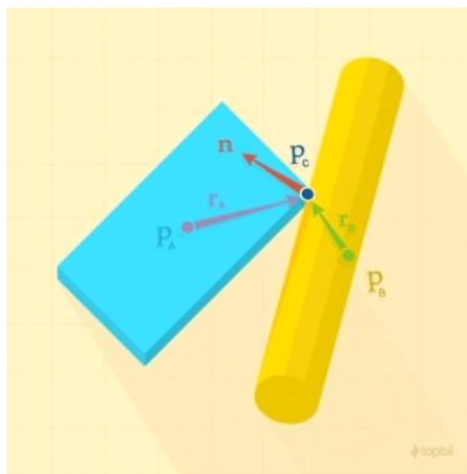


Рисунок 2 – Приклад колізії в іграх

В даний час відомо безліч алгоритмів виявлення зіткнень, більшість з яких обмежуються роботою з опуклими об'єктами, а нескладні неопуклі об'єкти зазвичай розбиваються на опуклі примітиви. Зараз найбільшого поширення набули геометричні методи виявлення зіткнень, засновані на використанні обмежувальних об'ємів [7].

Перевірка на зіткнення двох об'єктів може бути складним завданням, що включає перевірки між безліччю граней кожного об'єкта. Для швидкого виконання завдання часто використовуються спрощені форми для представлення кожного об'єкта, що дозволяє проводити швидкі тести на зіткнення. Ці тести, звичайно, не мають достатньої точності, і часто вони використовуються як швидкий тест, щоб визначити, чи потрібна подальша перевірка. Існує три таких метода обмежувальних об'ємів: коло, що обмежує, обмежуючий прямокутник, вирівняний по координатних осях і орієнтовні обмежувальні рамки.

Обмежуюче коло використовується як один із типів обмежуючого об'єму при визначенні зіткнень [8]. При застосуванні даного методу об'єкт повністю знаходиться всередині даної сфери, і зіткнення розраховуються від поверхні сфери, а не від поверхні об'єкта, що міститься в ній (рисунок 3). Використання обмежуючої сфери для виявлення зіткнень є найпростішим, швидким і грубим методом.

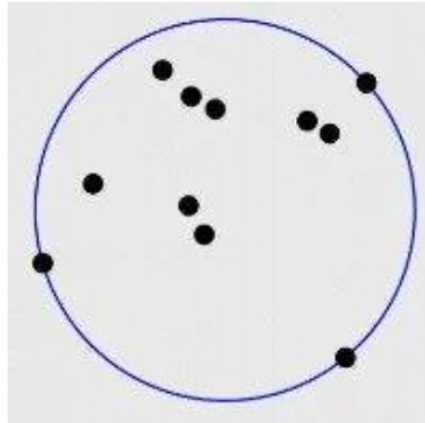


Рисунок. 3 – Обмежуюче коло

**Висновки** Використовуючи безмаркерну технологію Leap Motion, а також метод обмежуючої сфери забезпечує можливість відстеження рук людини в реальному часі, а також перевірки взаємодії з віртуальними об'єктами. Подальші дослідження спрямованні, на розробку метода для визначення коефіцієнта точності рухів людини за допомогою спіралі Архімеда, та за допомогою цих даних визначення проблем з координацією чи навіть хвороби.

#### **Перелік посилань**

1. Systematic evaluation of rating scales for impairment and disability in Parkinson's disease [Електронний ресурс]. Режим доступу: <https://pubmed.ncbi.nlm.nih.gov/12360535/>
2. Estimation of Physiological Tremor from Accelerometers for Real-Time Applications [Електронний ресурс]. Режим доступу: [https://www.researchgate.net/publication/51873260\\_Estimation\\_of\\_Physiological\\_Tremor\\_from\\_Accelerometers\\_for\\_Real-Time\\_Applications](https://www.researchgate.net/publication/51873260_Estimation_of_Physiological_Tremor_from_Accelerometers_for_Real-Time_Applications)
3. Spiral analysis: a new technique for measuring tremor with a digitizing tablet [Електронний ресурс]. Режим доступу: <https://pubmed.ncbi.nlm.nih.gov/9827601/>
4. Технології Захоплення руху [Електронний ресурс]. Режим доступу: [https://uk.wikipedia.org/wiki/Захоплення\\_руху](https://uk.wikipedia.org/wiki/Захоплення_руху)
5. Comparison of Markerless and Marker-Based Motion Capture Technologies through Simultaneous Data Collection during Gait: Proof of Concept
6. Leap Motion Controller [Електронний ресурс]. Режим доступу: [www.leapmotion.com/technology/](http://www.leapmotion.com/technology/).
7. Video Game Physics Tutorial - Part II: Collision Detection for Solid Objects [Електронний ресурс]. Режим доступу: <https://www.toptal.com/game/video-game-physics-part-ii-collision-detection-for-solid-objects>
8. Collision method bounding sphere [Електронний ресурс]. Режим доступу: [en.wikipedia.org/wiki/Bounding\\_sphere](http://en.wikipedia.org/wiki/Bounding_sphere)



```

        link.setAttribute("download", "my_data" +
accuracyPct + ".csv");
        document.body.appendChild(link); //
Required for FF

        link.click(); // This will download the
data file named "my_data.csv".

    };

    this.generateCSV = function(data){
        // utility function to output data in CSV
format for adding to spreadsheet
        var str = "";
        for(var i=0; i< data.length; i++){
            str += i + ", " + data[i].radius +
"\n";
        }

        return str;
    };
    this.xValues = [];
    this.yValues = [];

1) {
        this.generateData = function(value, i1, i2, step =
            for (let x = i1; x <= i2; x += step) {
                this.yValues.push(eval(value));
                this.xValues.push(x);
            }
        };

        this.showChart = function(xDotsValues, yDotsValues,
xyValues) {
            var fun = this.regression.slope + " * x + "
+ this.regression.intercept;
            var max = xDotsValues[xDotsValues.length -
1];

            this.generateData(fun, 0, max, 1);

            new Chart("lineChart", {
                type: "line",
                data: {
                    labels: this.xValues,
                    datasets: [{
                        label: fun,
                        fill: false,
                        pointRadius: 1,
                        borderColor:
"rgba(255,0,0,0.5)",
                        data: this.yValues
                    }
                ],
                options: {
                    legend: {display: false},
                    title: {
                        display: true,
                        text: "y = x * 2 + 7",
                        fontSize: 16
                    }
                }
            });

            new Chart("dotsChart", {
                type: "scatter",
                data: {
                    datasets: [{
                        label: 'Raddi dots',
                        pointRadius: 1,
                        pointBackgroundColor:
"rgba(0,0,255,1)",
                        data: xyValues
                    }
                ]
            });

            new Chart("resultsChart", {
                type: "line",
                data: {

```

```

                    labels: xDotsValues,
                    datasets: [{
                        label: 'Raddi dots',
                        data: yDotsValues,
                        borderColor: "blue",
                        showLine: false,
                        pointRadius: 1
                    }, {
                        label: 'Least squares
                    data: this.yValues,
                    borderColor: "red",
                    pointRadius: 1
                }
            ]
        },
        options: {
            legend: {display: false}
        }
    });
});

    };
}

Лістинг canvas:
// canvas.js
// This javascript is used to manage the cursor tracer canvas
$( document ).ready(function() {

    var spiral = new Spiral({
        // we create a Spiral object here.
        startPoint: {x: 400, y: 210},
        // See Spiral() object in spiral.js
        numberOfLoops: 3.15,
        radiusGrowthRate: 0.15
    });

    var hoverTargetsRadius = 15;

    var pathPoints = [];
        // stores the path of the mouse
    var isTracking = false;
        // flag to turn on/off tracking

    var radiusPlotForAnalysis = [];
        // create an array to store the sample #, radius
plot for analysis later

        addUserPathLayer();
        // Run a function to setup an
empty layer for userPath without any points yet.

        // look for
function addUserPathLayer() below to see the instructions

        // CREATE LAYER FOR SPIRAL GUIDELINE
$('#tracerCanvas').drawLine({
        strokeWidth: 3,
        strokeStyle: '#403f3', // gray
        visible: true,
        name: 'guideline',
        layer: true
    });

    // ADD THE SPIRAL GUIDELINE POINTS TO THE LAYER AND
DRAW IT
    $('#tracerCanvas').setLayer('guideline',
spiral.guidelinePoints)
        .drawLayers();

    // Draw the starting circle on the canvas using
drawArc method
    $('#tracerCanvas').drawArc({
        fillStyle: '#0a0', // green
        opacity: 0.75,
        x: spiral.startPoint.x,
        y: spiral.startPoint.y,
        radius: hoverTargetsRadius,
        layer: true,
        name: 'startCircle',
        mouseover: function() {
            $(this).animateLayer('startCircle', {
                fillStyle: '#0d0'
            }, 250);
            isTracking = true;
        }
    });

```

```

    },
    mouseout: function() {
        $(this).animateLayer('startCircle', {
            fillStyle: '#0a0'
        }, 250);
    }
});

// DRAW THE TARGET CIRCLE
$('#tracerCanvas').drawArc({
    fillStyle: '#00d', // green
    opacity: 0.75,
    x: spiral.endPoint.x,
    y: spiral.endPoint.y,
    radius: hoverTargetsRadius,
    layer: true,
    name: 'targetCircle',
    mouseover: function() {
        $(this).animateLayer('targetCircle', {
            fillStyle: '#55f'
        }, 250);
    }
});

if(isTracking){ // only trigger
if we already hit the green circle
    isTracking = false;
    var analysis = new
Analysis(radiusPlotForAnalysis);
    analysis.printResults();
}
},
mouseout: function() {
    $(this).animateLayer('targetCircle', {
        fillStyle: '#00d'
    }, 250);
}
});

// CREATE A PURPLE CIRCLE LAYER TO SEE THE FINGER
POSITION
$('#tracerCanvas').drawArc({
    fillStyle: '#c0f',
    radius: 10,
    layer: true,
    name: 'leapCursor',
    visible: false,
});

// DRAW INSTRUCTIONS TEXT
$('#tracerCanvas').drawText({
    fillStyle: '#403f3',
    x: 400, y: 20,
    fontSize: 14,
    fontFamily: 'Verdana',
    text: 'Place your index finger on the green circle
and spiral to the blue circle',
    layer: true,
    name: 'instructionsText'
});

// DRAW RESET BUTTON
$('#tracerCanvas').drawRect({
    fillStyle: '#f00',
    x: 60, y: 350,
    width: 50, height: 40,
    layer: true,
    name: 'resetButton',
    cornerRadius: 10,
    click: function(){
        resetPath();
    }
});

// DRAW TEXT ON RESET BUTTON
$('#tracerCanvas').drawText({
    fillStyle: '#fff',
    x:
$('#tracerCanvas').getLayer('resetButton').x,
    y:
$('#tracerCanvas').getLayer('resetButton').y,
    width: 50,
    height: 40,
    text: 'Reset',
    layer: true,
    name: 'resetText',
    intangible: true
});

});
var leapCursorLayer =
$('#tracerCanvas').getLayer('leapCursor');

// INSTRUCTIONS TRIGGERED WHEN MOUSE MOVES
$('#tracerCanvas').mousemove(function(event){

//
Create a path following the mouse
// add a point to the path array
var rect =
$('#tracerCanvas').offset();
// correct mouse coordinates
var cursor_x = event.pageX -
rect.left;
var cursor_y = event.pageY -
rect.top;

if(isTracking){
    pathPoints.push([cursor_x,
cursor_y]); // .push() adds an element to end of an
array

    var i = pathPoints.length; // use
this # to create the property name e.g. x1, x2, x3, etc
    var pathLayer =
$('#tracerCanvas').getLayer('userPath');
    pathLayer['x'+i] = pathPoints[i-
1][0];
    pathLayer['y'+i] = pathPoints[i-
1][1];

    // .push() adds an element (in
this case, an object) to the end of the array
    radiusPlotForAnalysis.push({x:
pathPoints[i-1][0] - spiral.startPoint.x,
y: pathPoints[i-1][1] -
spiral.startPoint.y});

    leapCursorLayer.x = cursor_x;
    leapCursorLayer.y = cursor_y;

    //$('#tracerCanvas').getLayer('leapCursor').visible
= true;
});

function addUserPathLayer(){
    $('#tracerCanvas').addLayer({
        name: 'userPath',
        type: 'line',
        strokeStyle: '#f00',
        strokeWidth: 3,
        index: 4
    });
}

function resetPath(){
    // clear the points array, delete userPath
and add a blank userPath layer
    pathPoints = [];
    radiusPlotForAnalysis = [];
    isTracking = false;
    $('#tracerCanvas').removeLayer('userPath');
    $('#results').html("");
    addUserPathLayer();
    $('#tracerCanvas').drawLayers();
}

});

// spiral.js
// Spiral object creation
function Spiral(params) {
    this.startPoint = params.startPoint;

```

```

this.numberOfLoops = params.numberOfLoops;

this.radiusGrowthRate = params.radiusGrowthRate;

this.xyForPolar = function(coords){
    return {x: Math.round(coords.r *
Math.cos(coords.angle)),
          y: Math.round(coords.r *
Math.sin(coords.angle))};
};

this.generateGuidelinePoints = function(){
    var allPoints = {};

    var maxAngleInDegrees = this.numberOfLoops
* 360;

    for(var currentDegrees = 0; currentDegrees
<= maxAngleInDegrees; currentDegrees++){

        var currInRadians = currentDegrees
* Math.PI / 180.0;
        var point = this.xyForPolar({r:
this.radiusGrowthRate * currentDegrees,
          angle: currInRadians});
        allPoints['x'+(currentDegrees+1)]
= point.x + this.startPoint.x;
        allPoints['y'+(currentDegrees+1)]
= point.y + this.startPoint.y;
    }
    return allPoints;
};

this.generateEndPoint = function(){
    var point = this.xyForPolar(
this.numberOfLoops * 360,
    angle: this.numberOfLoops * 2 *
Math.PI));

    return {x: point.x + this.startPoint.x,
          y: point.y +
this.startPoint.y};
};

this.endPoint = this.generateEndPoint();

this.guidelinePoints =
this.generateGuidelinePoints();
}

Лістинг spiral:
// spiral.js
// Spiral object creation
function Spiral(params) {

    this.startPoint = params.startPoint;

    this.numberOfLoops = params.numberOfLoops;

    this.radiusGrowthRate = params.radiusGrowthRate;

    this.xyForPolar = function(coords){

        return {x: Math.round(coords.r *
Math.cos(coords.angle)),
          y: Math.round(coords.r *
Math.sin(coords.angle))};
    };

    this.generateGuidelinePoints = function(){
        var allPoints = {};

        var maxAngleInDegrees = this.numberOfLoops
* 360;

        for(var currentDegrees = 0; currentDegrees
<= maxAngleInDegrees; currentDegrees++){

```

```

        var currInRadians = currentDegrees
* Math.PI / 180.0;
        var point = this.xyForPolar({r:
this.radiusGrowthRate * currentDegrees,
          angle: currInRadians});
        allPoints['x'+(currentDegrees+1)]
= point.x + this.startPoint.x;
        allPoints['y'+(currentDegrees+1)]
= point.y + this.startPoint.y;
    }
    return allPoints;
};

this.generateEndPoint = function(){
    var point = this.xyForPolar(
this.numberOfLoops * 360,
    angle: this.numberOfLoops * 2 *
Math.PI));

    return {x: point.x + this.startPoint.x,
          y: point.y +
this.startPoint.y};
};

this.endPoint = this.generateEndPoint();

this.guidelinePoints =
this.generateGuidelinePoints();
}

```

Лістинг cursor-spiral:

```

<!--
    Leap Tracer: Part 2 Tutorial
    Tutorial developed by George Marzloff
    (george@marzloffmedia.com)
-->

<!-- These are comments in HTML files and are ignored by the
browser -->

<!DOCTYPE html> <!-- tells the browser this document is html
-->
<html>

    <head>

        <title>LeapTracer - Spiral Test</title>
        <link href="style.css" rel="stylesheet"
type='text/css' />
    </head>

    <body>

        <h1 id="pageTitle">LeapTracer - Spiral
Test</h1>

        <h2 class="subhead">Beginner Tutorial for
using LeapMotion in Clinical Experiments</h2>

        <canvas id="tracerCanvas" width="800"
height="400"></canvas>

        <h3 id="results"></h3>

        <!-- The browser retrieves javascript files from
these URLs and will use this to control the HTML and CSS -->
        <script src="https://code.jquery.com/jquery-
3.2.1.min.js" integrity="sha256-
hwg4gsxgFZHoSEamD0YGBf13FyQuiTlA0gxVSNgt4="
crossorigin="anonymous"></script>
        <script
src="https://cdnjs.cloudflare.com/ajax/libs/jcanvas/16.7.3/jc
anvas.min.js" type="text/javascript"></script>
        <script type="text/javascript"
src="spiral.js"></script>
        <script type="text/javascript"
src="analysis.js"></script>
        <script type="text/javascript"
src="canvas.js"></script>

    </body>
</html>

```

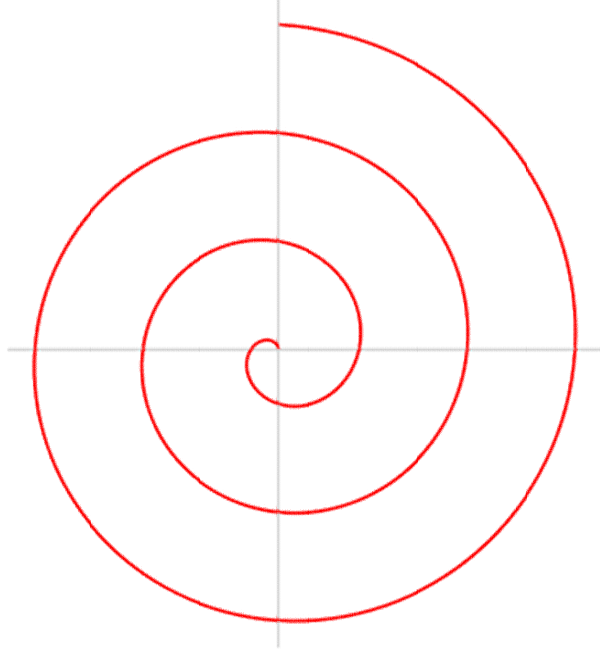
# Кваліфікаційна робота магістра

Метод визначення точності координації рухів кисті руки  
з використанням технології захоплення рухів

Виконав: студент 2 курсу групи КНм-21-1 Яницький О.О.  
Керівник: к.т.н., доцент кафедри КН Багрій Р.О.

# Мета

Метою кваліфікаційної роботи магістра є визначення точності координації рухів кисті руки з використанням технології захоплення рухів.



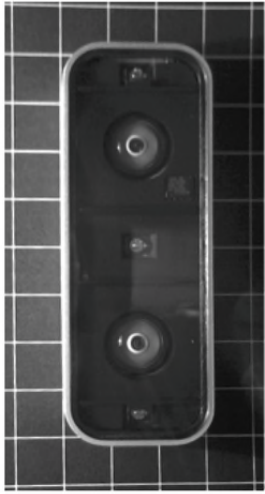
# Завдання роботи

- Провести аналіз технологій захоплення та відстеження рухів кисті руки.
  - Провести аналіз методів виявлення перетину та відстані між 2D об'єктами.
  - Удосконалити метод відстеження шляху пальця користувача та відображення його у 2D просторі.
  - Розробити метод визначення точності координації рухів кисті руки.
- Реалізувати розроблений метод для перевірки **ВИЗНАЧЕННЯ ТОЧНОСТІ** координації рухів за допомогою Leap Motion.
- Провести валідацію розробленого метода.

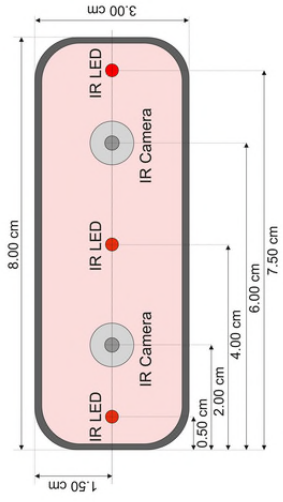
# Об'єкт та предмет дослідження

**Об'єкт дослідження** - процес відстеження рухів кисті руки в режимі реального часу та визначення точності координації цих рухів.

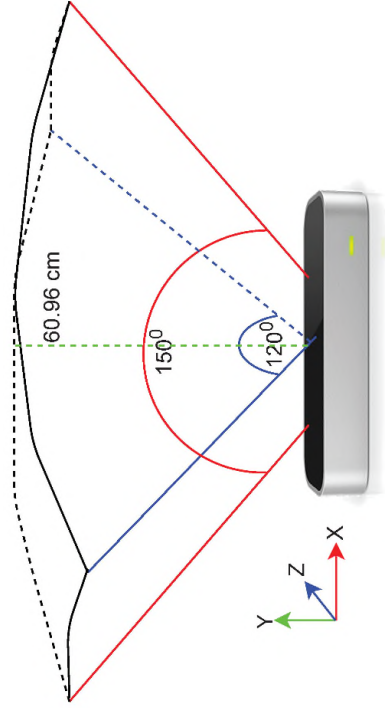
**Предмет дослідження** - модель кисті руки, обробка даних про положення руки у просторі в режимі реального часу та оцінка точності координації рухів кисті руки.

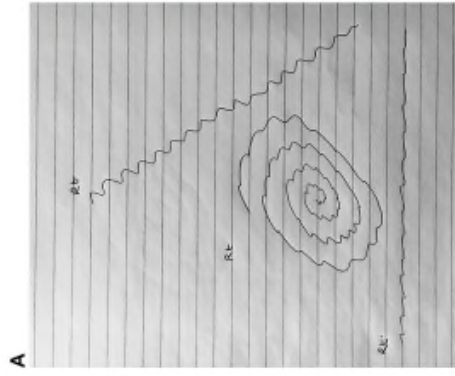
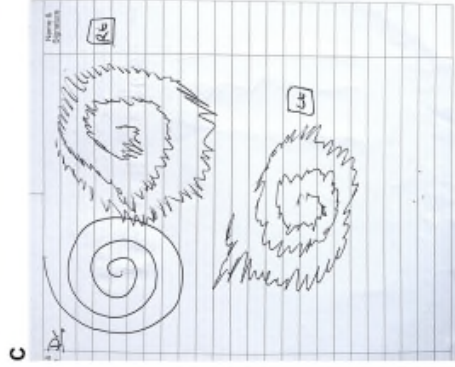
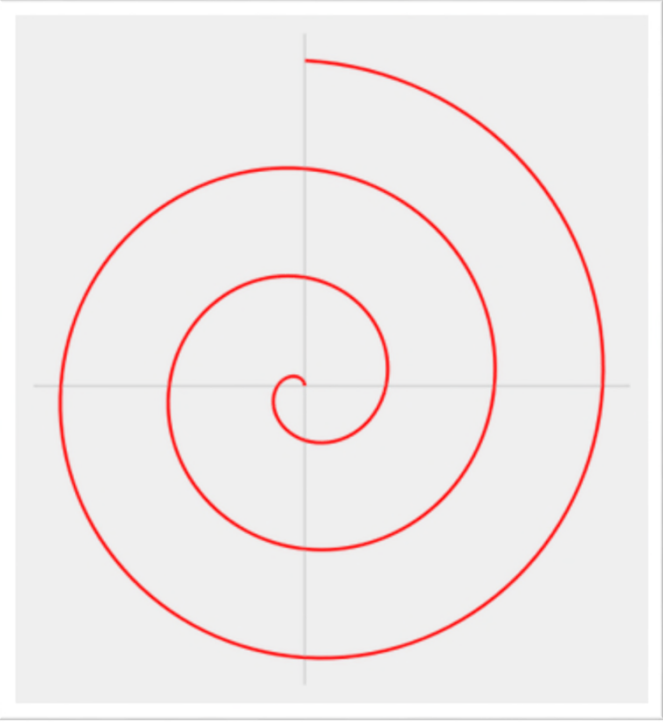


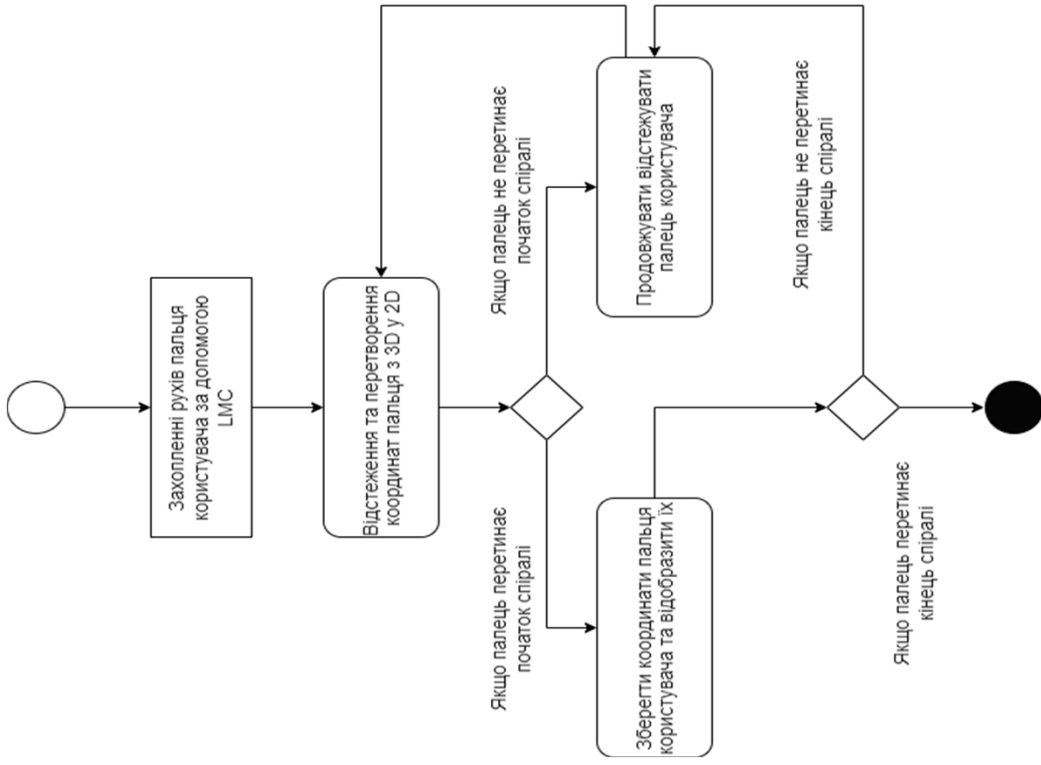
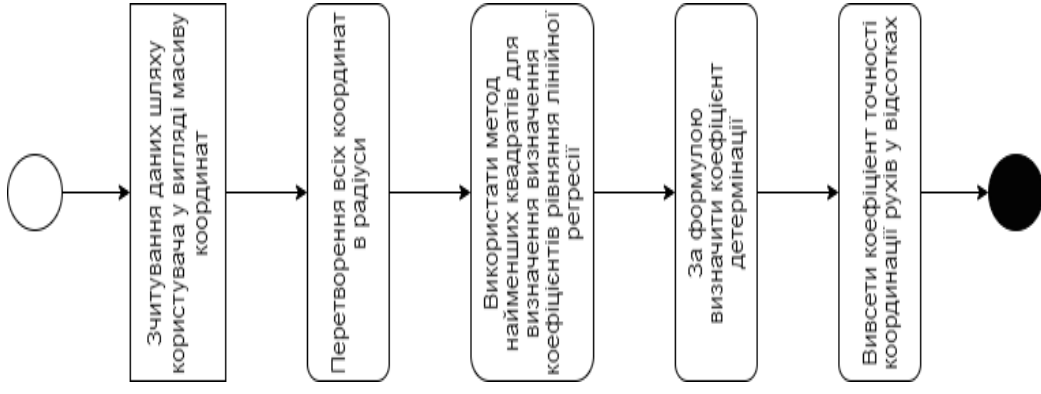
(a)



(b)

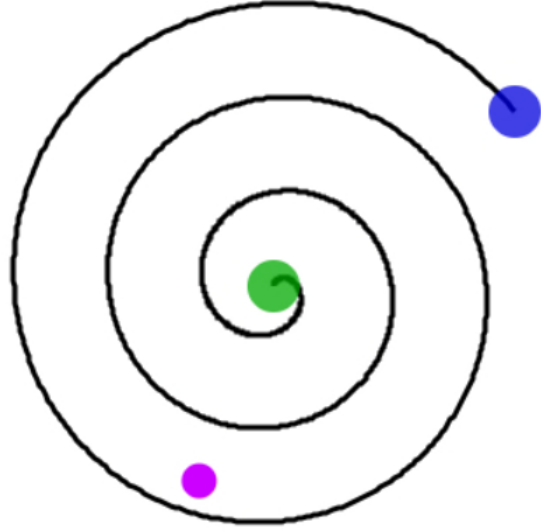






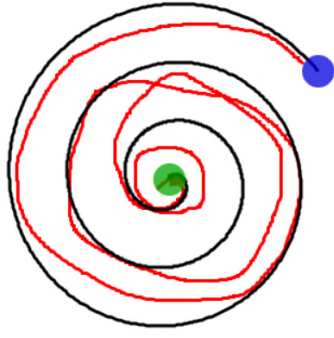


Mouseover the green circle and trace the spiral to the blue circle.



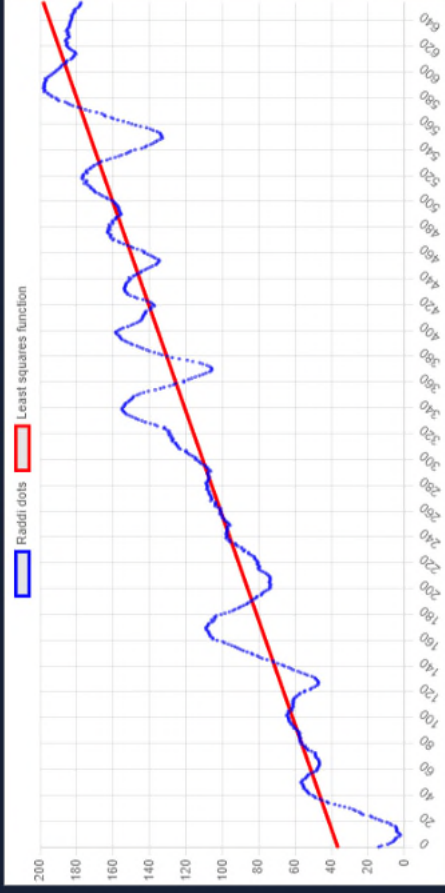
Reset

Place your index finger on the green circle and spiral to the blue circle

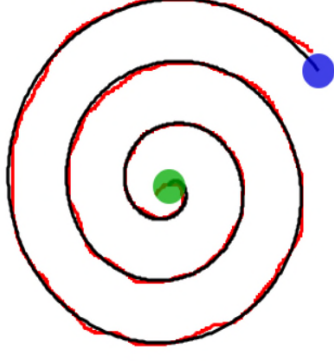


Reset

Accuracy: 68%

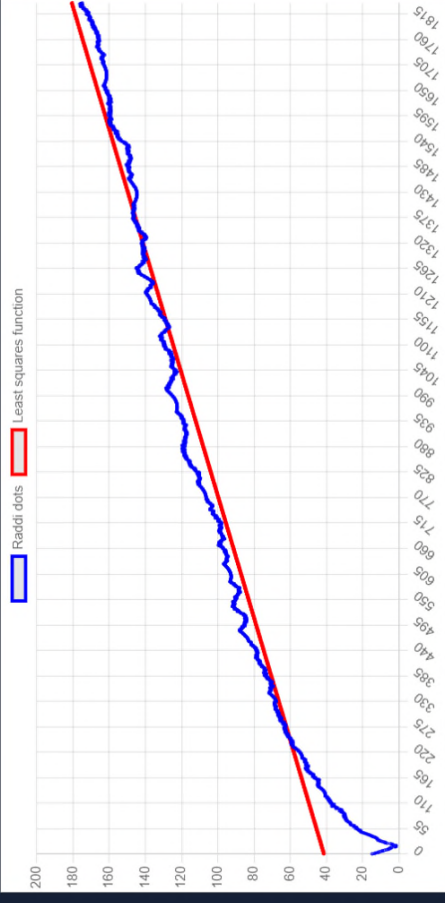


Place your index finger on the green circle and spiral to the blue circle

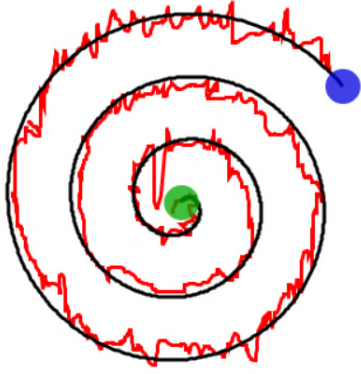


Reset

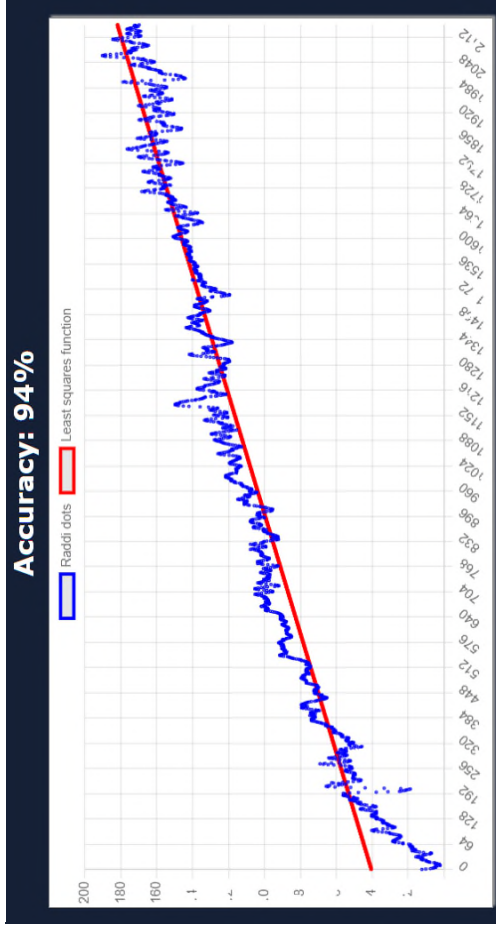
Accuracy: 96%



Place your index finger on the green circle and spiral to the blue circle



Reset



# Наукова новизна

В результаті роботи були отримані наступні положення наукової новизни:

- Вдосконалено метод для відстеження та відображення у 2D просторі переміщення пальця руки користувача, що дозволило отримувати та обробляти дані в режимі реального часу та обраховувати взаємодію з віртуальними об'єктами.
- Вперше розроблено метод для визначення точності координації рухів кисті руки, що дало можливість визначити коефіцієнт відхилення від норми, на основі якого, можна зробити висновок щодо проблем чи захворювань користувача.

# Тези на тему:

МЕТОД ДЛЯ ВИЗНАЧЕННЯ ТОЧНОСТІ КООРДИНАЦІЇ РУХІВ КИСТІ РУКИ У ПРОСТОРИ.

## *Анотація*

*Розглянуто метод визначення точності координації рухів кисті руки з використанням технології захоплення рухів, проведено аналіз технології захоплення та відстеження рухів кисті руки, а також аналіз методів виявлення перетину та відстані між 2D об'єктами.*

## *Abstract*

*Investigated method of determining the accuracy of the coordination of hand movements using motion capture technology was considered, an analysis of the technology of capturing and tracking hand movements was performed, as well as an analysis of methods for detecting the intersection and distance between 2D objects.*

Дякую за увагу!

# Anti-Plagiarism v-15.257

**Максимальне співпадіння з одним документом 1.0%**

Словники перевірки: en\_US, ru\_RU, ua\_UA. Помилки в документах: 7%

ID: 109334 Назва: КВАЛІФІКАЦІЙНА РОБОТА МАГІСТРА на тему Метод визначення точності координаті рухів кисті руки з використанням технології захоплення рухів Додано в БД: 2022-12-10 Автора: О.О. Яницький Керівники: Р.О. Багрій Консультанти: Опоненти:	Документ		Сумарний збіг по Базі Даних	
	Символи	Лексеми	Символи	Лексеми
	67960	994	2055 (3%)	34 (3%)

## Джерело плагіату

ID	Опис	Наявність плагіату в документі	
		Символи	Лексеми

Ім'я користувача:  
Кафедра КН

ID перевірки:  
1013265121

Дата перевірки:  
10.12.2022 13:53:19 EET

Тип перевірки:  
Doc vs Internet + Library

Дата звіту:  
10.12.2022 13:54:56 EET

ID користувача:  
100005671

Назва документа: КНм-21-1\_Яницький

Кількість сторінок: 74 Кількість слів: 10882 Кількість символів: 81855 Розмір файлу: 2.67 MB ID файлу: 1013023335

## 4.28% Схожість

Найбільша схожість: 2.19% з джерелом з Бібліотеки (ID файлу: 1012990292)

1.67% Джерела з Інтернету

68

Сторінка 76

3.87% Джерела з Бібліотеки

108

Сторінка 76

## 0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

## 0% Вилучень

Немає вилучених джерел

## Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи

50

РІШЕННЯ ЕКСПЕРНОЇ КОМІСІЇ

КАФЕДРИ КОМП'ЮТЕРНИХ НАУК

ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ МАГІСТРА ДО ЗАХИСТУ ЗА  
РЕЗУЛЬТАТАМИ АНАЛІЗУ ЗВІТУ ПОДІБНОСТІ

Підтверджуємо ознайомлення з результатом звіту подібності щодо роботи, генерованого системою виявлення текстових збігів/ідентичності/схожості:

Назва: Метод визначення точності координації рухів кисті руки з використанням технології захоплення рухів

Автор: Яницький Олег Олегович

Спеціальність: 122 – Комп'ютерні науки

Освітня програма: освітньо-професійна

Науковий керівник: к.т.н., доц. Багрій Р.О.

Після аналізу звіту подібності зроблено такий висновок:

№	Висновок	Позначка про відповідність
1	Запозичення, виявлені в роботі, є законними і не є плагіатом. Робота приймається до захисту.	відповідає
2	Виявлені запозичення не є плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота приймається до захисту, але має бути відкоригована. Відкоригований варіант має бути поданий на кафедру за 2 дні до захисту, разом із заявою щодо самостійності виконання письмової роботи та ідентичності друкованої та електронної версії роботи	
3	Виявлені запозичення не є плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. В зв'язку з цим мета роботи та поставлені завдання не були досягнені. Робота може бути допущена до захисту (наступного року) після того як буде відкоригована та допрацьована і успішно пройде повторну перевірку на академічний плагіат.	
4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	

Підтвердження:

Запозичення, виявлені в роботі, є законними і не є плагіатом, оскільки:




- 1) за програмою Anti-Plagiarism виявлені 1% є фрагментарними – містять поширені конструкції, загальновідомі терміни, скорочення та визначення.
- 2) За програмою UNICHECK виявлені 4.28%, що є запозиченнями, які розміщені в розділах аналізу існуючих технологій та прототипів, які не описують безпосередньо авторське дослідження і не стосуються результатів роботи.

Сумарний обсяг всіх запозичень, визначений системою виявлення збігів/ідентичності/схожості, складає 1% і 4.28% відповідно, що, з урахуванням наведених обґрунтувань, відповідає характеру наукового дослідження і свідчить на користь кваліфікаційної роботи.

Керівник роботи

Гарант ОП

Завідувач кафедри КН

Руслан Багрій

Руслан Багрій

Олександр Бармак



## ВІДГУК ОПОНЕНТА

### на кваліфікаційну роботу магістра

*гр. КНм-21-1 Яницького Олега Олеговича за темою: Метод визначення точності координації рухів кисті руки з використанням технології захоплення рухів*

#### **1. Актуальність обраної теми**

Тема визначення точності координації рухів кисті руки з використанням технології захоплення рухів є актуальною тому що допомагає виявити розладів рухів людини, а саме людський тремор. Актуальність задачі добре відображено у кваліфікаційній роботі магістра.

#### **2. Відповідність роботи предметній області спеціальності 122 Комп'ютерні науки та загальним вимогам до наукових робіт**

Робота відповідає вимогам предметної області спеціальності 122 Комп'ютерні науки, оскільки в роботі виконуються теоретичні та експериментальні дослідження в галузі комп'ютерних наук, застосовуються алгоритмічні принципи в моделюванні, проектуванні, розробці та супроводі інформаційних систем.

#### **3. Повнота розкриття мети та завдань дослідження**

Метою магістерської роботи є визначення точності координації рухів кисті руки з використанням технології захоплення рухів, яка б надала можливість швидко виконати перевірку координації рухів кисті руки та визначити відхилення від норми, на основі якого, можна зробити висновок щодо проблем чи захворювань користувача. Робота повністю розкриває поставлену мету та усі завдання, що були поставлені.

#### **4. Наявність наукової новизни**

В роботі описуються загальновідомі методи для відстеження та відображення у 2D просторі переміщення пальця руки користувача. Вдосконалення методу полягало в обрахунку взаємодії віртуальних об'єктів в режимі реального часу, що дозволило також розробити метод визначення точності координації рухів кисті руки. Окрім цього в роботі було проведено валідацію роботи методу на реальних даних. Зважаючи на це можна зробити висновок, що в роботі наявна наукова новизна в достатньому обсязі. Також слід відзначити, що проведені дослідження було опубліковано на XIV всеукраїнської науково-практичної конференції «Актуальні проблеми комп'ютерних наук АПКН-2022» 18-19 листопада 2022 р., м. Хмельницький, Україна.

#### **5. Зміст кожного розділу роботи**

В першому розділі проведено аналіз основних підходів для діагностування хвороб, що проявляються у поганій координації рухів руки. Розглянуто існуючі статі та публікації, з яких виділено плюси, мінуси та основні вимоги до пристроїв та методів для визначення точності рухів. Проведено аналіз існуючих технологій захоплення та відстеження рухів, розглянуто методи виявлення перетину та відстані між 2D об'єктами.

В другому розділі вдосконалено метод відстеження шляху пальця користувача та відображення його у 2D просторі. Також розроблено метод для визначення точності координації рухів кисті руки у просторі з використанням методу найменших квадратів.

У третьому розділі описано проектування інформаційної системи визначення точності координації рухів кисті руки у просторі. Також описано об'єктно орієнтовану модель інформаційної системи, для виділення основних сутностей предметної області та визначення їх властивостей та функцій. Також для даної моделі побудовано ER-діаграму та розроблено структури інформаційної системи.

У четвертому розділі описано архітектуру та прикладні компоненти інформаційної системи. Виконано дослідження методу визначення точності координації рухів кисті руки, який використовується в інформаційній системі. Проведено тестування інформаційної системи визначення точності координації рухів кисті руки у просторі.

## **6. Ступінь розкриття теми роботи**

В роботі недостатньо уваги приділено існуючим методам визначення точності координації рухів кисті руки у просторі. Основний акцент роботи був спрямований на методі тестування, що використовує спіраль Архімеда для діагностики.

## **7. Якість оформлення кваліфікаційної роботи**

Загалом оформлення роботи відповідає поставленим вимогам, але є певні недоліки що стосуються мови та граматики.

## **8. Недоліки кваліфікаційної роботи**

В роботі недостатньо уваги приділено тестування розробленого методу на реальних пацієнтах та виявлення нестандартних ситуацій, що можуть виникнути під час проходження процедури виявлення тремору. Також відсутнє порівняння з існуючими підходами, що не дозволяє зробити висновки щодо переваги запропонованого методу над іншими.

## **9. Загальний висновок (допускається чи не допускається до захисту), якої оцінки заслуговує кваліфікаційна робота.**

Роботі виконана студентом в достатньому обсязі, наявна наукова новизна та виконані усі поставлені завдання. Кваліфікаційна робота допускається до захисту, але враховуючи допущені помилки, заслуговує оцінку “добре”.

Опонент \_\_\_\_\_ д. ф-м. н., професор Бедратюк Л.П.



## ВІДГУК НАУКОВОГО КЕРІВНИКА

### на кваліфікаційну роботу магістра

гр. КНм-21-1 Яницького Олега Олеговича за темою: Метод визначення точності координат рухів кисті руки з використанням технології захоплення рухів

#### 1. Актуальність теми

Актуальність теми достатньо обґрунтована, оскільки визначення точності координат рухів кисті руки надає можливість виявити проблеми пов'язані з розладами моторики рук на ранніх стадіях.

#### 2. Відповідність роботи предметній області спеціальності 122 Комп'ютерні науки та загальним вимогам до наукових робіт

Теми кваліфікаційної роботи "Метод визначення точності координат рухів кисті руки з використанням технології захоплення рухів" відповідає предметній області спеціальності 122 Комп'ютерні науки та вимогам до кваліфікаційної роботи магістра, оскільки об'єктом дослідження є процес відстеження рухів кисті руки в режимі реального часу та визначення точності координат цих рухів, предметом дослідження – модель кисті руки, обробка даних про положення руки у просторі в режимі реального часу та оцінка точності координат рухів кисті руки.

#### 3. Професійні та особистісні якості магістранта

Яницький О. О. під час роботи над кваліфікаційною роботою магістра продемонстрував достатній рівень знань та умінь за спеціальністю "Комп'ютерні науки".

#### 4. Ступінь самостійності під час виконання кваліфікаційної роботи

Робота виконана самостійно, академічного плагіату не виявлено, стосовно всіх запозичень наведено відповідні посилання на джерела.

#### 5. Наукова новизна та оригінальність запропонованих підходів

Отримані такі результати: вдосконалено метод для відстеження та відображення у 2D просторі переміщення пальця руки користувача, що дозволило отримувати та обробляти дані в режимі реальному часі та обраховувати взаємодію з віртуальними об'єктами; вперше розроблено метод для визначення точності координат рухів кисті руки, що дало можливість визначити коефіцієнт відхилення від норми, на основі якого, можна зробити висновок щодо проблем чи захворювань користувача. Отримані результати оприлюднені на XIII всеукраїнської науково-практичної конференції «Актуальні проблеми

комп'ютерних наук АПКН-2022», 18-19 листопада 2022 р., м. Хмельницький, Україна, доповідь на тему «Визначення точності координації рухів кисті руки у просторі».

#### **6. Ступінь оволодіння методами дослідження**

Студент Яницький О. О. має достатньо високий ступінь володіння методами дослідження, що були використанні у роботі.

#### **7. Повнота та якість розкриття теми роботи**

Мета роботи повністю розкрита, отримані результати підтверджують достовірність наукових положень.

#### **8. Логічність, послідовність, аргументованість, літературна грамотність викладу матеріалу**

Викладення матеріалу логічне, послідовне та аргументоване. Мова і стиль викладення кваліфікаційної роботи магістра відповідають стандартам, що забезпечує доступність сприймання матеріалу і відповідає вимогам до сучасних наукових робіт.

#### **9. Можливість практичного застосування кваліфікаційної роботи, окремих її частин**

Може мати практичне значення при виявленні проблем пов'язаних з розладами моторики рук.

#### **10. Висновок про можливість допуску кваліфікаційної роботи до захисту, на яку оцінку заслуговує робота**

Вважаю, що кваліфікаційна робота студента Яницького Олега Олеговича може бути рекомендована до захисту та заслуговує на оцінку "добре".

Науковий керівник \_\_\_\_\_ к.т.н., доц. Руслан Багрій

