



Харківський національний університет радіоелектроніки

Факультет \_\_\_\_\_ комп'ютерної інженерії та управління \_\_\_\_\_

Кафедра \_\_\_\_\_ електронних обчислювальних машин \_\_\_\_\_

Рівень вищої освіти \_\_\_\_\_ перший (бакалаврський) \_\_\_\_\_

Спеціальність \_\_\_\_\_ 123 «Комп'ютерна інженерія» \_\_\_\_\_  
(код і повна назва)

Тип програми \_\_\_\_\_ освітньо-професійна \_\_\_\_\_  
(освітньо-професійна або освітньо-наукова)

Освітня програма \_\_\_\_\_ Комп'ютерна інженерія \_\_\_\_\_  
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_  
(підпис)

“ \_\_\_\_\_ ” \_\_\_\_\_ 20\_\_ р.

## ЗАВДАННЯ

### НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві \_\_\_\_\_ Слуту Сергію Валерійовичу \_\_\_\_\_  
(прізвище, ім'я, по батькові)

1. Тема роботи \_\_\_\_\_ Тіньове відтворення рухів кінцівок людини \_\_\_\_\_

затверджена наказом по університету від “ 26 ” травня 2025 р. № 424 Ст

2. Термін подання здобувачем роботи до екзаменаційної комісії \_\_\_\_\_ 17 червня 2025 \_\_\_\_\_

3. Вхідні дані до роботи \_\_\_\_\_

документація Arduino Uno;

документація C/C++;

документація Python;

інтегроване середовище Arduino

документація Tensorflow

документація MPU6050

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

аналіз предметної області;

аналіз використаних технологій;

програмна реалізація;

схема та макет проекту;

інструкція користувача;

висновки.

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій 16 слайдів

---

---

---

---

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


Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата

### КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Строк / терміни виконання етапів роботи	Примітка
	Аналіз проблеми та огляд існуючих рішень	27.05.25-28.05.25	
	Вибір технології розробки апаратного та програмного забезпечення	29.05.25-30.05.25	
	Розробка алгоритмів	31.05.25-5.06.25	
	Розробка програмних модулів	6.06.25-9.06.25	
	Відлагодження програмних модулів	10.06.25-12.06.25	
	Оформлення матеріалів кваліфікаційної роботи	13.06.25-14.06.25	
	Подання кваліфікаційної роботи керівникові	14.06.25-15.06.25	
	Подання кваліфікаційної роботи на рецензування	15.06.25-16.06.25	

Дата видачі завдання “ 26 ” травня 2025 р.

Здобувач

  
(підпис)

Керівник роботи

(підпис)

ас. Ігор МИХАЙЛЧЕНКО

(посада, власне ім'я, прізвище)

## РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 79 с., 11 рис., 4 табл., 2 дод., 85 джерел.

КІБЕРФІЗИЧНА СИСТЕМА, ARDUINO, MPU6050, SPECTRAFLEX, НЕЙРОННА МЕРЕЖА, ВІДСТЕЖЕННЯ В РЕАЛЬНОМУ ЧАСІ, РЕАБІЛІТАЦІЯ, ВИЯВЛЕННЯ РУХУ.

Метою кваліфікаційної роботи є розробка кіберфізичної системи для зчитування, розпізнавання та відтворення рухів і положень верхніх кінцівок людини. Основними характеристиками розробленої системи є спеціалізовані нейронні мережі, датчики spectraflex для виявлення руху та м'язової активності, платформа мікроконтролера на базі Arduino та інерційний вимірювальний блок MPU6050 для відстеження руху в режимі реального часу. Забезпечуючи точний зворотний зв'язок та гнучкість до індивідуальних потреб, система призначена для сприяння індивідуальній реабілітації.

Під час кваліфікаційної роботи було досліджено існуючі рішення для визначення їхніх переваг та недоліків; потім було визначено необхідну функціональність системи. Після цього було створено повний прототип з використанням апаратних компонентів та спеціального програмного забезпечення для обробки даних руху, розпізнавання образів та генерації вихідних даних.

Дослідження призвело до створення робочої системи, яку можна використовувати в медичних, біологічних та спортивних умовах, зокрема для реабілітації та допоміжних технологій для людей з руховими та сенсорними обмеженнями. Розробку нейронних мереж та обробку даних здійснено за допомогою Python. Система була впроваджена з використанням Arduino Uno, інерційних датчиків MPU6050 та сенсорів SpectraFlex.

## ABSTRACT

Bachelor's thesis: 79 pages, 11 figures, 4 tables, 2 appendices, 85 sources.

CYBER-PHYSICAL SYSTEM, ARDUINO, MPU6050, SPECTRAFLEX, NEURAL NETWORK, REAL-TIME TRACKING, REHABILITATION, MOTION DETECTION.

The major goal of this thesis is to develop a cyber-physical system for reading, recognizing, and reproducing the movements and positions of the human upper limbs. The main features of the developed system are custom neural networks, spectraflex sensors for motion and muscle activity detection, an Arduino-based microcontroller platform, and the MPU6050 inertial measurement unit for real-time motion tracking. By providing exact feedback and flexibility to individual needs, the system is meant to assist tailored rehabilitation.

Existing solutions were examined during the qualification work to determine their benefits and drawbacks; the necessary system functionality was then specified. A total prototype was then created using hardware parts and bespoke software for motion data processing, pattern recognition, and output generation. The study produced a working system that can be used in medical, biological, and sports-related settings, particularly for rehabilitation and assistive technologies for people with motor and sensory limitations. Programming in C/C++ for embedded control and Python for neural network development and data processing guided the implementation, which included Arduino Uno, MPU6050 sensors, and spectraflex sensors.

## ЗМІСТ

СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ .....	8
ВСТУП .....	9
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ .....	11
1.1 Програмне забезпечення, машинне навчання та штучний інтелект .....	11
1.1.1 Методи машинного навчання: класифікація та застосування в ШІ.....	11
1.1.2 Інтерпретованість і прозорість ШІ-систем у медицині (XAI, LIME, SHAP) .....	14
1.2 Технології реабілітації та пристрої .....	16
1.2.1 Сучасні реабілітаційні технології: екзоскелети, сенсори та ШІ .....	16
1.2.2 Технології реабілітації та комунікації для осіб з порушеннями слуху .....	18
1.3 Постановка задачі.....	21
2 АНАЛІЗ ВИКОРИСТОВУВАНИХ ТЕХНОЛОГІЙ .....	23
2.1 Вибір та обґрунтування апаратних компонентів .....	23
2.2 Інструментарій розробки.....	27
2.2.1 Visual Studio Code .....	27
2.2.2 Arduino IDE .....	28
2.2.3 Python.....	29
2.2.4 TensorFlow та Scikit-learn .....	30
2.2.5 NumPy та Pandas.....	30
2.2.6 Matplotlib .....	31
3 АРХІТЕКТУРА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ.....	32
3.1 Розробка загальної архітектури апаратного забезпечення .....	32
3.2 Формат даних з датчиків згину та положення .....	34
3.3 Попередня обробка та фільтрація сигналів .....	36
4 АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	39

4.1 Технології.....	39
4.2 Робочий процес .....	39
4.2.1 Збір, нормалізація та підготовка даних .....	39
4.2.2 Модельне навчання.....	40
4.2.3 Впровадження прогнозування в режимі реального часу .....	41
5 ТЕСТУВАННЯ АПАРАТНИХ КОМПОНЕНТІВ .....	42
5.1 Тестування датчиків Spectra Flex .....	42
5.2 Тестування MPU6050.....	42
5.3 Тестування швидкості системи.....	43
5.4 Методологія тестування апаратного забезпечення .....	44
5.5 Аналіз результатів тестування .....	47
ВИСНОВКИ.....	51
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ .....	53
ДОДАТОК А ГРАФІЧНИЙ МАТЕРІАЛ КВАЛІФІКАЦІЙНОЇ РОБОТИ .....	63
ДОДАТОК Б ПРОГРАМНА РЕАЛІЗАЦІЯ СИСТЕМИ .....	71
Б.1 Отримання даних з сенсорів.....	71
Б.1.1 Arduino-скетч для зчитування даних .....	71
Б.1.2 Python-скрипт для збору та збереження даних.....	72
Б.1.3 Виправлення руfirmata для сумісності .....	74
Б.2 Навчання моделі розпізнавання жестів .....	75
Б.2.1 Побудова та тренування моделі на TensorFlow.....	75
Б.2.2 Розпізнавання жестів у реальному часі .....	77

## СКОРОЧЕННЯ ТА УМОВНІ ПОЗНАКИ

AI – штучний інтелект

APM – автоматизоване робоче місце

ІАЦ – інформаційно-аналітичний центр

ІМБ – інерційний вимірювальний блок

МН – машинне навчання

МП – мікропроцесор

МПУ-6050 / MPU6050 – модуль з акселерометром та гіроскопом

ПЗ – програмне забезпечення

СФ / SpectraFlex – датчик згину пальців (торговельна назва)

ШІ – штучний інтелект

CSV (Comma-Separated Values) – формат табличних даних

IDE (Integrated Development Environment) – інтегроване середовище розробки

I2C (Inter-Integrated Circuit) – послідовний протокол зв'язку

ML (Machine Learning) – машинне навчання

MPU6050 – Motion Processing Unit, інерційний сенсор

ReLU (Rectified Linear Unit) – функція активації

SHAP (Shapley Additive Explanations) – метод пояснення моделей

XAI (Explainable Artificial Intelligence) – пояснюваний штучний інтелект

## ВСТУП

Технологічні інновації та цифрова трансформація створюють можливості в охороні здоров'я завдяки культурі, орієнтованій на пацієнта, покращуючи самопочуття пацієнтів та знижуючи вартість послуг. Штучний інтелект (ШІ) є галуззю досліджень, зосередженою на розумінні природи складних моделей машинного навчання (МН), зокрема, в галузі охорони здоров'я [1-3]. Медицина є широкою галуззю для застосування нових програмних рішень, оскільки вона включає діагностику, профілактику, лікування, реабілітацію.

Використання високотехнологічного обладнання та отримання великих обсягів даних, як результатів, ускладнюють традиційну оцінку інформації при вивченні знімків рентгенівських, магнітно-резонансної томографії, комп'ютерної томографії, ультразвукових зображень тощо. Діагностика може не обмежуватися лише аналізом даних зображень, а вимагає інтерпретації текстових даних. Лікування та профілактика стають трудомісткими для лікарів, але нові можливості відкриваються за участю надійних та прозорих моделей ШІ. Впровадження ШІ та сенсорних технологій в проекти з реабілітації рухів людини сприяє прогресу в отриманні точного зворотного зв'язку при дослідженні розробок для персоналізованих рішень, оптимізації результатів реабілітації з покращенням рухливості, функціональної незалежності осіб, підвищення якості життя осіб у різних умовах [4-7].

В Україні інтеграція реабілітаційних заходів та створення програмного супроводу з застосуванням МН та ШІ у спільні проекти набуває особливої актуальності у теперішній час.

Мета роботи – розробка кіберфізичної системи для зчитування, визначення та подальшого відтворення рухів і положення верхніх кінцівок людини, з подальшим застосуванням у медичних, реабілітаційних і комунікаційних рішеннях.

Для досягнення цієї мети в роботі вирішуються питання, пов'язані з розробкою апаратної та програмної частин системи, збором і обробкою даних з сенсорів, розпізнаванням рухів за допомогою методів машинного навчання, а також з тестуванням точності та функціональності створеної системи.

Теоретичне значення роботи полягає в створенні proof of concept системи передачі інформації для проектування та розвитку людиноцентричної моделі в реабілітації та адаптації осіб з особливими потребами з можливістю подальшої імплементації у практичну діяльність в галузях медицини, біології, спорту.

Система може бути використана у реабілітаційних, інтеграційних проектах для осіб з особливостями/порушеннями опорно-рухової та сенсорної систем з урахуванням індивідуальних потреб.

## 1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

### 1.1 Програмне забезпечення, машинне навчання та штучний інтелект

#### 1.1.1 Методи машинного навчання: класифікація та застосування в ШІ

Наразі існують три основні категорії методології машинного навчання: навчання з учителем, навчання без учителя та навчання з підкріпленням. Залежно від типу вхідних даних, тобто чи є вони маркованими, немаркованими або комбінацією обох, більш традиційна класифікація, що базується на моделі, побудованій за допомогою цього методу, поділяється на моделі з учителем, без учителя та напів-навчальні моделі [1, 8-12]. Кожна точка даних має мітку (правильна/очікувана відповідь), що вимагає розробки моделі машинного навчання в моделях з учителем, які складають основну частину опублікованих методів машинного навчання. Навчальні дані та тестові дані – це дві підмножини, на які зазвичай поділяються дані. Модель налаштовується та навчається за допомогою навчальних даних. Загальність моделі оцінюється за допомогою тестових даних. Проблеми регресії та класифікації також можна вирішити за допомогою навчання з учителем. Мітки в задачах класифікації – це дискретні значення, тоді як мітки в задачах регресії – це неперервні значення. Зрештою, метрики, що використовуються для оцінки якості моделі, залежать від застосування та типу проблеми [13-14, 8].

Вимірювання вихідної мітки зазвичай недоступне в моделях без учителя. У цьому випадку алгоритм використовує структури даних для вивчення зв'язків, створюючи приховані шаблони, корисність яких необхідно оцінити. Одним з основних застосувань машинного навчання без учителя є виявлення підгруп зі спільними характеристиками, що часто досягається шляхом кластеризації та зменшення розмірності [16, 17].

Агент, який навчається приймати рішення, взаємодіючи з оточенням, називається таким, що бере участь у навчанні з підкріпленням. Агент навчається оптимізувати кумулятивну винагороду з часом після отримання вхідних даних у формі стимулів або штрафів за свою поведінку. Навчання з підкріпленням не вимагає навчального набору або позначених даних, на відміну від навчання з учителем та навчання без учителя. Проблеми, що стосуються складної динаміки, на яку впливають змінні стимули та обставини, як-от у клінічних умовах, можна вирішити за допомогою навчання з підкріпленням. Навчання з підкріпленням може допомогти лікарям у виборі найкращих планів лікування пацієнта та може коригувати свої прогнози, наприклад, коли спостерігається побічна реакція від взаємодії агента з середовищем [13, 15, 18-21].

У напівконтрольованій стратегії існує комбінація маркованих та немаркованих даних. Тема напівконтрольованого навчання значно просунулася завдяки кільком методам, запропонованим дослідниками для ефективного використання поєднання маркованих та немаркованих даних. Ці стратегії спрямовані на вирішення проблем, спричинених зростанням кількості немаркованих даних та обмеженою кількістю маркованих даних [8, 22, 23].

Інтерпретація результатів - це вивчення зв'язків та факторів, що впливають на прогнози, тоді як прогнозування – це оцінка або прогноз невідомих подій. Обидва елементи є важливими для машинного навчання, оскільки висновок допомагає навчатися на навчених моделях та отримувати розуміння основних механізмів, тоді як прогнозування дозволяє створювати корисні прогнози. Компроміс між інтерпретованістю та точністю моделі є ключовою темою в машинному навчанні. Деякі методи простіші для розуміння, але більш негнучкі та менш точні, тоді як інші є більш адаптивними у визначенні форми функції, але їх може бути складно інтерпретувати [8, 22, 24].

Вхідні дані машинного навчання можуть надходити з різних джерел,

включаючи часові ряди, оміку, клінічні дані та медичну візуалізацію. Унімодальне машинне навчання використовує один тип вхідних даних, тоді як мультимодальне машинне навчання використовує кілька типів вхідних даних. Раннє об'єднання створює єдину модель машинного навчання шляхом об'єднання типів вхідних даних, проміжне об'єднання включає взаємопов'язані моделі машинного навчання, а пізнє об'єднання створює окремі унімодальні моделі, які об'єднуються в кінцеву модель. Мультимодальне машинне навчання дає більш ретельні та точні прогнози порівняно з унімодальними моделями. Процедури проміжного та пізнього об'єднання перевершують інші мультимодальні підходи, оскільки вони враховують додаткову інформацію під час навчання моделі.

Стратегії пізнього об'єднання машинного навчання можуть покращити діагностичні критерії, класифікацію пухлин та ідентифікацію підтипів, як показано в таких дослідженнях, як НМРЛ, де об'єднання п'яти різних джерел даних досягло кращої продуктивності класифікації, ніж алгоритми з одним джерелом. Ці стратегії також можуть підтримувати тераностику раку, поєднуючи точну молекулярну візуалізацію, ранню діагностику та адаптовану променеви терапію відповідно до унікального профілю оміки кожного пацієнта [25, 26]. Використання машинного навчання у вивченні мультифакторіальних захворювань дозволило виявити приховані сигнатури та досі неідентифіковані змінні, які впливають на початок, перебіг та рецидив захворювання. Крім того, машинне навчання проливає світло на невдалі клінічні випробування, допомагаючи зрозуміти їхні недоліки та можливі переваги, одночасно запобігаючи, наприклад, токсичності ліків та іншим наслідкам, які можуть вплинути на якість життя пацієнтів та ефективність лікування. Однією з обчислювальних стратегій, досліджених за допомогою технології машинного навчання, є скринінг для прогнозування синергетичних комбінацій ліків.

Наприклад, 11 валідованих, неперевіраних комбінацій ліків з різними результатами були знайдені шляхом оцінки повторного введення понад 40

різних методів лікування клітин меланоми [27-30].

### 1.1.2 Інтерпретованість і прозорість ШІ-систем у медицині (XAI, LIME, SHAP)

У медицині інтерпретованість та прозорість систем машинного навчання та штучного інтелекту мають критичне значення. Для запобігання наслідкам, пов'язаним з неправильними діагнозами та методами лікування, алгоритми пояснювального ШІ (explainable AI, XAI), локальних інтерпретованих модельно-агностичних пояснень (local interpretable model-agnostic explanations, LIME) та адитивних пояснень Шеплі (Shapley additive explanations, SHAP) можуть забезпечити рішення, підвищуючи довіру до прогнозів та підвищуючи впевненість у системах. XAI – це галузь, яка надає пояснення результатам, отриманим за допомогою моделей штучного інтелекту, або способу, яким модель досягла цього результату чи рішення. Метою XAI є створення прозорих та надійних систем штучного інтелекту, які можна інтегрувати в процес прийняття рішень на додатковий спосіб [1,3]. Автори статті [1] представляють результати аналізу близько ста статей, присвячених використанню моделей машинного навчання разом з алгоритмами XAI для забезпечення інтерпретованості та прозорості в медицині. LIME був найбільш обговорюваним та найчастіше використовуваним у більшості досліджень. Після LIME йшли SHAP, CAM та GradCAM.

Оцінка медичних наборів даних, які найчастіше використовувалися для таких досліджень, показала, що, наприклад, рентгенівські знімки хворих на COVID-19 були найчастіше аналізованими наборами даних у різних країнах світу. Крім того, лікарні частіше діляться даними приватно з дослідниками, і основною проблемою є брак даних медичних зображень. У таких випадках одним із варіантів вирішення проблеми може бути використання попередньо навчених моделей та генерація синтетичних зображень [1,31]. На думку

дослідників [32], питання, пов'язані з розмежуванням між прозорістю та інтерпретаційністю, потребують пильної уваги, особливо щодо того, як складні моделі ШІ/МН формують свої прогнози або рекомендації. Використання моделей ШІ/МН, в яких обчислення, що лежать в основі вихідних даних, зокрема, в діагностиці інфекційних захворювань, профілюванні резистентності етіологічних агентів, терапевтичних рекомендаціях та розробці захисних антигенів для вакцин, має бути зрозумілим як для фахівців з обробки даних, так і для клініцистів [33–35]. Пояснювальний I&M ( Explanatory I&M) викликає все більший інтерес. Здатність пояснити рішення моделі чорної скриньки часто досягається шляхом створення додаткової моделі, яка є менш точною, але краще інтерпретованою. Це називається пояснювальним. Локальні пояснення, які інтерпретуються, не лежать в основі LIME та SHAP. SHAP надає значення знакам на межах конкретного прогнозу. У той час як LIME створює локалізовані пояснення, які зосереджені на окремих прогнозах, використовуючи додаткові інтерпретаційні моделі [1, 32, 36, 37].

Для розуміння проблем та потенціалу, необхідно розрізнити інтерпретацію та пояснення. Модель можна інтерпретувати, якщо розрахувати скін, щоб привести до кінцевого прогнозу, а додаткові вхідні дані різних ознак можна легко вивести з моделі «білого екрану». У той час, коли модель описується як «пояснювальна», необхідно пам'ятати, що мета-пояснення максимально наближене до інтерпретації. Таким чином, правильність пояснення може бути дуже різною, і це також може бути важливим або суттєвим. Зв'язок з цими причинами вже давно займають різні точки зору; деякі з них представили різні аргументи щодо переваг інтерпретаційних моделей, а не пояснювальних [38, 39].

Аргументи щодо корисності пояснення ґрунтуються на досягненнях дослідження пояснювального ШІ/МН, дотриманні типу та чіткості вхідних даних, оцінці ефективності та безпеки обробки/інструментів ШІ/МН за допомогою рандомізованого контролю та спеціальних досліджень. Можна

використовувати рандомізовані або незмінні пояснювальні моделі ШІ/МН для оцінки їх ефективності та безпеки в клінічній практиці з високим рівнем доказів. Таким чином, існує правова та етична основа, яка регулює використання обґрунтованих розвідувальних даних у медицині, щоб гарантувати постійне оновлення та дотримання сучасних розробок у цій галузі, а також чинних правил і норм конфіденційності [32, 40, 41].

## 1.2. Технології реабілітації та пристрої

### 1.2.1 Сучасні реабілітаційні технології: екзоскелети, сенсори та ШІ

Дослідження та розвиток передових технологій у напрямі покращення рухів людини надає цінну платформу для визначення ключових механізмів реабілітації рухів людини, інтегруючи фізіологічні стратегії з розробкою допоміжних пристроїв. Важлива роль належить передовим алгоритмам, віртуальній реальності, датчикам, що носяться, і МН у вирішенні ключових проблем реабілітації. Дослідження охоплюють широкий спектр тем: порушення рухової активності, спричинені інсультом, запальними процесами, нейродегенеративні стани, порушення внаслідок травм, порушення рухливості в дітей, спортивні тренування, сенсорні порушення/особливості, що потребують додаткової активності рухів. Демонструючи новаторські технології із застосуванням датчиків та ШІ, дослідження потребують нейрофізіологічних оцінок та моделей МН для оптимізації результатів реабілітації, покращенням якості життя осіб та досягнення мети [42, 43].

Зокрема, до роботизованих екзоскелетів відносять механізми, що носяться та здатні доповнювати, відновлювати або допомагати функціонуванню людських кінцівок, діючи паралельно. Ця технологія застосовується для осіб в багатьох областях активності, від промислових або військових, де користувач має виконувати важке завдання, до космічних

телеуправління або охорони здоров'я. Напрями, в яких роботизовані екзоскелети кінцівок показали багатообіцяючі результати, це реабілітація або допомога рухів, зокрема, ходи людини [44].

Роботизовані екзоскелети, які носять, вже набули визнання за їх потенціал для допомоги в реабілітації при різних станах неврологічного генезу. Представлено результати порівняльного аналізу роботизованої терапії (РТ) з традиційною терапією для покращення рухової функції верхніх кінцівок у пацієнтів після інсульту. Згідно результатів, що РТ значно прискорила покращення показників оцінки верхніх кінцівок за шкалою Фугля-Майєра, особливо у пацієнтів на стадії Бруннстрема III, хоча обидві групи показали схожі покращення функціональної незалежності, виміряні за допомогою іншого показника, модифікованого індексу Бартела [45].

Запропоновано неінвазивний метод виявлення остеоартриту колінного суглоба та моніторингу прогресу реабілітації шляхом інтеграції даних про ходу та м'язову активність з моделями. Носні датчики та змінні, не пов'язані з колінним суглобом, можуть бути використані для раннього виявлення остеоартриту колінного суглоба та персоналізованого втручання, демонструючи потенціал III та сенсорних технологій за рахунок точного зворотного зв'язку [46].

Реабілітаційна система на основі VR, яка реконструює пози всього тіла із розріджених сигналів руху, забезпечує корекцію руху в реальному часі та дає можливість хворим проходити ефективну терапію без частих візитів до лікаря, демонструючи потенціал для покращення відновлення моторики, залученості та доступності [47].

Описана структура для роботизованої реабілітації верхніх кінцівок, використовуючи обмеження робочого простору, специфічні для суб'єкта, та стратегії управління на основі продуктивності для підвищення ефективності навчання, природного руху та залучення пацієнта [48].

Потужний потенціал екзоскелетів та адаптованих систем управління для розвитку реабілітаційних проєктів застосовується не тільки для

покращення активності різних груп хворих.

Для вдосконалення розпізнавання плавальних рухів та отримання зворотного зв'язку в реальному часі з винятковою точністю від плавців створена система Swimtrans Net, яка використовує моделі Swin-Transformer та CLIP [49].

С одного боку, роботизовані пристрої, що носять (Wearable robotic devices, WRD) для допомоги, підтримки або доповнення людських функцій та дій, стають дедалі більш популярними та практичними. Розроблено велику різноманітність від м'яких, що повністю носилися, до стаціонарних і жорстких пристроїв, що демонструють перспективні діапазони функціональності. В той же час, кількість WRD у щоденному використанні залишається низкою через обмеженість та недоступність на ринку, через обмеження прийняття технологій [42, 51-55]. Крім того, зручність у використанні, яка визначається як «ступінь, в якій система, продукт або послуга можуть використовуватися певними користувачами для досягнення певних цілей з ефективністю, результативністю та задоволенням у певному контексті використання», та необхідність урахування персоналізованих потреб зацікавлених осіб вважаються ключовими факторами обмеження [42, 56-60].

Рішенням для мінімізації обмежень зручності використання є проектування, орієнтоване на користувача, яке спрямоване на залучення зацікавлених сторін у технологію протягом усієї розробки пристрою для ефективного вирішення та вирішення потреб, зокрема, для допомоги, підтримки або розширення людських можливостей. [42, 61-63].

### 1.2.2 Технології реабілітації та комунікації для осіб з порушеннями слуху

Технологічні інновації та цифрова трансформація створюють можливості в охороні здоров'я завдяки культурі, орієнтованій на пацієнта,

покращуючи самопочуття пацієнтів та знижуючи вартість послуг. Розробка інноваційних, недорогих технологій з ефективними моделями обслуговування для передачі інформації може бути одним із можливих рішень для реабілітації та адаптації людей з потребами трансформації мови жестів в синтезований аудіоконтент. Оскільки форма мовлення, голосу та слуху є єдиною єдиною функціональною системою в онтогенезі та соціокультурному розвитку людини, тому особами, які потребують комунікаційної допомоги, здебільшого вважаються люди з втратою слуху різної етіології [64-66].

В останні роки втрата слуху призвела до глобальних економічних витрат, що перевищують 981 мільярд доларів США. Аналіз сценаріїв показує, що навіть 5% зниження поширеності втрати слуху може заощадити 49 мільярдів доларів США у всьому світі [67].

Традиційні рішення для підвищення інклюзивності, адаптації, аудіореабілітації людей, які цього потребують, включають навчання жестовій мові, реабілітаційну терапію; слухові апарати та імплантати; а також інші сенсорні замінники, такі як читання мовлення; використання допоміжних технологій, таких як персональні FM-системи та послуги, такі як субтитри [68, 69]; розробка та впровадження стандартів аудіологічної практики, таких як рекомендації Організації зі стандартів аудіологічної практики (APSO) [70]. Тим не менш, експерти відзначають техноцентричний характер цих підходів, водночас зростаючий попит на «орієнтовану на пацієнта допомогу», яка б ефективніше підтримувала аудіологічну реабілітацію [71].

Незважаючи на це, володіння слуховими апаратами та іншими допоміжними пристроями залишається проблемою для багатьох через високу вартість, відсутність страхового покриття, соціальну стигму, марнославство та труднощі в доступі до медичної допомоги у сфері слуху [67]. Наразі 83% світової потреби в слухових апаратах не задоволені, що розраховується як частка людей з помірною або тяжкою втратою слуху, які не користуються слуховим апаратом [72]. У 2019 році втрата слуху торкнулася 20,3%

населення світу, приблизно 1,57 мільярда людей, включаючи близько 70 мільйонів дітей віком від 0 до 15 років. Прогнози показують, що до 2050 року цей показник зросте на 56,1%, досягнувши 2,45 мільярда людей [68].

Зростання поширеності втрати слуху зумовлене глобальними демографічними тенденціями та стійкими, зростаючими факторами ризику. Екологічні, поведінкові та біологічні фактори відіграють певну роль у траєкторіях втрати слуху. Такі випадки часто потребують довічної підтримки та реабілітації [73]. У європейському регіоні поширеність часткової або повної втрати слуху становила 197 мільйонів людей у 2019 році, а прогнози передбачають приблизно 236 мільйонів людей до 2050 року [68]. Ці люди могли б скористатися доступними технологічними рішеннями та послугами.

Важливо зазначити, що більшість існуючих технологій зосереджені на допомозі людям в отриманні інформації. Однак передача інформації, яка може вплинути на соціальне благополуччя, є не менш важливою [68]. Мова жестів є основною формою комунікації для тих, хто не може передавати інформацію за допомогою звуку. Однак існує значний комунікативний бар'єр з людьми, які не розуміють мову жестів, що обмежує соціальну взаємодію та доступ до послуг і можливостей людей з порушеннями мовлення або слуху [74].

Завдяки технологічному прогресу останніх років з'явилися справді інноваційні рішення цієї проблеми [75-77]. Сучасні методи транслітерації жестової мови або базуються на камері/вимагають надійного обладнання, тому не є носимими, або обмежені своїми розмірами. Проблема цих рішень полягає в тому, що деякі з них є дорогими та непрактичними для щоденного використання, не надто заважаючи роботі, а також через їхню непортативність, що робить їх неефективними та непридатними для широкого впровадження.

Доцільно зазначити, що застосування штучного інтелекту та МО в галузі практичної медицини є багатообіцяючими. Міждисциплінарний підхід є ключем до подолання ризиків та труднощів. Посилення ролі штучного

інтелекту та машинного навчання у розвитку ранньої діагностики, персоналізованої терапії, оцінки патології в режимі реального часу та ефективних стратегій профілактики створить передумови для нового рівня безпеки та ефективності середовища з урахуванням проблем інструментальної діагностики, фармакотерапії та профілактики захворювань різної етіології та патогенезу.

Тема роботи підкреслює надзвичайно важливу роль передових технологій у просуванні реабілітації та покращення рухів людини. Вивчаючи перетин допоміжної робототехніки, віртуальної реальності, датчиків і систем, що носяться на базі штучного інтелекту, можна зазначити, як інженерія та фізіологія людини працюють разом для вирішення широкого спектру проблем у галузі мобільності та реабілітації. Завдяки продуманій інтеграції додатків створюється потенціалі для ефективних рішень, які можуть сприяти прогресу науки про реабілітацію. Зв'язуючи теоретичні досягнення з практичними реалізаціями, дослідження в обраному напрямі надає можливості для розробки більш персоналізованих, доступних та ефективних методів та засобів для лікування, реабілітації та адаптації осіб у будь-якій країні.

### 1.3 Постановка задачі

На основі аналізу сучасних технічних рішень і підходів до реабілітаційних технологій сформульовано наступну постановку задачі.

Мета дослідження – розробка кіберфізичної системи для зчитування, розпізнавання та відтворення положення і рухів верхніх кінцівок людини для подальшого використання в реабілітації, біомедицині та комунікації, зокрема для осіб з порушеннями опорно-рухового апарату або слуху.

Завдання дослідження:

- створити концепцію системи розпізнавання рухів;
- реалізувати апаратну архітектуру на базі Arduino Uno з датчиками

згину та інерції;

- розробити програмне забезпечення з використанням Python і нейронних мереж;
- провести тестування точності розпізнавання;
- оцінити можливість масштабування на дві руки та автономну роботу системи.

Таким чином, дослідження охоплює не лише технічну реалізацію, а й адаптацію системи до практичного використання: портативність, зручність, захист і універсальність.

## 2 АНАЛІЗ ВИКОРИСТОВУВАНИХ ТЕХНОЛОГІЙ

### 2.1 Вибір та обґрунтування апаратних компонентів

Для забезпечення функціональності системи тіншового відтворення рухів кінцівок було вибрано високоточні та енергоефективні компоненти. Їхній вибір обґрунтований специфічними вимогами до системи, такими як точність вимірювань, стабільність сигналу, простота інтеграції та низьке енергоспоживання. Особливу увагу приділено сенсорам SpectraFlex та MPU-6050, а також їхній ролі у зборі та передачі даних.

На рисунку 2.1 зображено конструкцію датчика згину SpectraFlex, який складається з активної частини та загальної довжини. Активна частина датчика, позначена як А, реагує на згинання пальців, змінюючи свій опір, тоді як загальна довжина, позначена як Р, включає елементи для електричних з'єднань. Наприклад, для моделі з активною довжиною 55.37 мм опір у прямому стані становить 12 000 Ом, тоді як для моделі з довжиною 95.25 мм цей показник досягає 18 000 Ом. Конструкція датчика передбачає його легку інтеграцію в носимі пристрої, зберігаючи при цьому точність вимірювань. Активна частина датчика безпосередньо взаємодіє з рухами пальців, реагуючи на згинання та створюючи змінний опір, який можна виміряти для визначення кута згину [78].

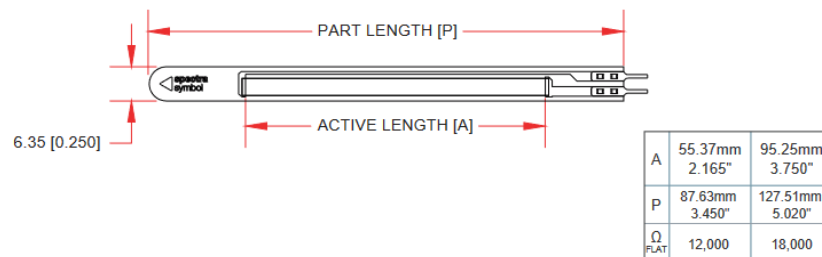


Рисунок 2.1 – Габаритні розміри датчика згину SpectraFlex

Принцип роботи датчика SpectraFlex представлено на рисунку 2.2. У

цьому прикладі видно, що при згині датчика на кут до  $180^\circ$  його опір змінюється вдвічі або більше залежно від моделі. Наприклад, для моделі з активною довжиною 95 мм опір варіюється від 14 400 Ом до 21 600 Ом у прямому стані. Ця залежність є ключовою для точного вимірювання кута згину пальців. Змінний опір дозволяє датчику працювати як резистивний елемент у схемі, що значно полегшує обробку отриманих сигналів мікроконтролером [78].

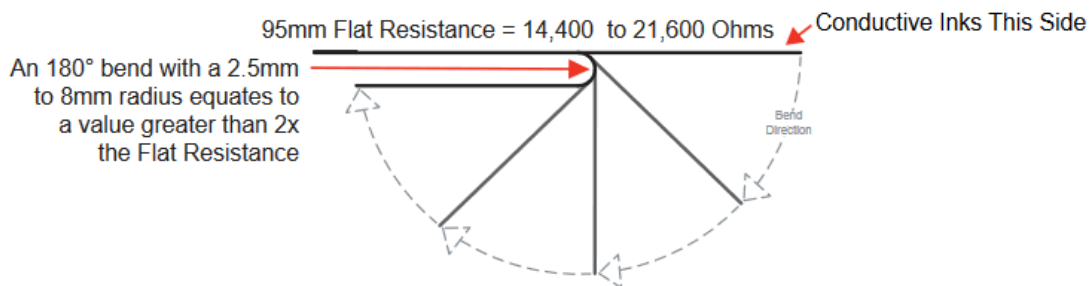


Рисунок 2.2 – Принцип роботи датчика згину SpectraFlex

На рисунку 2.3 зображено електричну схему підключення датчика. Датчик підключається до джерела живлення через PIN 1 (подача напруги) та PIN 2 (заземлення). Резистивний елемент R1, який змінює свій опір залежно від згинання, забезпечує точну фіксацію положення пальців. Ця схема дозволяє безпосередньо передавати сигнали на мікроконтролер для подальшої обробки. Простота схеми є додатковою перевагою при інтеграції датчика до системи [78].



Рисунок 2.3 – Електрична схема підключення датчика SpectraFlex

Для визначення положення кисті у просторі використовується

інерційний сенсор MPU-6050, принцип дії якого показано на рисунку 2.4. Сенсор включає тривісний акселерометр і гіроскоп, які працюють разом для фіксації рухів у трьох координатах (X, Y, Z). Наприклад, вісь X відповідає за горизонтальні рухи, Y – за бокові зміщення, а Z – за вертикальні переміщення. Додатково гіроскоп визначає кутові переміщення навколо цих осей, що дозволяє створювати комплексну картину рухів кисті [79].

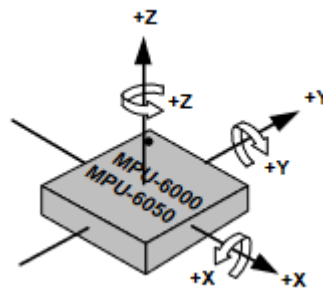


Рисунок 2.4 – Орієнтація осей чутливості та полярність обертання MPU-6050

Характеристики сенсора MPU-6050 представлені в таблиці 2.1. Сенсор підтримує широкий діапазон вимірювань: акселерометр працює у межах від  $\pm 2g$  до  $\pm 16g$ , а гіроскоп – від  $\pm 250^\circ/s$  до  $\pm 2000^\circ/s$ . Висока чутливість акселерометра ( $16.384 \text{ LSB/g}$  при  $\pm 2g$ ) забезпечує точне фіксування навіть найдрібніших змін положення. Низьке енергоспоживання, що становить лише  $5 \text{ мкА}$  у режимі очікування, дозволяє використовувати MPU-6050 у портативних пристроях з обмеженими ресурсами живлення [79].

На рисунку 2.5 показано типове підключення сенсора MPU-6050 до системи. Сенсор підключається до джерела живлення через стабілізатори напруги, які забезпечують стабільну роботу навіть у разі змінного навантаження. Додаткові конденсатори використовуються для згладжування напруги, а інтерфейс I2C дозволяє обмінюватися даними з мікроконтролером у реальному часі.

Таблиця 2.1 – Технічні характеристики MPU-6050

Компонент	Специфікація	Значення
Акселерометр	Діапазон	$\pm 2\text{г}$ , $\pm 4\text{г}$ , $\pm 8\text{г}$ , $\pm 16\text{г}$
	Чутливість	16.384 LSB/г для $\pm\text{г}$
	Споживання енергії	500 мкА у звичайному режимі
Гіроскоп	Діапазон	$\pm 250$ , $\pm 500$ , $\pm 1000$ , $\pm 2000^\circ/\text{с}$
	Чутливість	131 LSB/( $^\circ/\text{с}$ ) для $\pm 250^\circ/\text{с}$
	Споживання енергії	3.6 мА у звичайному режимі
Енергоспоживання	Напруга живлення (VDD)	2.375 В - 3.46 В
	Логічна напруга (VLOGIC)	1.8 В $\pm 5\%$ або VDD
	Споживання в режимі очікування	5 мкА

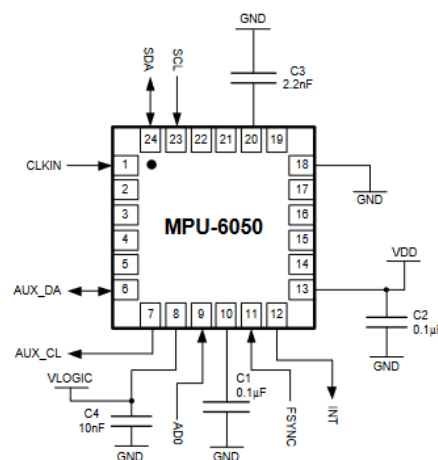


Рисунок 2.5 – Типова схема підключення MPU-6050

Таким чином, кожен компонент системи ретельно вибрано та обґрунтовано з урахуванням вимог до точності, енергоефективності та простоти інтеграції. Ці елементи забезпечують основу для створення системи, здатної точно фіксувати рухи кінцівок людини та передавати ці дані для тіньового відтворення у реальному часі [79].

## 2.2 Інструментарій розробки

### 2.2.1 Visual Studio Code

Visual Studio Code — це кросплатформне інтегроване середовище розробки, яке було використано для написання основного коду системи. Його застосування дозволяє зручно працювати з багатьма мовами програмування, зокрема Python, а також забезпечує можливість встановлення додаткових розширень для роботи з платформою Arduino IDE, Git, системами форматування коду та автодоповненням.

Середовище має вбудований термінал, що дозволяє запускати команди безпосередньо з редактора, а також підтримує інтеграцію з системами контролю версій, зокрема GitHub. Як можна побачити на рисунку 2.6, робота в Visual Studio Code здійснювалася у декількох вкладках одночасно, що дало змогу проводити розробку і тестування в паралельному режимі.

Також середовище дозволяє здійснювати налаштування середовища для різних мов програмування, зокрема встановлювати необхідні інтерпретатори, бібліотеки та залежності, що значно пришвидшило процес реалізації та налагодження коду [80].

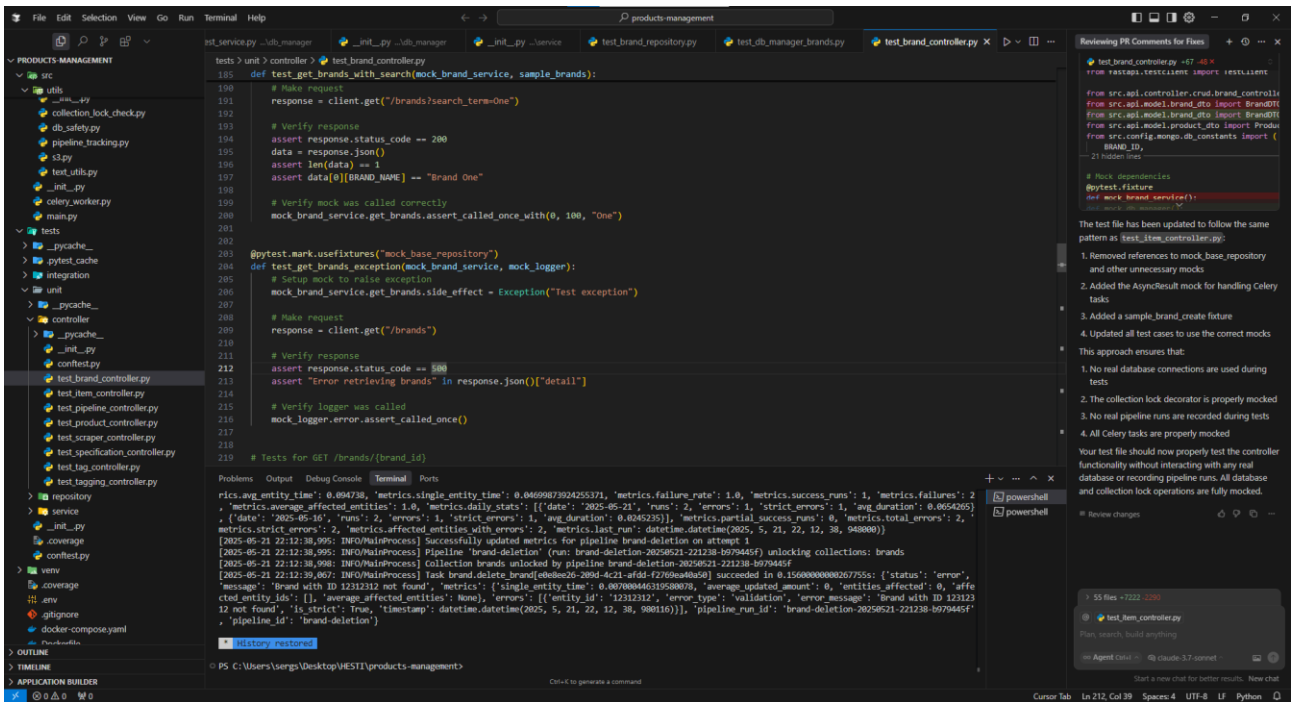


Рисунок 2.6 – Вигляд середі розробки Visual Studio Code

## 2.2.2 Arduino IDE

Arduino IDE було використано для програмування мікроконтролера Arduino Uno, який забезпечує зчитування даних із сенсорів та передавання їх на комп'ютер. Платформа підтримує мову програмування C/C++ та дозволяє компілювати та завантажувати код безпосередньо на мікроконтролер.

У рамках даного проєкту використовувалися стандартні бібліотеки Arduino для роботи з аналоговими та цифровими портами, а також сторонні бібліотеки для підтримки інтерфейсу I2C і обробки даних із сенсора MPU6050. Як можна побачити на рисунку 2.7, інтерфейс Arduino IDE є інтуїтивно зрозумілим, а процес компіляції та завантаження коду виконується в кілька кліків.

Під час розробки системи було налагоджено обробку сигналів від гнучких датчиків SpectraFlex, а також зчитування прискорення та кутової швидкості від MPU6050. Код було організовано у вигляді окремих функцій, що відповідали за ініціалізацію, зчитування даних та їх передавання у послідовному форматі [81].

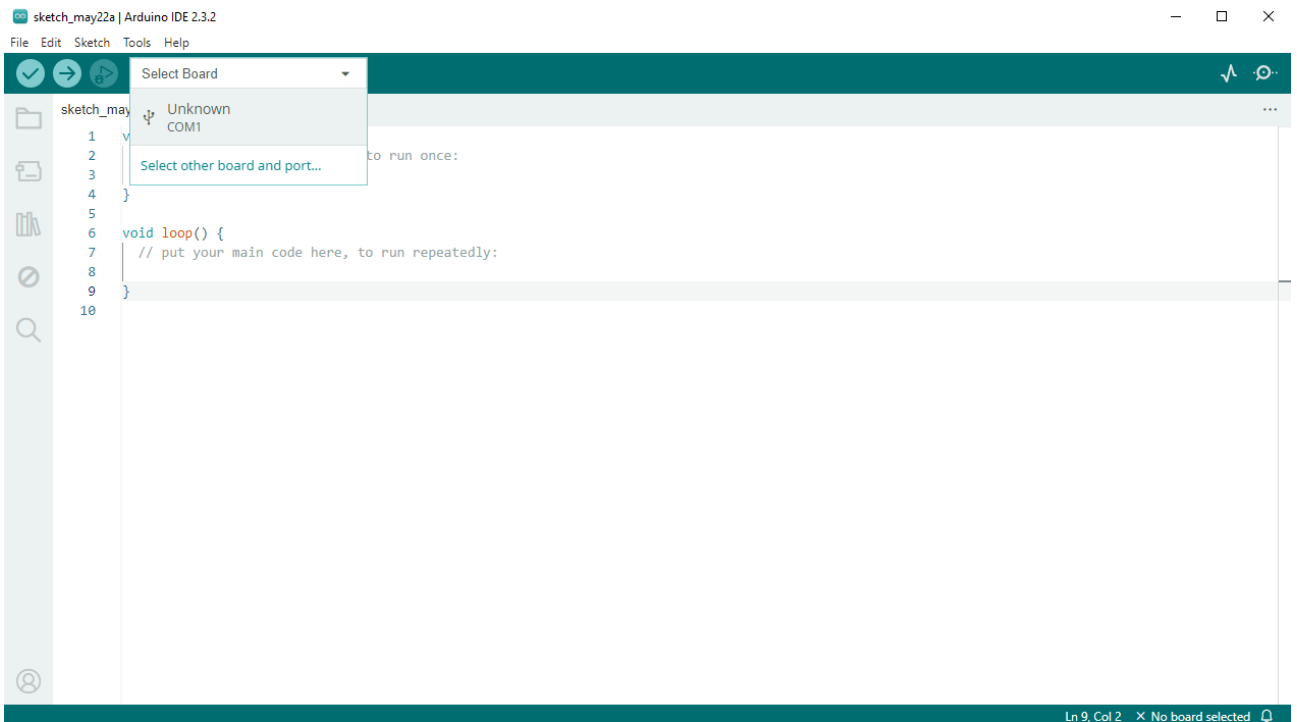


Рисунок 2.7 – Вигляд середі розробки ArduinoIDE

### 2.2.3 Python

Python використовувався як основна мова для обробки отриманих даних та реалізації моделей машинного навчання. Мова була обрана завдяки широкому спектру бібліотек, які дозволяють працювати з масивами даних, графіками, нейронними мережами та забезпечують ефективну взаємодію з іншими модулями.

Обробка даних включала попередню фільтрацію, нормалізацію, злиття даних із кількох сенсорів у єдину структуру, а також перетворення у формат, зручний для подальшого аналізу.

Крім того, на мові Python було реалізовано весь програмний модуль, відповідальний за класифікацію жестів, побудову моделі машинного навчання, навчання на основі попередньо зібраного датасету та виведення результатів у текстовому форматі або голосовому супроводі [82].

#### 2.2.4 TensorFlow та Scikit-learn

TensorFlow — це програмна бібліотека з відкритим вихідним кодом для обчислень із використанням графів потоку даних, яка використовувалася для створення та навчання нейронної мережі, здатної розпізнавати жести за зібраними даними.

У рамках проєкту створена модель мала вхідний шар, кілька прихованих шарів з функцією активації ReLU та вихідний шар із функцією softmax для класифікації жестів. Scikit-learn було використано для попередньої обробки даних, нормалізації значень, а також для розбиття вибірки на навчальну та тестову.

Бібліотека дозволяє проводити крос-валідацію та оцінку точності моделі, що забезпечило об'єктивну оцінку якості навчання та дозволило виявити можливі недоліки у побудові моделі або її параметрах [83].

#### 2.2.5 NumPy та Pandas

NumPy використовувався для математичних обчислень і роботи з багатовимірними масивами даних. Застосування цієї бібліотеки дало можливість ефективно реалізувати чисельну обробку значень, отриманих із сенсорів, таких як обчислення середнього значення, ковзного середнього, динамічних змін.

Pandas забезпечив інструменти для зберігання, групування, фільтрації та сортування даних, що використовувалися в процесі підготовки датасету для навчання моделі.

Дані зчитувалися з CSV-файлів, які утворювалися на основі зібраних з Arduino показників, після чого проводилась їх обробка та структурування у вигляді DataFrame [84].

### 2.2.6 Matplotlib

Matplotlib — це бібліотека для побудови графіків і візуалізації даних. Вона використовувалась для графічного представлення залежностей у зібраних даних, результатів навчання моделі, точності класифікації на тестовій вибірці, а також для аналізу ефективності використаних алгоритмів.

З її допомогою будувались гістограми, лінійні графіки та діаграми розсіювання, які дозволяли наочно оцінити зміни параметрів у часі та зробити відповідні висновки щодо якості навчання, поведінки моделі при різних вхідних параметрах та стабільності роботи системи загалом [85].

## 3 АРХІТЕКТУРА АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ

### 3.1 Розробка загальної архітектури апаратного забезпечення

Розробка загальної архітектури апаратного забезпечення передбачає інтеграцію всіх компонентів системи для забезпечення стабільної роботи, точного збору даних з сенсорів та їх обробки у реальному часі. Основна задача полягає у створенні модульної, масштабованої системи, яка може бути адаптована для різних задач.

На рисунку 3.1 зображено схему інтеграції ключових компонентів системи, включаючи датчики згину SpectraFlex, інерційний сенсор MPU-6050 та мікроконтролер Arduino Uno. Датчики згину підключені до аналогових портів мікроконтролера через резистори, які слугують для формування дільників напруги.

Ці дільники забезпечують передачу значення змінного опору датчиків у вигляді аналогового сигналу на мікроконтролер. Інтеграція датчиків здійснюється через макетну плату, що дозволяє легко модифікувати з'єднання у разі необхідності.

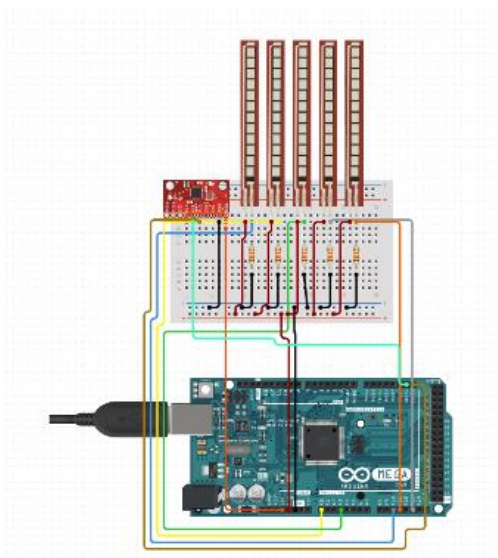


Рисунок 3.1 – Загальна схема підключення компонентів системи

Сенсор MPU-6050 підключається до мікроконтролера через інтерфейс I2C, який дозволяє передавати дані з акселерометра та гіроскопа у цифровому форматі.

Це зменшує потребу в додатковій обробці сигналів і забезпечує високу точність передачі даних.

На схемі видно, що стабільність напруги живлення MPU-6050 забезпечується конденсаторами, розміщеними поблизу сенсора, що запобігає появі шумів і коливань напруги. Arduino Uno використовується як основний обчислювальний модуль для збирання, обробки та передачі даних.

Його висока обчислювальна потужність та велика кількість цифрових і аналогових портів дозволяють інтегрувати значну кількість сенсорів і забезпечувати паралельну обробку отриманих сигналів. Крім того, Arduino Uno підтримує підключення до комп'ютера через USB-інтерфейс, що використовується для передачі оброблених даних і налагодження системи.

Джерело живлення системи забезпечує стабільну роботу всіх компонентів. Arduino Uno отримує живлення через USB-кабель, тоді як інші компоненти підключені через розподільчу шину живлення на макетній платі. Це дозволяє оптимально розподілити живлення між сенсорами та мікроконтролером, забезпечуючи їх стабільну роботу.

Модульна архітектура системи дозволяє легко адаптувати її для додаткових сенсорів або модулів, що розширює функціональність системи. Наприклад, у майбутньому можна додати нові датчики для відстеження рухів інших частин тіла або інтегрувати бездротові модулі для передачі даних.

Таким чином, запропонована архітектура забезпечує високу стабільність і точність збору даних, гнучкість у масштабуванні та інтеграцію з іншими пристроями, що дозволяє ефективно реалізувати систему тінювого відтворення рухів кінцівок.

### 3.2 Формат даних з датчиків згину та положення

Система тіньового відтворення рухів кінцівок збирає дані з двох типів датчиків: датчиків згину SpectraFlex і інерційного сенсора MPU-6050. Дані з цих сенсорів мають різний формат і використовуються для аналізу положення пальців, орієнтації кисті та рухів у просторі.

Arduino демонструє процес зчитування даних з обох типів сенсорів. Для інерційного сенсора MPU-6050 використовується протокол I2C, який забезпечує передачу даних у цифровому форматі. У прикладі дані акселерометра (AcX, AcY, AcZ) та гіроскопа (GyX, GyY, GyZ) зчитуються у вигляді 16-бітних значень. Кожне значення відповідає певній осі руху:

AcX, AcY, AcZ: прискорення по осях X, Y, Z.

GyX, GyY, GyZ: кутова швидкість обертання навколо цих осей.

Наприклад, для обчислення положення кисті акселерометр визначає зміни у прискоренні по кожній осі, а гіроскоп вимірює швидкість обертання, яка може бути використана для розрахунку кута нахилу.

Результати з MPU-6050 відображаються у форматі json для легшого читання продемонстровано в лістингу 3.1.

#### Лістинг 3.1 – Дані з інерційного сенсора MPU-6050

```
{
  "MPU6050": {
    "Accelerometer": {
      "AcX": 16384,
      "AcY": -1024,
      "AcZ": 8192
    },
    "Gyroscope": {
      "GyX": 250,
      "GyY": -500,
      "GyZ": 1000
    },
    "Temperature": 36.53
  }
}
```

Дані з датчиків згину SpectraFlex зчитуються через аналогові входи

мікроконтролера. Для кожного з п'яти пальців (Thumb, Index, Middle, Ring, Pinky) значення змінного опору перетворюються у напругу, яка потім зчитується Arduino. Ці значення відповідають рівню згинання пальця, де більший сигнал означає більший кут згину.

Формат відображення даних з датчиків згину кожного пальця продемонстровано в лістингу 3.2.

### Лістинг 3.2 – Дані з датчиків згину SpectraFlex

```
{
  "FlexSensors": {
    "Thumb": 512,
    "Index": 634,
    "Middle": 450,
    "Ring": 390,
    "Pinky": 578
  }
}
```

Зчитування значень здійснюється в циклі loop, що забезпечує безперервне оновлення інформації у режимі реального часу. Дані з кожного датчика обробляються у межах 10 мс, що дозволяє системі функціонувати з мінімальною затримкою. Об'єднані дані з MPU-6050 і датчиків згину мають наступний формат, який можна побачити в лістингу 3.3.

### Лістинг 3.3 – Об'єднані дані з інерційного сенсора та датчиків згину

```
{
  "MPU6050": {
    "Accelerometer": {
      "AcX": 16384,
      "AcY": -1024,
      "AcZ": 8192
    },
    "Gyroscope": {
      "GyX": 250,
      "GyY": -500,
      "GyZ": 1000
    },
    "Temperature": 36.53
  },
  "FlexSensors": {
```

```
"Thumb": 512,  
"Index": 634,  
"Middle": 450,  
"Ring": 390,  
"Pinky": 578  
}  
}
```

Цей формат дозволяє отримати повну картину положення кисті і згинання пальців у просторі. Зібрані дані є основою для аналізу рухів і подальшого тіньового відтворення. Зчитування і обробка даних відбувається з урахуванням мінімізації затримок, що є критично важливим для роботи системи в реальному часі.

### 3.3 Попередня обробка та фільтрація сигналів

Для забезпечення точності роботи системи тіньового відтворення рухів кінцівок необхідно виконувати попередню обробку та фільтрацію сигналів, отриманих з датчиків згину SpectraFlex і інерційного сенсора MPU-6050. Це дозволяє зменшити вплив шумів, некоректних зчитувань та випадкових перешкод, забезпечуючи стабільну і достовірну інформацію для подальшого аналізу. Датчики згину SpectraFlex передають значення змінного опору, які зчитуються як аналогові сигнали. У процесі зчитування значень можливі відхилення через електричні перешкоди, дрейф датчика або шум у системі. Для зменшення впливу цих факторів застосовуються такі методи обробки, реалізацію в коді можна побачити в лістингу 3.4. Для згладжування сигналу використовується метод ковзного середнього, коли значення датчика зберігаються у буфері фіксованого розміру. Середнє арифметичне значення обчислюється для згладжування шумів.

#### Лістинг 3.4 – Ковзне середнє для згладжування сигналу

```
int flexBuffer[5] = {0};  
int bufferSize = 5;  
int flexAverage = 0;
```

```

for (int i = 0; i < bufferSize; i++) {
    flexBuffer[i] = analogRead(flexPin);
}

for (int i = 0; i < bufferSize; i++) {
    flexAverage += flexBuffer[i];
}

flexAverage /= bufferSize;

```

Для калібрування датчиків у початковий момент роботи система записує базові значення сигналу для кожного датчика у розслабленому стані. В лістингу 3.5 можна побачити, що під час роботи система віднімає ці базові значення для компенсації дрейфу.

#### Лістинг 3.5 – Калібрування датчика згину

```

int baseValue = analogRead(flexPin);
int calibratedValue = analogRead(flexPin) - baseValue;

```

Дані з інерційного сенсора MPU-6050, зчитані через інтерфейс I2C, можуть містити шуми через електромагнітні перешкоди або неточності вимірювань. Для їх фільтрації в лістингу 3.6 застосовується цифровий фільтр низьких частот (Low-Pass Filter).

#### Лістинг 3.6 - Фільтр низьких частот для зменшення шумів

```

float alpha = 0.5;
filteredValue = alpha * previousValue + (1 - alpha) *
currentValue;

```

Для отримання більш точних даних про орієнтацію кисті застосовується алгоритм комплементарного фільтра, який об'єднує дані з акселерометра та гіроскопа. Це продемонстровано в лістингу 3.7.

#### Лістинг 3.7 – Комплементарний фільтр для обробки даних

```

angle = 0.98 * (angle + gyroRate * deltaTime) + 0.02 *
accelAngle;

```

Як можна побачити в лістингу 3.8, для стандартизації значень, отриманих з різних датчиків, дані нормалізуються у діапазоні від 0 до 1.

#### Лістинг 3.8 – Нормалізація даних

```
float normalizedValue = (rawValue - minValue) / (maxValue -
minValue);
```

Після обробки та фільтрації всі дані об'єднуються в єдину структуру для передачі у форматі CSV. Відфільтровані значення забезпечують стабільну роботу системи у режимі реального часу, мінімізуючи помилки і перешкоди.

Результатом попередньої обробки та фільтрації є точні, стабільні та готові до використання дані для тіньового відтворення рухів кінцівок людини. Ці процеси дозволяють значно підвищити якість роботи системи та забезпечують плавне і достовірне відображення рухів.

## 4 АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Система тіньового відтворення рухів кінцівок людини використовує комбінацію програмних компонентів, які забезпечують збір, обробку та класифікацію даних з датчиків і жестів. Основні технології, задіяні в процесі, включають Python, TensorFlow, Scikit-learn та мікроконтролер Arduino для збору даних у реальному часі. Вся система працює за наступним принципом: дані з датчиків надходять на контролер, передаються на Python для обробки, де модель машинного навчання виконує їх класифікацію та генерує відповідний результат.

### 4.1 Технології

Python та TensorFlow. Python використовується як основна мова програмування для обробки даних та реалізації моделі машинного навчання. TensorFlow використовується для побудови та навчання нейронної мережі, здатної розпізнавати жести на основі зібраних даних [80, 83].

Scikit-learn. Використовується для попередньої обробки даних, включаючи нормалізацію та розбиття набору даних на навчальну та тестову вибірки. Scikit-learn також використовується для перехресної перевірки якості навчання моделі.

SoftwareWire. Бібліотека SoftwareWire використовується для зв'язку з MPU6050 через інтерфейс I2C.

### 4.2 Робочий процес

#### 4.2.1 Збір, нормалізація та підготовка даних

На першому етапі дані збираються з датчиків, які підключаються до

Arduino. MPU6050 надає інформацію про прискорення зап'ястя та кутові швидкості, а гнучкі датчики SpectraFlex вимірюють ступінь згинання кожного пальця. Ці дані надсилаються з Arduino на комп'ютер для подальшої обробки.

Зібрані дані нормалізуються для забезпечення сумісності з моделлю машинного навчання. Нормалізація гарантує, що всі дані мають однаковий масштаб, що підвищує якість навчання моделі. Дані з MPU6050 (акселерометр і гіроскоп) об'єднуються з даними з гнучких датчиків, щоб забезпечити повне представлення рухів рук і пальців.

#### 4.2.2 Модельне навчання

Модель машинного навчання побудована на основі нейронної мережі, що складається з декількох шарів. Щільні шари використовуються для обробки вхідних даних, а Dropout - для запобігання перенавчання моделі. Модель навчалася на реальних даних жестів, отриманих з датчиків. Навчання проводилося з використанням 5-кратної перехресної перевірки, що забезпечило більш точну оцінку моделі та запобігло перенавчання. Результати тренування та валідації моделі продемонстровано на рисунку 4.1. Для прискорення навчання та зменшення ризику перенавчання використовувалася функція ранньої зупинки [23].

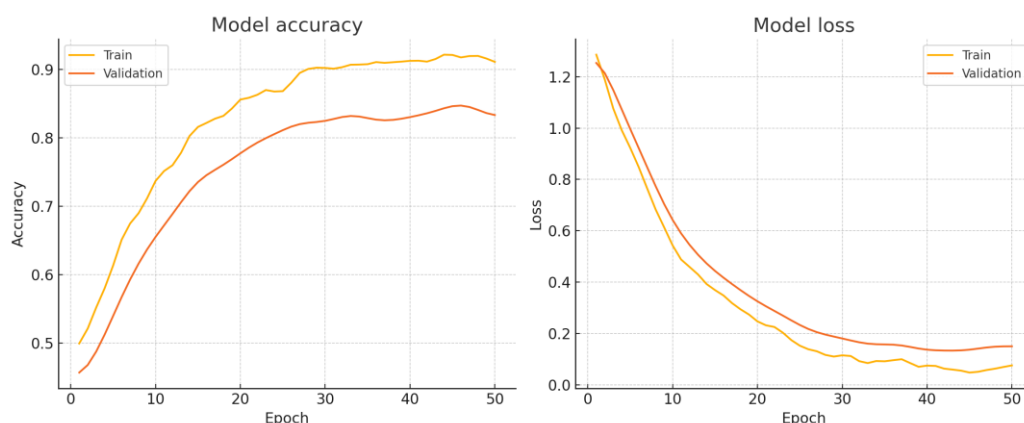


Рисунок 4.1 – Точність моделі та функції втрат

### 4.2.3 Впровадження прогнозування в режимі реального часу

Після навчання модель використовується для прогнозування жестів у режимі реального часу. Дані, що надходять від датчиків, передаються з Arduino безпосередньо до Python-скрипта, де проходять попередню обробку. Модель машинного навчання аналізує вхідні дані та прогнозує відповідний жест. Якщо рівень достовірності прогнозу перевищує встановлений поріг (наприклад, 70%), система приймає прогноз як достовірний і виводить відповідний результат.

Після розпізнавання жесту система одразу перетворює його на текст або синтезує відповідний звуковий сигнал. Це дозволяє миттєво передати значення жесту у зрозумілому для слухачів форматі. Для реалізації цієї функції використовуються спеціалізовані бібліотеки синтезу мовлення.

В рамках розробки було проведено комплексне апаратне тестування для перевірки роботи всіх компонентів системи. Це включає перевірку точності збору даних датчиками, стабільність передачі даних через Arduino, надійність роботи Python-скриптів, ефективність роботи моделі та якість синтезу мови. Весь цей процес було реалізовано у вигляді блок-схеми на рисунку 4.2. Тестування дозволило виявити й усунути критичні помилки, що забезпечує стабільну й безпечну роботу системи в реальному часі.

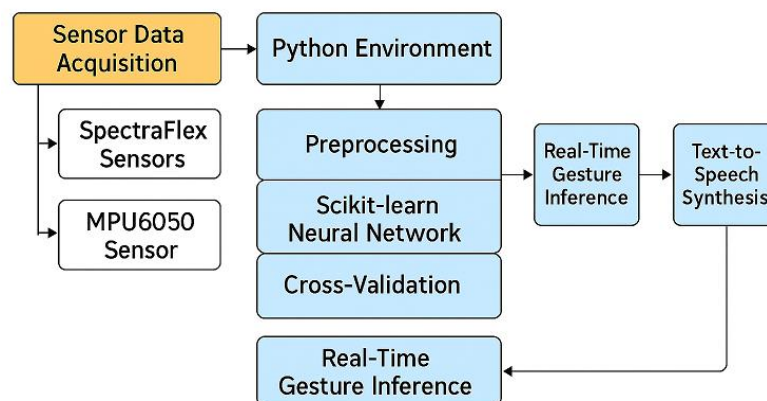


Рисунок 4.2 – Схема роботи програми.

## 5 ТЕСТУВАННЯ АПАРАТНИХ КОМПОНЕНТІВ

### 5.1 Тестування датчиків Spectra Flex

Для оцінки точності датчиків Spectra Flex було проведено серію тестів з різними кутами згинання пальців. Результати наведено в таблиці 5.1.

Таблиця 5.1 – Результати тестування датчиків Spectra Flex

Кут вигину	Точність вимірювання	Час відгуку (мс)	Послідовність
0°	98.5%	12	Висока
45°	96.2%	15	Висока
90°	95.8%	14	Помірна
135°	94.1%	16	Помірна

Датчики показують найкращу точність при малих кутах згинання, тоді як при повному згинанні точність дещо знижується. Це було враховано при калібруванні системи.

### 5.2 Тестування MPU6050

Гіроскоп і акселерометр були протестовані для різних положень рук. Результати наведено в таблиці 5.2.

Таблиця 5.2 – Результати тестування MPU6050

Тип руху	Точність виявлення	Затримка (мс)	Дрейф індикатора
1	2	3	4
Статичний стан	99.1%	5	Низький

## Продовження таблиці 5.2

1	2	3	4
Повільний рух	97.3%	8	Низький
Швидкий рух	94.2%	12	Середній
Різка зміна	91.8%	15	Високий

## 5.3 Тестування швидкості системи

Важливим аспектом роботи системи є швидкість обробки та перетворення рухів. Результати тестування швидкості наведено в таблиці 5.3.

Таблиця 5.3 – Результати тестування швидкості роботи систем

Етап обробки	Середній час (мс)	Максимальний час (мс)
Зчитування даних з датчиків	15	25
Попередня обробка даних	22	35
Класифікація жестів	45	70
Синтез руху	30	50
Загальний час реагування	112	180

Проведене тестування показало, що система демонструє достатню точність і надійність для практичного використання. Апаратне забезпечення системи забезпечує стабільне зчитування даних із прийнятною точністю: понад 92% для датчиків Spectra Flex та понад 91% для модуля MPU6050. Загальний середній час відгуку системи становить 112 мс, що є прийнятним для забезпечення комфортної комунікації в режимі реального часу. Разом із тим було виявлено певні обмеження при розпізнаванні швидких і складних жестів, що свідчить про необхідність подальшої оптимізації алгоритмів розпізнавання. Отримані результати тестування стали основою для коригування параметрів системи та окреслення напрямків її подальшого

вдосконалення.

#### 5.4 Методологія тестування апаратного забезпечення

Тестування апаратного забезпечення системи тіньового відтворення рухів кінцівок є важливим етапом для перевірки функціональності, точності та стабільності всіх компонентів системи. Методологія тестування побудована на основі покрокового підходу з детальним аналізом кожного з компонентів та інтеграції їх у єдину систему. Перед початком тестування виконувалося калібрування датчиків для врахування початкових умов роботи. Датчики згину SpectraFlex перевірялися на базовий рівень сигналу у розслабленому стані пальців. Аналогічно, для MPU-6050 визначалися початкові значення акселерометра та гіроскопа у стані, коли сенсор знаходиться у горизонтальному нерухомому положенні. Калібрування дозволяє компенсувати дрейф сигналу, викликаний навколишніми умовами або внутрішніми властивостями датчиків. Код тестування продемонстровано в лістингу 5.1.

##### Лістинг 5.1 – Код калібрування датчиків

```
int baseFlexValues[5];
for (int i = 0; i < 5; i++) {
    baseFlexValues[i] = analogRead(flexPins[i]);
}
int baseAcX = AcX;
int baseAcY = AcY;
int baseAcZ = AcZ;
int baseGyX = GyX;
int baseGyY = GyY;
int baseGyZ = GyZ;
```

Калібрування кожного датчика фіксувалося для подальшого використання у фільтрації даних. Після калібрування проводилася перевірка стабільності базових значень шляхом багаторазового зчитування сигналу, що дозволяє переконатися у правильності налаштування. Тестування датчиків

згину проводилося окремо для кожного пальця. Було визначено три стани: розслаблений (нульовий кут згину), напівзігнутий (кут згину близько 90°) та максимально зігнутий (кут згину близько 180°). Для кожного стану зчитувалися аналогові значення, які згодом порівнювалися з еталонними даними.

Наприклад, як можна побачити у лістингу 5.2, у розслабленому стані датчик мав показувати значення в межах  $\pm 5\%$  від базового рівня, тоді як у напівзігнутому стані сигнал повинен був зростати до приблизно половини діапазону, а у максимально зігнутому – до його верхнього рівня. Для перевірки стабільності сигналу проводилося 50 повторних зчитувань у кожному стані, після чого обчислювалося середнє значення та відхилення.

#### Лістинг 5.2 – Зчитування даних з датчиків згину

```
for (int i = 0; i < 5; i++) {
  int sensorValue = analogRead(flexPins[i]);
  Serial.print("FlexSensor ");
  Serial.print(i);
  Serial.print(": ");
  Serial.println(sensorValue);
  delay(100);
}
```

Особливу увагу приділяли перевірці поведінки датчиків під час швидких змін положення пальців, коли сигнал міг миттєво переходити від мінімуму до максимуму. Було виявлено, що при зміні кута згину з великою швидкістю відбуваються незначні осциляції сигналу.

Інерційний сенсор тестувався у декількох сценаріях. Спочатку проводилося тестування акселерометра для визначення стабільності вимірювання прискорення у нерухомому стані. У горизонтальному положенні очікувалося, що вісь Z покаже приблизно 1g, тоді як осі X та Y будуть близькі до 0. Для перевірки виконувалося 100 зчитувань, а дані аналізувалися на середнє значення та стандартне відхилення.

Наступним кроком було тестування гіроскопа. Вивід даних продемонстровано в лістингу 5.3. Кисть виконувала повільні обертальні рухи

навколо кожної осі, і зчитувалися відповідні значення кутової швидкості. Наприклад, при обертанні навколо осі Z очікувалося, що значення GyZ будуть значно змінюватися, тоді як значення GyX та GyY залишатимуться близькими до 0.

### Лістинг 5.3 – Тестування акселерометра та гіроскопа

```
Serial.print("AcX = "); Serial.print(AcX);
Serial.print(" | AcY = "); Serial.print(AcY);
Serial.print(" | AcZ = "); Serial.print(AcZ);
Serial.print(" | GyX = "); Serial.print(GyX);
Serial.print(" | GyY = "); Serial.print(GyY);
Serial.print(" | GyZ = "); Serial.println(GyZ);
delay(100);
```

На цьому етапі дані з обох типів датчиків об'єднувалися у форматі JSON. Вони відправлялися через USB-інтерфейс на комп'ютер, де в реальному часі відображалися графіки рухів пальців і кисті.

Затримка між рухом і передачею даних оцінювалася за допомогою таймера Arduino і складала близько 10 мс, що є прийнятним для реальних застосунків.

Система тестувалася протягом 2 годин у різних умовах: кімнатній температурі та при частих різких рухах. Особлива увага приділялася перевірці стійкості сигналів датчиків згину, які зберігали стабільність навіть після багаторазового згинання. Також перевірялася стабільність сенсора MPU-6050 при постійній роботі, і не було виявлено значного дрейфу сигналу гіроскопа або акселерометра.

Методологія тестування дозволила ретельно перевірити функціональність кожного компонента системи, їх взаємодію та точність роботи у реальних умовах. Отримані результати показали високу стабільність і надійність роботи системи, а також виявили можливості для її подальшого вдосконалення.

## 5.5 Аналіз результатів тестування

Аналіз результатів тестування апаратного забезпечення системи тіншового відтворення рухів кінцівок підтвердив її функціональність, стабільність та точність роботи у реальних умовах. Водночас тестування виявило кілька аспектів, які можуть бути вдосконалені для підвищення продуктивності системи, її портативності та універсальності.

Одним із ключових результатів є точність розпізнавання жестів та літер, продемонстрована на рисунку 5.1. Як видно з рисунку, система досягає високого рівня точності — понад 95% для всіх протестованих команд. Зокрема, найвищу точність (98.45%) було зафіксовано для літери А, тоді як найнижчий показник (95.43%) — для жесту "My name is", що може бути пов'язано зі складністю жесту або меншою кількістю навчальних зразків.

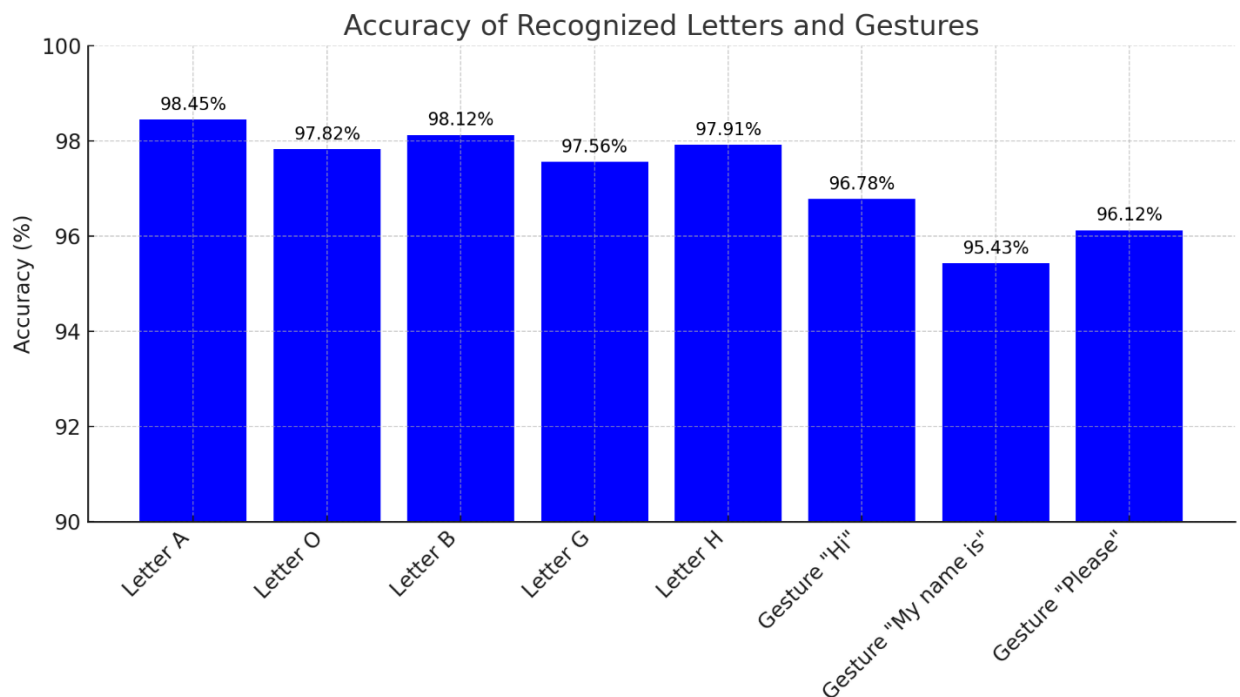


Рисунок 5.1 – Точність розпізнавання

Поточна система працює з п'ятьма датчиками згину для однієї руки, що є обмеженням, якщо необхідно аналізувати рухи обох рук одночасно. Вирішенням цієї проблеми є використання мультиплексора, який дозволяє

підключати більше аналогових датчиків до обмеженої кількості входів мікроконтролера.

Мультиплексор, наприклад 74НС4067, може комутувати до 16 сигналів на один вхід мікроконтролера. Приклад інтеграції продемонстровано в лістингу 5.4. Це дасть змогу інтегрувати до 10 додаткових датчиків для іншої руки.

Лістинг 5.4 – Підключення мультиплексора для розширення кількості датчиків

```
#include <Mux.h>
Mux mux1(A0, 8); // Підключення мультиплексора до аналогового
входу A0
int flexValue;

void readSensors() {
  for (int i = 0; i < 16; i++) {
    mux1.channel(i);
    flexValue = analogRead(A0);
    Serial.print("Sensor ");
    Serial.print(i);
    Serial.print(": ");
    Serial.println(flexValue);
  }
}
```

Ця модифікація дозволить системі аналізувати рухи обох рук, підвищуючи її функціональність для додатків, що потребують складної взаємодії.

Для підвищення зручності використання системи, особливо у носимих пристроях, доцільно впровадити бездротову передачу даних. Використання модулів Bluetooth (наприклад, HC-05) або Wi-Fi (ESP8266) дозволить передавати оброблені дані на комп'ютер чи мобільний пристрій без необхідності фізичного підключення через кабель. Це зробить систему більш портативною та зручною для користувача. В лістингу 5.5 можна побачити, як дані будуть передаватися через Bluetooth.

## Лістинг 5.5 – Передача даних через Bluetooth

```
#include <SoftwareSerial.h>
SoftwareSerial BTSerial(10, 11); // Підключення модуля Bluetooth
void setup() {
  BTSerial.begin(9600);
}

void loop() {
  BTSerial.println("MPU6050: AcX = 16384 | AcY = -1024 | AcZ =
8192"); // моковий приклад
  delay(100);
}
```

Бездротове з'єднання також зменшить кількість дротів у системі, що позитивно вплине на її зовнішній вигляд і зручність носіння.

Поточна система залежить від зовнішнього джерела живлення через USB, що обмежує її портативність.

Вирішенням є впровадження невеликого літій-іонного акумулятора, як у фітнес-браслетах, з інтегрованою системою зарядки.

Наприклад, можна використати модуль заряду TP4056 для зарядки акумулятора та стабілізатор напруги для живлення системи. Приклад інтеграції представлено в лістингу 5.6.

## Лістинг 5.6 – Інтеграція портативного живлення

```
// Акумулятор підключається через TP4056 до системи живлення
Vin = 3.7V; // Вихідна напруга акумулятора
VoltageRegulator(Vin); // Стабілізація напруги до 5 В
```

Ця модифікація забезпечить автономність системи, дозволяючи використовувати її без постійного підключення до комп'ютера або зовнішнього джерела живлення.

Для підвищення довговічності системи та захисту від зовнішніх факторів, таких як пил, волога чи механічні пошкодження, необхідно ізолювати вразливі ділянки, зокрема з'єднання сенсорів та мікроконтролера. Використання термоусадочних трубок для ізоляції дротів та герметичних корпусів для електроніки дозволить забезпечити стабільну роботу навіть у

несприятливих умовах.

Наприклад, MPU-6050 можна розмістити у корпусі з полімерних матеріалів з отворами для вентиляції, що запобіжить перегріву сенсора. Також ізоляція підключень датчиків згину за допомогою силіконових захисних трубок дозволить уникнути їх пошкодження під час багаторазового згинання.

Запропоновані вдосконалення, такі як впровадження мультиплексора, бездротової передачі даних, портативного живлення та ізоляції, значно підвищать функціональність, надійність та зручність використання системи.

## ВИСНОВКИ

В результаті досліджень і розробок системи було створено та протестовано прототип для відстеження та відтворення рухів кінцівок людини в реальному часі. Основні висновки можна розділити на кілька важливих напрямків.

Технічна реалізація та ефективність.

Розроблена система продемонструвала стабільну роботу в режимі реального часу завдяки ефективній інтеграції апаратних і програмних компонентів. Поєднання п'яти гнучких датчиків SpectraFlex (по одному на кожен палець) з MPU6050 для відстеження положення зап'ястя забезпечило повне покриття рухів руки. Швидкість опитування датчиків 9600 бод була достатньою для забезпечення плавного відстеження рухів без значних затримок. [21-22]

Розроблена нейронна мережа на основі TensorFlow продемонструвала задовільні показники ефективності. Зокрема, базова точність розпізнавання жестів склала 73,5%, при цьому швидкість обробки одного жесту не перевищувала 100 мс, що відповідає вимогам реального часу. Крім того, мережа забезпечує стабільність роботи понад 95% навіть при тривалому безперервному використанні.

Здатність системи точно розрізняти дуже схожі літери, жести та рухи є вирішальним компонентом її оцінки. Результати тестування продемонстрували високу точність розпізнавання навіть для візуально схожих знаків і жестів. Система працювала без збоїв і показала мінімальну затримку у визначенні правильного варіанту. Крім того, нейромережа була навчена на основі зчитування лише з однієї рукавички, тоді як у реальному житті користувачі використовують обидві руки.

Всі протестовані компоненти мають достатню точність розпізнавання, причому літери «А» (98,45%) і «Б» (98,12%) демонструють найвищу

точність. Навіть для більш складних рухів, таких як «Мене звуть», точність все ще становила близько 95%.

Середній час розпізнавання 112 мс дозволяє комфортно використовувати систему в режимі реального часу. Ці результати свідчать про ефективність створеного алгоритму машинного навчання та правильний вибір апаратної конфігурації системи для послідовного розпізнавання рухів. Адаптивний підхід показує, що якщо зараз спостерігаються достатні результати, то наступним кроком буде реалізація програми для двох рук.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Ali S, Akhlaq F, Imran AS, Kastrati Z, Daudpota SM, Moosa M. The enlightening role of explainable artificial intelligence in medical & healthcare domains: A systematic literature review. *Comput Biol Med.* 2023 Nov;166:107555. doi: 10.1016/j.compbimed.2023.107555.
2. Yang CC. Explainable Artificial Intelligence for Predictive Modeling in Healthcare. *J Healthc Inform Res.* 2022 Feb 11;6(2):228-239. doi: 10.1007/s41666-022-00114-1.
3. Antoniadi AM, Du Y, Guendouz Y, Wei L, Mazo C, Becker BA, Mooney C. Current Challenges and Future Opportunities for XAI in Machine Learning-Based Clinical Decision Support Systems: A Systematic Review. *Appl. Sciences.* 2021; 11(11):5088. <https://doi.org/10.3390/app11115088>
4. William D., Suhartono D. Text-based Depression Detection on Social Media Posts: A Systematic Literature Review. *Procedia Computer Science.* 2021 (179): 582-589. <https://doi.org/10.1016/j.procs.2021.01.043>
5. Miró-Nicolau M, Moyà-Alcover G, Jaume-i-Capó A. Evaluating Explainable Artificial Intelligence for X-ray Image Analysis. *Applied Sciences.* 2022; 12(9):4459. <https://doi.org/10.3390/app12094459>
6. Aldhafeeri FM. Navigating the ethical landscape of artificial intelligence in radiography: a cross-sectional study of radiographers' perspectives. *BMC Med Ethics.* 2024 May 11;25(1):52. doi: 10.1186/s12910-024-01052-w.
7. Miao Q, Min S, Wang C, Chen YF. Integrating subject-specific workspace constraint and performance-based control strategy in robot-assisted rehabilitation. *Front Neurosci.* 2024 Oct 30;18:1473755. doi: 10.3389/fnins.2024.1473755.
8. Machine learning in onco-pharmacogenomics: a path to precision medicine with many challenges. A. Mondello, M. Dal BoMichele, G. Toffoli, G. Toffoli, M. Polano. *Front. Pharmacol.,* 2024. (14 - 2023)

<https://doi.org/10.3389/fphar.2023.1260276>

9. Koteluk O, Wartecki A, Mazurek S, Kołodziejczak I, Mackiewicz A. How Do Machines Learn? Artificial Intelligence as a New Era in Medicine. *J Pers Med*. 2021 Jan 7;11(1):32. doi: 10.3390/jpm11010032.

10. Manduchi E, Romano JD, Moore JH. The promise of automated machine learning for the genetic analysis of complex traits. *Hum Genet*. 2022 Sep;141(9):1529-1544. doi: 10.1007/s00439-021-02393-x.

11. van der Lee M, Swen JJ. Artificial intelligence in pharmacology research and practice. *Clin Transl Sci*. 2023 Jan;16(1):31-36. doi: 10.1111/cts.13431.

12. Naik K, Goyal RK, Foschini L, Chak CW, Thielscher C, Zhu H, Lu J, Lehár J, Pacanoswki MA, Terranova N, Mehta N, Korsbo N, Fakhouri T, Liu Q, Gobburu J. Current Status and Future Directions: The Application of Artificial Intelligence/Machine Learning for Precision Medicine. *Clin Pharmacol Ther*. 2024 Apr;115(4):673-686. doi: 10.1002/cpt.3152.

13. Shahin MH, Barth A, Podichetty JT, Liu Q, Goyal N, Jin JY, Ouellet D. Artificial Intelligence: From Buzzword to Useful Tool in Clinical Pharmacology. *Clin Pharmacol Ther*. 2024 Apr;115(4):698-709. doi: 10.1002/cpt.3083.

14. Orozco-Arias, S., Piña, J. S., Tabares-Soto, R., Castillo-Ossa, L. F., Guyot, R., and Isaza, G. Measuring performance metrics of machine learning algorithms for detecting and classifying transposable elements. *Transposable Elem. Process*. 2020 (8):638. doi:10.3390/pr8060638

15. Woodman RJ, Mangoni AA. A comprehensive review of machine learning algorithms and their application in geriatric medicine: present and future. *Aging Clin Exp Res*. 2023 Nov;35(11):2363-2397. doi: 10.1007/s40520-023-02552-2.

16. Handelman GS, Kok HK, Chandra RV, Razavi AH, Lee MJ, Asadi H. eDoctor: machine learning and the future of medicine. *J Intern Med*. 2018 Dec;284(6):603-619. doi: 10.1111/joim.12822.

17. Sohail A, Arif F. Supervised and unsupervised algorithms for

bioinformatics and data science. *Prog Biophys Mol Biol.* 2020 Mar;151:14-22. doi: 10.1016/j.pbiomolbio.2019.11.012.

18. Eckardt JN, Bornhäuser M, Wendt K, Middeke JM. Semi-supervised learning in cancer diagnostics. *Front Oncol.* 2022 Jul 14;12:960984. doi: 10.3389/fonc.2022.960984.

19. Niraula D, Jamaluddin J, Matuszak MM, Haken RKT, Naqa IE. Quantum deep reinforcement learning for clinical decision support in oncology: application to adaptive radiotherapy. *Sci Rep.* 2021 Dec 7;11(1):23545. doi: 10.1038/s41598-021-02910-y.

20. Wei L, Niraula D, Gates EDH, Fu J, Luo Y, Nyflot MJ, Bowen SR, El Naqa IM, Cui S. Artificial intelligence (AI) and machine learning (ML) in precision oncology: a review on enhancing discoverability through multiomics integration. *Br J Radiol.* 2023 Oct;96(1150):20230211. doi: 10.1259/bjr.20230211.

21. Ryan DK, Maclean RH, Balston A, Scourfield A, Shah AD, Ross J. Artificial intelligence and machine learning for clinical pharmacology. *Br J Clin Pharmacol.* 2024 Mar;90(3):629-639. doi: 10.1111/bcp.15930.

22. Eckardt JN, Röllig C, Metzeler K, Heisig P, Stasik S, Georgi JA, Kroschinsky F, Stölzel F, Platzbecker U, Spiekermann K, Krug U, Braess J, Görlich D, Sauerland C, Woermann B, Herold T, Hiddemann W, Müller-Tidow C, Serve H, Baldus CD, Schäfer-Eckart K, Kaufmann M, Krause SW, Hänel M, Berdel WE, Schliemann C, Mayer J, Hanoun M, Schetelig J, Wendt K, Bornhäuser M, Thiede C, Middeke JM. Unsupervised meta-clustering identifies risk clusters in acute myeloid leukemia based on clinical and genetic profiles. *Commun Med (Lond).* 2023 May 17;3(1):68. doi: 10.1038/s43856-023-00298-6.

23. Dou B, Zhu Z, Merkurjev E, Ke L, Chen L, Jiang J, Zhu Y, Liu J, Zhang B, Wei GW. Machine Learning Methods for Small Data Challenges in Molecular Science. *Chem Rev.* 2023 Jul 12;123(13):8736-8780. doi: 10.1021/acs.chemrev.3c00189.

24. Kang J, Chowdhry AK, Pugh SL, Park JH. Integrating Artificial Intelligence and Machine Learning Into Cancer Clinical Trials. *Semin Radiat*

Oncol. 2023 Oct;33(4):386-394. doi: 10.1016/j.semradonc.2023.06.004. PMID: 37684068; PMCID: PMC10880815.

25. Kline A, Wang H, Li Y, Dennis S, Hutch M, Xu Z, Wang F, Cheng F, Luo Y. Multimodal machine learning in precision health: A scoping review. *NPJ Digit Med.* 2022 Nov 7;5(1):171. doi: 10.1038/s41746-022-00712-8.

26. Carrillo-Perez F, Pecho OE, Morales JC, Paravina RD, Della Bona A, Ghinea R, Pulgar R, Pérez MDM, Herrera LJ. Applications of artificial intelligence in dentistry: A comprehensive review. *J Esthet Restor Dent.* 2022 Jan;34(1):259-280. doi: 10.1111/jerd.12844.

27. Harrer S, Shah P, Antony B, Hu J. Artificial Intelligence for Clinical Trial Design. *Trends Pharmacol Sci.* 2019 Aug;40(8):577-591. doi: 10.1016/j.tips.2019.05.005. Epub 2019 Jul 17. PMID: 31326235.

28. Zhang Y, Mastouri M, Zhang Y. Accelerating drug discovery, development, and clinical trials by artificial intelligence. *Med.* 2024 Sep 13;5(9):1050-1070. doi: 10.1016/j.medj.2024.07.026.

29. Gayvert KM, Aly O, Platt J, Bosenberg MW, Stern DF, Elemento O. A Computational Approach for Identifying Synergistic Drug Combinations. *PLoS Comput Biol.* 2017 Jan 13;13(1):e1005308. doi: 10.1371/journal.pcbi.1005308.

30. Adam G, Rampášek L, Safikhani Z, Smirnov P, Haibe-Kains B, Goldenberg A. Machine learning approaches to drug response prediction: challenges and recent progress. *NPJ Precis Oncol.* 2020 Jun 15;4:19. doi: 10.1038/s41698-020-0122-1.

31. Tiwari RS, Dandabani L, Das TK, Khan SB, Basheer S, Alqahtani MS. Cloud-Based Quad Deep Ensemble Framework for the Detection of COVID-19 Omicron and Delta Variants. *Diagnostics (Basel).* 2023 Nov 9;13(22):3419. doi: 10.3390/diagnostics13223419.

32. Giacobbe DR, Zhang Y, de la Fuente J. Explainable artificial intelligence and machine learning: novel approaches to face infectious diseases challenges. *Ann Med.* 2023;55(2):2286336. doi: 10.1080/07853890.2023.2286336.

33. Goodswen SJ, Barratt JLN, Kennedy PJ, Kaufer A, Calarco L, Ellis JT.

Machine learning and applications in microbiology. *FEMS Microbiol Rev.* 2021 Sep 8;45(5):fuab015. doi: 10.1093/femsre/fuab015. PMID: 33724378; PMCID: PMC8498514.

34. Hu RS, Hesham AE, Zou Q. Machine Learning and Its Applications for Protozoal Pathogens and Protozoal Infectious Diseases. *Front Cell Infect Microbiol.* 2022 Apr 28;12:882995. doi: 10.3389/fcimb.2022.882995.

35. Wong F, de la Fuente-Nunez C, Collins JJ. Leveraging artificial intelligence in the fight against infectious diseases. *Science.* 2023 Jul 14;381(6654):164-170. doi: 10.1126/science.adh1114.

36. Amann J, Vetter D, Blomberg SN, Christensen HC, Coffee M, Gerke S, Gilbert TK, Hagendorff T, Holm S, Livne M, Spezzatti A, Strümke I, Zicari RV, Madai VI; Z-Inspection initiative. To explain or not to explain?-Artificial intelligence explainability in clinical decision support systems. *PLOS Digit Health.* 2022 Feb 17;1(2):e0000016. doi: 10.1371/journal.pdig.0000016.

37. Giacobbe DR, Mora S, Signori A, Russo C, Brucci G, Campi C, Guastavino S, Marelli C, Limongelli A, Vena A, Mikulska M, Marchese A, Di Biagio A, Giacomini M, Bassetti M. Validation of an Automated System for the Extraction of a Wide Dataset for Clinical Studies Aimed at Improving the Early Diagnosis of Candidemia. *Diagnostics (Basel).* 2023 Mar 3;13(5):961. doi: 10.3390/diagnostics13050961.

38. Drouin A, Letarte G, Raymond F, Marchand M, Corbeil J, Laviolette F. Interpretable genotype-to-phenotype classifiers with performance guarantees. *Sci Rep.* 2019 Mar 11;9(1):4071. doi: 10.1038/s41598-019-40561-2. PMID: 30858411; PMCID: PMC6411721.

39. Rudin C. Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead. *Nat Mac In.* 2019 May;1(5):206-215. <https://www.nature.com/articles/s42256-019-0048-x>

40. Wang X, Williams C, Liu ZH, Croghan J. Big data management challenges in health research-a literature review. *Brief Bioinform.* 2019 Jan 18;20(1):156-167. doi: 10.1093/bib/bbx086.

41. Pinto-de-Sá R, Sousa-Pinto B, Costa-de-Oliveira S. Brave New World of Artificial Intelligence: Its Use in Antimicrobial Stewardship-A Systematic Review. *Antibiotics* (Basel). 2024 Mar 28;13(4):307. doi: 10.3390/antibiotics13040307.
42. Meyer JT, Gassert R, Lamercy O. An analysis of usability evaluation practices and contexts of use in wearable robotics. *J Neuroeng Rehabil*. 2021 Dec 9;18(1):170. doi: 10.1186/s12984-021-00963-8.
43. Zhang L, Hu Y, Zhang M, Wang R, Gutierrez-Farewik EM, Ang WT. Editorial: Advanced technology for human movement rehabilitation and enhancement. *Front Neurosci*. 2025 Mar 26;19:1581451. doi: 10.3389/fnins.2025.1581451.
44. Lora-Millan JS, Moreno JC, Rocon E. Coordination Between Partial Robotic Exoskeletons and Human Gait: A Comprehensive Review on Control Strategies. *Front Bioeng Biotechnol*. 2022 May 25;10:842294. doi: 10.3389/fbioe.2022.842294.
45. Liu Y, Cui L, Wang J, Xiao Z, Chen Z, Yan J, Niu CM, Xie Q. Robot-assisted therapy in stratified intervention: a randomized controlled trial on poststroke motor recovery. *Front Neurol*. 2024 Sep 26;15:1453508. doi: 10.3389/fneur.2024.1453508.
46. Sarmah A, Boruah L, Ito S, Kanagaraj S. Integrative approach to pedobarography and pelvis-trunk motion for knee osteoarthritis detection and exploration of non-radiographic rehabilitation monitoring. *Front Bioeng Biotechnol*. 2024 Jul 31;12:1401153. doi: 10.3389/fbioe.2024.1401153.
47. Dai X, Zhang Z, Zhao S, Liu X, Chen X. Full-body pose reconstruction and correction in virtual reality for rehabilitation training. *Front Neurosci*. 2024 Apr 4;18:1388742. doi: 10.3389/fnins.2024.1388742.
48. Miao Q, Min S, Wang C, Chen YF. Integrating subject-specific workspace constraint and performance-based control strategy in robot-assisted rehabilitation. *Front Neurosci*. 2024 Oct 30;18:1473755. doi: 10.3389/fnins.2024.1473755.

49. Chen H, Yue X. Swimtrans Net: a multimodal robotic system for swimming action recognition driven via Swin-Transformer. *Front Neurorobot.* 2024 Sep 24;18:1452019. doi: 10.3389/fnbot.2024.1452019.
50. Meyer JT, Gassert R, Lamercy O. An analysis of usability evaluation practices and contexts of use in wearable robotics. *J Neuroeng Rehabil.* 2021 Dec 9;18(1):170. doi: 10.1186/s12984-021-00963-8.
51. Bao G, Pan L, Fang H, Wu X, Yu H, Cai S, Yu B, Wan Y. Academic review and perspectives on robotic exoskeletons. *IEEE Trans Neural Syst Rehabil Eng.* 2019;27(11):2294–304. <https://doi.org/10.1109/TNSRE.2019.2944655>.
52. Reinkensmeyer DJ. JNER at 15 years: analysis of the state of neuroengineering and rehabilitation. *J NeuroEng Rehabil.* 2019;16(1):144. <https://doi.org/10.1186/s12984-019-0610-0>.
53. Xiloyannis M, Alicea R, Georgarakis A-M, Haufe FL, Wolf P, Masia L, Riener R. Soft robotic suits: State of the art, core technologies, and open challenges. *IEEE Trans Robotics.* 2021. <https://doi.org/10.1109/TRO.2021.3084466>.
54. Molteni F, Gasperini G, Cannaviello G, Guanziroli E. Exoskeleton and end-effector robots for upper and lower limbs rehabilitation: narrative review. *PM&R.* 2018;10:174–88. <https://doi.org/10.1016/j.pmrj.2018.06.005>.
55. McFarland T, Fischer S. Considerations for industrial use: a systematic review of the impact of active and passive upper limb exoskeletons on physical exposures. *IIEE Trans Occup Ergonomics Hum Factors.* 2019;7(3–4):322–47. <https://doi.org/10.1080/24725838.2019.1684399>.
56. Ergonomics of human-system interaction—part 11: usability: definitions and concepts. Standard, International Organization for Standardization, Geneva, CH. 2018.
57. Scherer MJ, Sax C, Vanbiervliet A, Cushman LA, Scherer JV. Predictors of assistive technology use: the importance of personal and psychosocial factors. *Disabil Rehabil.* 2005;27(21):1321–31. <https://doi.org/10.1080/09638280500164800>.

58. Biddiss EA, Chau TT. Upper limb prosthesis use and abandonment: a survey of the last 25 years. *Prosthet Orthot Int.* 2007;31(3):236–57. <https://doi.org/10.1080/03093640600994581>.
59. van Ommeren AL, Smulders LC, Prange-Lasonder GB, Buurke JH, Veltink PH, Rietman JS. Assistive technology for the upper extremities after stroke: systematic review of users' needs. *JMIR Rehabil Assistive Technol.* 2018;5(2):10510. <https://doi.org/10.2196/10510>.
60. AT, Ramos VD, Alfieri FM, Battistella LR. Abandonment of assistive products: assessing abandonment levels and factors that impact on it. *Disabil Rehabil Assistive Technol.* 2018;13(7):716–23. <https://doi.org/10.1080/17483107.2018.1425748>.
61. SGS, Robinson I. Benefits of and barriers to involving users in medical device technology development and evaluation. *Int J Technol Assessment Health Care.* 2007;23(1):131–7. <https://doi.org/10.1017/S0266462307051677>.
62. Blanco T, Berbegal A, Blasco R, Casas R. Xassess: crossdisciplinary framework in user-centred design of assistive products. *J Eng Design.* 2016;27(9):636–64. <https://doi.org/10.1080/09544828.2016.1200717>.
63. Fischer B, Peine A, Östlund B. The importance of user involvement: a systematic review of involving older users in technology design. *Gerontologist.* 2020;60(7):513–23. <https://doi.org/10.1093/geront/gnz163>.
64. Stoumpos AI, Kitsios F, Talias MA. Digital Transformation in Healthcare: Technology Acceptance and Its Applications. *Int J Environ Res Public Health.* 2023 Feb 15;20(4):3407. doi: 10.3390/ijerph20043407.
65. Y. Liu et al., "Human-Centered Evaluation of EMG-Based Upper-Limb Prosthetic Control Modes," in *IEEE Transactions on Human-Machine Systems*, vol. 54, no. 3, pp. 271-281, June 2024, doi: 10.1109/THMS.2024.3381094.
66. Podury A, Jiam NT, Kim M, Donnenfield JI, Dhand A. Hearing and sociality: the implications of hearing loss on social life. *Front Neurosci.* 2023 Oct 3;17:1245434. doi: 10.3389/fnins.2023.1245434.
67. McDaid, D., Park, A. L., & Chadha, S. (2021). Estimating the global

costs of hearing loss. *International Journal of Audiology*, 60(3):162–170. <https://doi.org/10.1080/14992027.2021.1883197>

68. Haile LM, Orji AU, Reavis KM, Briant PS, Lucas KM, Alahdab F, Bärnighausen TW, Bell AW, Cao C, Dai X, Hay SI, Heidari G, Karaye IM, Miller TR, Mokdad AH, Mostafavi E, Natto ZS, Pawar S, Rana J, Seylani A, Singh JA, Wei J, Yang L, Ong KL, Steinmetz JD; GBD 2019 USA Hearing Loss Collaborators. Hearing Loss Prevalence, Years Lived With Disability, and Hearing Aid Use in the United States From 1990 to 2019: Findings From the Global Burden of Disease Study. *Ear Hear*. 2024 Jan-Feb 01;45(1):257-267. doi: 10.1097/AUD.0000000000001420.

69. Lin MJ, Chen CK. Breaking Sound Barriers: Exploring Tele-Audiology's Impact on Hearing Healthcare. *Diagnostics (Basel)*. 2024 Apr 22;14(8):856. doi: 10.3390/diagnostics14080856.

70. English K. Guidance on Providing Patient-Centered Care. *Semin Hear*. 2022 Jul 26;43(2):99-109. doi: 10.1055/s-0042-1748834.

71. Audiology Practice Standards Organization. S2.1: Hearing aid fitting standards for adult and geriatric patients. Published May, 2021. June 25. at:<https://www.audiologystandards.org/standards/display.php?id=102>

72. Orji A, Kamenov K, Dirac M, Davis A, Chadha S, Vos T. Global and regional needs, unmet needs and access to hearing aids. *Int J Audiol*. 2020 Mar;59(3):166-172. doi: 10.1080/14992027.2020.1721577.

73. Hahn R, Avraham KB. Gene Therapy for Inherited Hearing Loss: Updates and Remaining Challenges. *Audiol Res*. 2023 Dec 4;13(6):952-966. doi: 10.3390/audiolres13060083.

74. McKee M, James TG, Helm KVT, Marzolf B, Chung DH, Williams J, Zazove P. Reframing Our Health Care System for Patients With Hearing Loss. *J Speech Lang Hear Res*. 2022 Oct 17;65(10):3633-3645. doi: 10.1044/2022\_JSLHR-22-00052.

75. SignAloud solution <https://www.washington.edu/news/2016/04/12/uw-undergraduate-team-wins-10000-lemelson-mit-student-prize-for-gloves-that->

translate-sign-language/

76. BrightSign solution <https://www.brightsignglove.com/>

77. Sign-IO solution <https://fashnerd.com/2019/02/sign-io-breaking-the-language-barrier-between-those-who-are-deaf-and-those-who-arent%E2%80%8B/>

78. SpectraFlex Flex Sensor Datasheet. – [Электронный ресурс]. – Режим доступа: <https://www.spectrasymbol.com/resistive-flex-sensors/spectraflex-flex-sensors> – Дата звернення: 23.12.2024.

79. MPU-6050 Product Specification. – [Электронный ресурс]. – Режим доступа: <https://invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf> – Дата звернення: 23.12.2024.

80. Kameron Hussain, Frahaan Hussain. Mastering Visual Studio Code: Navigating the Future of Development, 1st Edition, 2024. – 232 с. – ISBN-13 979-8877664975.

81. Simon Monk. Programming Arduino: Getting Started with Sketches (Tab), 2nd Edition, Kindle Format, 2016. – 256 с. – ISBN-13 978-1259641633

82. Guido van Rossum, Fred L. Drake. Python 3 Reference Manual, 1st Edition, 2009. – 126 с. – ISBN-13 978-1441412690.

83. Aurélien Géron. Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, 3rd Edition, 2022. – 1104 с. – ISBN-13 978-1098125974.

84. Stefanie Molin. Hands-On Data Analysis with Pandas: Efficiently Perform Data Collection, Wrangling, Analysis, and Visualization Using Python, 1st Edition, 2019. – 740 с. – ISBN-13 978-1789615326.

85. Nicolas P. Rougier. Scientific Visualization: Python + Matplotlib, 1st Edition, 2021. – 248 с. – URL: <https://www.labri.fr/perso/nrougier/scientific-visualization.html>.