

Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерних наук
(повна назва)

Кафедра _____ програмної інженерії
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА Пояснювальна записка

рівень вищої освіти _____ другий (магістерський)

Дослідження алгоритмів розпізнавання рухів для мобільних застосунків,
призначених для відстеження фізичної активності користувачів.
(тема)

Виконав:
здобувач _____ 2 _____ року навчання
групи _____ ІІЗМ-23-2

_____ Володимир РОМАШОВ
(Власне ім'я, ПРІЗВИЩЕ)

Спеціальність _____ 121 – Інженерія програмного
забезпечення
(код і повна назва спеціальності)

Тип програми _____ освітньо-наукова

Керівник _____ доц. Наталія РУСАКОВА
(посада, Власне ім'я, ПРІЗВИЩЕ)

Допускається до захисту
Зав. кафедри

_____ Кирило СМЕЛЯКОВ
(підпис) (Власне ім'я, ПРІЗВИЩЕ)

2025 р.

Харківський національний університет радіоелектроніки

Факультет _____ комп'ютерних наук _____

Кафедра _____ програмної інженерії _____

Рівень вищої освіти _____ другий (магістерський) _____

Спеціальність _____ 121 – інженерія програмного забезпечення _____

Тип програми _____ освітньо-наукова _____

Освітня програма _____ інженерія програмного забезпечення _____

ЗАТВЕРДЖУЮ:

Зав. кафедри _____

(підпис)

«____» _____ 2025 р.

ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ

здобувачеві _____ Ромашову Володимирі Вадимовичу _____

(прізвище, ім'я, по батькові)

1. Тема роботи «Дослідження алгоритмів розпізнавання рухів для мобільних застосунків, призначених для відстеження фізичної активності користувачів»

Затверджена наказом по університету від 15.04. 2025р. № 290 Ст

2. Термін подання студентом роботи до екзаменаційної комісії 20.06.2025

3. Вихідні дані до проєкту: опис досліджуваних алгоритмів для розпізнавання рухів, Decision Trees, Random Forest, Convolutional Neural Networks, Hidden Markov Models та Support Vector Machines

4. Перелік питань, які потрібно опрацювати в роботі вступ, аналіз предметної галузі, огляд й аналіз літературних, наукових джерел, постановка задачі, теоретичне дослідження, висновки.

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Отримання завдання	16.04.2025	Виконано
2	Аналіз предметної галузі і постановка задачі	18.04.2025	Виконано
3	Визначення та аналіз додаткових критеріїв порівняння фреймворків, які впливають на користувацький досвід	22.04.2025	Виконано
4	Формалізація критеріїв	24.04.2025	Виконано
5	Проектування та розробка програмного забезпечення	30.04.2025	Виконано
6	Вимірювання продуктивності	02.05.2025	Виконано
7	Підготовка до апробації результатів дослідження. Публікація матеріалів	10.05.2025	Виконано
8	Підготовка пояснювальної записки	20.05.2025	Виконано
9	Підготовка презентації та доповіді	31.05.2025	Виконано
10	Перевірка на плагіат	08.06.2025	Виконано
11	Нормоконтроль	10.06.2025	Виконано
12	Рецензування	12.06.2025	Виконано
13	Попередній захист	14.06.2025	Виконано
14	Занесення диплома в електронний архів	16.06.2025	Виконано
15	Допуск до захисту у зав. кафедри	18.06.2025	Виконано

Дата видачі завдання 16 квітня 2025р.

Здобувач _____

(підпис)

Володимир РОМАШОВ

Керівник роботи _____

(підпис)

доц. Наталія РУСАКОВА

(посада, Власне ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ / ABSTRACT

Пояснювальна записка містить: 48 с., 1 рис., 6 табл., 14 джерел, 10 формул.

АНАЛІЗ, ДЕРЕВА РІШЕНЬ, ВИПАДКОВИЙ ЛІС, ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, ПРИХОВАНІ МАРКОВСЬКІ МОДЕЛІ, МЕТОДИ ОПОРНИХ ВЕКТОРІВ, МАШИННЕ НАВЧАННЯ, ЗАДАЧА ЛІНІЙНОГО ПРОГРАМУВАННЯ, ЗГОРТКОВА МОДЕЛЬ.

Об'єктом дослідження алгоритми розпізнавання рухів, що використовуються для аналізу даних із сенсорів у мобільних застосунках.

Метою роботи є оцінити та вибрати оптимальний алгоритм розпізнавання рухів для інтеграції в мобільні застосунки, враховуючи технічні, функціональні та користувацькі критерії.

Методами дослідження є формування системи критеріїв для оцінки їхньої ефективності (точність розпізнавання, швидкість роботи, енергоспоживання, адаптивність до нових типів рухів та масштабованість), використання згорткової моделі для інтегральної оцінки алгоритмів із врахуванням вагових коефіцієнтів, визначених методом аналізу ієрархій, а також розрахунок корисності альтернатив для порівняння алгоритмів за всіма критеріями. Такий підхід дозволив провести комплексний аналіз алгоритмів і визначити найефективніший варіант.

У результаті роботи було проведено оцінку п'яти алгоритмів розпізнавання рухів із використанням згорткової моделі, визначено найефективніший алгоритм для мобільних застосунків, спрямованих на відстеження фізичної активності.

ANALYSIS, DECISION TREES, RANDOM FOREST, CONVOLUTIONAL NEURAL NETWORKS, HIDDEN MARKOV MODELS, SUPPORT VECTOR MACHINES, MACHINE LEARNING, LINEAR PROGRAMMING PROBLEM, WRAPPING MODEL.

The object of research is motion recognition algorithms used to analyze data from sensors in mobile applications.

The aim of the work is to evaluate and select the optimal motion recognition algorithm for integration into mobile applications, taking into account technical, functional and user criteria.

The research methods are the formation of a system of criteria for evaluating their effectiveness (recognition accuracy, speed of operation, energy consumption, adaptability to new types of movements and scalability), the use of a convolutional model for the integral evaluation of algorithms taking into account the weighting factors determined by the method of analysis of hierarchies, as well as the calculation of the usefulness of alternatives to compare algorithms by all criteria. This approach made it possible to carry out a comprehensive analysis of algorithms and determine the most effective option.

As a result of the work, five motion recognition algorithms were evaluated using a convolutional model, and the most effective algorithm for mobile applications aimed at tracking physical activity was determined.

Завідувачу кафедри ПІ
проф. Кирилу СМЕЛЯКОВУ

ЗАЯВА

щодо самостійності виконання кваліфікаційної роботи та можливості її публікації (та/або публікації анотації кваліфікаційної роботи) в електронному архіві відкритого доступу EIAr KhNURE

Я, Ромашов Володимир Вадимович, здобувач вищої освіти на другому (магістерському) рівні вищої освіти академічної групи ПЗм-23-2, кафедри «Програмна інженерія», заявляю: моя кваліфікаційна робота на тему «Дослідження алгоритмів розпізнавання рухів для мобільних застосунків, призначених для відстеження фізичної активності користувачів», що буде представлена в екзаменаційну комісію для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в репозиторії "EIArKhNURE". Погоджуюся з авторським договором, відповідно до Положення про репозиторій ХНУРЕ "EIArKhNURE". Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений з вимогами академічної доброчесності, згідно з якими виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

19.06.2015

Володимир РОМАШОВ

ЗМІСТ

Вступ.....	9
1. Аналіз предметної галузі	11
1.1 Тенденції та перспективи.....	11
1.2 Огляд існуючих підходів	12
1.3 Обмеження існуючих підходів	13
1.4 Визначення рівня інноваційності	14
2. Огляд та аналіз літературних джерел.....	15
2.1 Огляд основних джерел	15
2.1.1 Теоретична база для моделювання часових і просторових залежностей	15
2.1.2 Ефективність комбінованих методів у задачах аналізу руху	15
2.1.3 Порівняння алгоритмів для класифікаційних задач.....	16
2.2 Оцінка актуальності та новизни	16
3. Постановка задачі.....	17
3.1 Визначення тематики та кінцевих результатів	17
3.2 Обґрунтування вибору методів дослідження	17
3.3 Обмеження дослідження.....	19
4. Опис теоретичного дослідження	21
4.1.1 Decision Trees	21
4.1.2 Random Forest.....	23
4.1.3 Convolutional Neural Networks (CNN).....	23
4.1.4 Hidden Markov Models (HMM).....	24
4.1.5 Support Vector Machines (SVM).....	25
4.2. Критерії оцінки моделей	26
4.2.1 Точність розпізнавання	27
4.2.2 Швидкість роботи	27
4.2.3 Енергоспоживання	28
4.2.4 Адаптивність до нових типів рухів.....	28
4.2.5 Масштабованість	29
4.3 Обґрунтування вибору загорткової моделі.....	29
4.4 Опис задачі багатокритеріального вибору.....	31

4.3.1 Векторний опис альтернатив за критеріями	31
4.3.2 Перетворення векторного опису	32
4.3.3 Альтернатив принципу Парето	33
4.3.4 Опис загорткової моделі	33
4.3.5 Розрахунки корисності	34
5 Опис експеримента дослідження.....	36
5.1 Опис експерименту	36
5.2 Технології та програмні інструменти.....	36
5.3 Підхід до розробки	37
5.4 Результати експерименту.....	42
Висновки.....	44
Перелік джерел посилання	46
Перелік джерел посилання за науковими напрямками керівника та науковців кафедри програмної інженерії.....	48

ВСТУП

У сучасному світі мобільні технології стають невід'ємною частиною життя мільйонів людей. Одним із найбільш динамічно зростаючих напрямів є мобільні застосунки для відстеження фізичної активності, які дозволяють користувачам аналізувати свій стан здоров'я, контролювати фізичні навантаження та досягати поставлених цілей. Ці застосунки знаходять застосування не лише серед любителів спорту, але й у медичних програмах реабілітації, що підкреслює їх важливість у сучасному суспільстві.

Основою роботи таких застосунків є алгоритми розпізнавання рухів, які аналізують дані зі спеціалізованих сенсорів, вбудованих у смартфони, фітнес-трекери або смарт-годинники. Зокрема, акселерометри, гіроскопи та магнітометри забезпечують високоточний збір даних, але ефективність цих алгоритмів залежить від їх здатності правильно інтерпретувати різні рухи користувача, враховуючи можливі перешкоди, такі як шум сенсорів або індивідуальні особливості рухів.

Дослідження у сфері алгоритмів розпізнавання рухів активно розвиваються, оскільки ця технологія має широкий спектр застосувань: від простих програм для відстеження кроків до комплексних систем для розпізнавання видів фізичної активності, аналізу постави та оцінки ефективності тренувань. Завдяки використанню сучасних підходів, таких як машинне навчання та аналіз великих даних, точність цих алгоритмів постійно вдосконалюється.

Актуальність дослідження обумовлена тим, що надійне розпізнавання рухів є критично важливим для створення високоякісних застосунків, які відповідають очікуванням користувачів і сприяють популяризації активного способу життя. У цьому контексті постає необхідність глибокого аналізу існуючих алгоритмів, вивчення їх переваг і недоліків, а також розробки нових підходів, які могли б забезпечити вищу точність, надійність і енергоефективність роботи.

Метою цього звіту є дослідження алгоритмів розпізнавання рухів, які застосовуються у мобільних застосунках для відстеження фізичної активності користувачів, з метою підвищення їхньої точності та ефективності. У роботі

передбачається аналіз існуючих підходів, тестування обраних алгоритмів на експериментальних даних, а також розробка рекомендацій для їх удосконалення.

Цей проєкт спрямований на вирішення актуальних проблем, пов'язаних із використанням алгоритмів розпізнавання рухів, і має потенціал стати вагомим внеском у розвиток технологій відстеження фізичної активності.

1. АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

Мобільні застосування для відстеження фізичної активності стають важливим інструментом у підтримці здорового способу життя та оптимізації тренувального процесу. Ці рішення базуються на сучасних технологіях обробки даних, що дозволяє користувачам відстежувати різноманітні аспекти своєї фізичної активності: кроки, відстань, витрати калорій, якість сну тощо. Особливу роль у цьому відіграють алгоритми розпізнавання рухів, які забезпечують точний аналіз даних, отриманих зі смартфонів, смарт-годинників та фітнес-трекерів.

У цьому розділі буде розглянуто основні тенденції розвитку технологій відстеження фізичної активності, а також перспективи їхнього використання для покращення якості життя користувачів.

1.1 Тенденції та перспективи

Мобільні додатки для відстеження фізичної активності зазнають значних змін під впливом новітніх технологій та зростаючих потреб користувачів. Серед основних тенденцій розвитку цієї галузі можна виділити наступні:

- використання комп'ютерного зору: сучасні додатки починають інтегрувати технології комп'ютерного зору для аналізу рухів користувачів. Це дозволяє забезпечити точніше відстеження та аналіз фізичної активності, що сприяє підвищенню ефективності тренувань та зниженню ризику травм [1];
- покращення точності носимих пристроїв: виробники фітнес-трекерів та смарт-годинників постійно вдосконалюють свої продукти, забезпечуючи більш точне вимірювання показників, таких як кількість кроків, якість сну та частота серцевих скорочень [2];
- розширення функціональності смарт-годинників: сучасні смарт-годинники пропонують широкий спектр функцій, включаючи контроль фізичної активності, вибір різних режимів спортивних тренувань, спостереження за якістю сну, підрахунок спалених калорій та моніторинг

медичних показників, таких як вимірювання тиску та рівня кисню в крові [3];

- інтеграція штучного інтелекту (ШІ): використання ШІ у фітнес-додатках дозволяє персоналізувати тренувальні програми, аналізувати техніку виконання вправ та надавати рекомендації щодо покращення [4].

Перспективи розвитку мобільних додатків для відстеження фізичної активності включають подальшу інтеграцію з медичними системами для моніторингу здоров'я, використання великих даних для аналізу та прогнозування фізичної активності, а також розвиток технологій доповненої реальності для покращення користувацького досвіду.

1.2 Огляд існуючих підходів

Розпізнавання рухів у мобільних застосунках здійснюється за допомогою різних підходів, які можна умовно поділити на традиційні методи аналізу сигналів та методи машинного навчання. Кожен із цих підходів має свої особливості, переваги та недоліки залежно від специфіки застосування, апаратних обмежень і обчислювальних можливостей.

Перший із них, традиційні методи аналізу сигналів, базується на обробці даних із сенсорів, таких як акселерометри та гіроскопи. Основний інструментарій включає обчислення базових характеристик сигналів, таких як середнє значення, дисперсія чи спектральний аналіз. Завдяки цим методам можна з достатньою точністю визначати прості види руху, наприклад, ходьбу або біг. Однак, їх ефективність значно знижується у випадках складніших патернів рухів, які потребують урахування більш комплексних особливостей сигналу [5].

Другий підхід, методи машинного навчання, дозволяє досягти вищої точності та гнучкості у розпізнаванні рухів завдяки здатності алгоритмів виявляти приховані залежності в даних.

Застосування цих алгоритмів відкриває нові можливості для обробки складних і багатовимірних сигналів, що надходять від сенсорів, забезпечуючи якісніше розпізнавання навіть складних фізичних активностей.

1.3 Обмеження існуючих підходів

Попри значний прогрес у розробці алгоритмів розпізнавання рухів, існуючі підходи мають низку обмежень, які впливають на їхню точність, ефективність та практичне застосування.

Традиційні методи аналізу сигналів демонструють низьку здатність до розпізнавання складних рухів, оскільки вони базуються на обробці базових характеристик сигналів. Ці методи добре працюють із простими та повторюваними типами активності, такими як ходьба чи біг, але виявляються неефективними для визначення складних рухів із багатьма компонентами чи нерегулярними патернами крім того, традиційні підходи мають високу чутливість до шуму [6]. Дані з акселерометрів та гіроскопів можуть спотворюватися через фізичні перешкоди, як-от тремтіння чи удари, що значно впливає на точність аналізу. Ще однією проблемою є недостатня адаптивність, адже ці методи часто розробляються для роботи зі стандартними моделями поведінки і не враховують індивідуальних відмінностей користувачів, таких як особливості стилю ходьби, ваги чи зросту [7].

Алгоритми машинного навчання, хоч і забезпечують вищу гнучкість і точність, також мають свої недоліки. Вони потребують значних обчислювальних ресурсів, що може бути проблемою для мобільних пристроїв із низькою продуктивністю чи обмеженим енергоспоживанням [8]. Крім того, такі алгоритми вимагають великого обсягу якісних даних для навчання, однак збір таких даних часто ускладнюється через конфіденційність або етичні обмеження. Ще однією важливою проблемою є різні рівні точності моделей у різних умовах. Наприклад, алгоритм, навчений на даних здорових людей, може працювати неточно для людей із порушеннями руху чи фізичними обмеженнями [9].

Чутливість до змін у середовищі використання також є суттєвим викликом. Зміна умов, наприклад, носіння пристрою на іншій частині тіла або виконання унікальних рухів, може значно знизити точність розпізнавання.

Серед інших обмежень слід зазначити високий рівень енергоспоживання сучасних алгоритмів, особливо тих, що працюють у режимі реального часу, що

негативно впливає на тривалість роботи мобільних пристроїв від батареї. Крім того, інтеграція складних алгоритмів у мобільні застосунки потребує узгодження між програмним забезпеченням та апаратною частиною пристрою, що може бути складним завданням, враховуючи різноманітність апаратних платформ.

Розв'язання цих обмежень потребує впровадження нових підходів, які забезпечуватимуть оптимальний баланс між точністю, продуктивністю та енергоефективністю. Одним із перспективних напрямів є розробка гібридних методів, що поєднують переваги традиційного аналізу сигналів та сучасних адаптивних алгоритмів машинного навчання.

1.4 Визначення рівня інноваційності

Інноваційність алгоритмів розпізнавання рухів у мобільних застосунках для моніторингу фізичної активності залежить від використання сучасних технологій, їхньої унікальності та здатності вирішувати актуальні проблеми.

Серед ключових аспектів виділяється застосування передових технологій, таких як глибокі нейронні мережі (CNN, RNN) для розпізнавання складних патернів рухів і обробки багатовимірних сигналів навіть за наявності шуму. Використання гібридних моделей, що поєднують традиційні методи з машинним навчанням, забезпечує високу точність і енергоефективність, а впровадження розподілених обчислень (хмарних або edge computing) знижує навантаження на пристрої.

Важливим елементом є персоналізація, яка дозволяє налаштовувати алгоритми для конкретних користувачів, включно зі спортсменами, людьми похилого віку чи пацієнтами з фізичними обмеженнями. Інтеграція з іншими технологіями, такими як доповнена реальність (AR), медичні системи та IoT, розширює можливості цих рішень.

Рівень інноваційності цих технологій можна оцінити як високий, оскільки вони мають значний вплив на різні сфери та сприяють підвищенню якості життя користувачів, водночас зберігаючи потенціал для вдосконалення.

2. ОГЛЯД ТА АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

2.1 Огляд основних джерел

У цьому підрозділі розглянемо три ключові джерела:

- НММ та CNN для класифікації тексту.
- Random Forest та НММ для розпізнавання транспорту.
- Порівняння НММ та Random Forest у класифікації даних.

Вони будуть використані для дослідження алгоритмів розпізнавання рухів у мобільних застосунках. Обрані джерела охоплюють різні підходи та методи, що дозволяє отримати всебічне уявлення про сучасний стан цієї галузі.

2.1.1 Теоретична база для моделювання часових і просторових залежностей

У "Method of Combining Hidden Markov Model and Convolutional Neural Network for Text Classification" досліджується поєднання прихованих марковських моделей та згорткових нейронних мереж для класифікації тексту. Хоча основна увага приділяється текстовим даним, запропонований підхід може бути адаптований для розпізнавання рухів, оскільки обидва завдання мають справу з послідовностями даних. Поєднання НММ та CNN дозволяє ефективно моделювати часові залежності та просторові особливості, що є критичним для точного розпізнавання складних рухів [10].

2.1.2 Ефективність комбінованих методів у задачах аналізу руху

"Random Forest/Hidden Markov Model for Transportation Mode Recognition Using Smartphone Sensors" демонструє використання алгоритмів Random Forest та НММ для розпізнавання режимів транспорту за допомогою сенсорів смартфона. Результати показують, що комбінація цих методів забезпечує високу точність класифікації, що свідчить про їхню ефективність у задачах, пов'язаних з аналізом руху. Цей підхід може бути застосований для розпізнавання різних типів фізичної активності користувачів мобільних застосунків [11].

2.1.3 Порівняння алгоритмів для класифікаційних задач

Стаття "A Performance Study of Hidden Markov Model and Random Forest in Internet Traffic Classification" акцентує увагу на порівняльний аналіз продуктивності НММ та Random Forest у задачах класифікації інтернет-трафіку. Незважаючи на те, що дослідження зосереджене на мережевих даних, отримані висновки щодо сильних та слабких сторін кожного алгоритму можуть бути корисними при їх застосуванні до розпізнавання рухів. Зокрема, стаття надає цінну інформацію про точність та ефективність цих методів у різних умовах [12].

2.2 Оцінка актуальності та новизни

Актуальність алгоритмів розпізнавання рухів зумовлена потребою у високій точності класифікації складних типів активності, адаптації до різних платформ і мінімізації енергоспоживання. Обрані джерела демонструють новизну підходів через використання гібридних моделей, інтеграцію глибокого навчання та аналіз реальних сценаріїв. Зокрема, поєднання НММ із Random Forest і CNN дозволяє досягати високої точності завдяки моделюванню складних залежностей, тоді як підходи, що залучають глибоке навчання, забезпечують якісний аналіз багатовимірних даних.

Висновки з аналізованих джерел свідчать про перспективність їх впровадження в мобільні додатки, орієнтовані на точність, енергоефективність та адаптивність до нових типів рухів..

3. ПОСТАНОВКА ЗАДАЧІ

3.1 Визначення тематики та кінцевих результатів

Тематика цього проєкту спрямована на дослідження алгоритмів розпізнавання рухів для інтеграції в мобільні застосунки, орієнтовані на відстеження фізичної активності користувачів. У рамках роботи буде проведено оцінку технічних, функціональних та користувацьких критеріїв для визначення оптимального підходу.

Результатом роботи стане обґрунтований вибір алгоритму, який відповідатиме потребам сучасних мобільних застосунків, забезпечуючи високу точність розпізнавання рухів, енергоефективність, швидкість роботи та адаптивність до нових типів рухів. У дослідженні застосовуватиметься Задача Лінійного Програмування (ЗЛП) для багатокритеріального аналізу ефективності обраних алгоритмів, що дозволить визначити найкраще рішення з урахуванням усіх ключових вимог.

3.2 Обґрунтування вибору методів дослідження

Для досягнення мети дослідження обґрунтовано вибір Задачі Лінійного Програмування (ЗЛП) як основного методу для аналізу ефективності алгоритмів розпізнавання рухів. Цей метод є універсальним підходом, що дозволяє формалізувати задачу оцінки у вигляді математичної моделі, де цільова функція враховує різні критерії ефективності. Завдяки здатності ЗЛП працювати з багатьма критеріями одночасно, цей метод забезпечує комплексну оцінку таких характеристик алгоритмів, як точність, швидкість роботи, енергоспоживання, адаптивність та масштабованість. Окрім цього, ЗЛП дозволяє оптимально розподілити ресурси, зменшуючи обчислювальні витрати, що є ключовим у мобільних застосунках, де обчислювальні можливості обмежені.

Вибір алгоритмів для аналізу зумовлений їхньою ефективністю у задачах розпізнавання рухів і здатністю задовольняти різні критерії ефективності:

- Decision Trees обрано через простоту і швидкість роботи. Цей алгоритм забезпечує інтуїтивно зрозумілий підхід до класифікації, що є важливим для швидкого прототипування рішень. Однак, він може бути обмеженим у випадках складних залежностей між даними;
- Random Forest було включено до аналізу завдяки його здатності використовувати ансамблевий підхід. Цей алгоритм не лише підвищує точність класифікації, але й є стійким до шуму, що особливо важливо при роботі з мобільними сенсорами, дані яких можуть бути нерівномірними;
- Convolutional Neural Networks (CNN) обрано через високу ефективність у складних завданнях аналізу даних. Цей алгоритм має здатність виділяти ознаки з високою деталізацією, що робить його незамінним у контексті розпізнавання складних рухів. Однак, CNN вимагає значних обчислювальних ресурсів, тому його використання виправдане для пристроїв із високою продуктивністю;
- Hidden Markov Models (HMM) підходить для аналізу часових залежностей, що характерні для послідовностей рухів. Цей алгоритм ефективно моделює динаміку рухів, дозволяючи враховувати їхній контекст;
- Support Vector Machines (SVM) було обрано через здатність працювати у високовимірних просторах та створювати нелінійні межі між класами; SVM показує високу точність на невеликих наборах даних, що робить його придатним для швидкого тестування.

Для забезпечення комплексного підходу до оцінки алгоритмів, у дослідженні буде використаний якісний тест, який включатиме створення тестового набору даних, максимально наближеного до реальних умов використання мобільного застосунку.

Це дозволить не лише оцінити ефективність алгоритмів у лабораторних умовах, але й перевірити їхню роботу у сценаріях, що відповідають практичному використанню. Тестовий набір даних включатиме різноманітні типи рухів, зібрані за допомогою сенсорів мобільних пристроїв, а також враховуватиме можливі шуми та артефакти, що виникають у реальному середовищі.

Оцінка алгоритмів буде проводитись за такими критеріями:

- точність розпізнавання: Цей критерій є ключовим, оскільки від нього залежить правильність класифікації рухів.
- швидкість роботи: Здатність алгоритму обробляти дані в реальному часі є критичною для мобільних застосунків, де затримка може знижувати користувацький досвід.
- енергоспоживання: З огляду на обмежений ресурс батареї мобільних пристроїв, мінімізація енергоспоживання є важливим параметром.
- адаптивність до нових типів рухів: Цей критерій демонструє здатність алгоритму адаптуватись до змін у поведінці користувача.
- масштабованість: Оцінюється ефективність алгоритму на пристроях із різним рівнем апаратних можливостей.

Застосування якісного тесту у поєднанні з вищевказаними критеріями дозволить отримати всебічну оцінку кожного алгоритму та виявити найкраще рішення для використання у мобільних застосунках.

3.3 Обмеження дослідження

У межах даного дослідження існують певні обмеження, які впливають на його результати та загальні висновки. Одним із ключових обмежень є вибіркова база даних для навчання та тестування алгоритмів. Оскільки дані для тренування алгоритмів розпізнавання рухів збираються з використанням сенсорів мобільних пристроїв, вони можуть бути обмежені за різноманітністю типів рухів, поведінкових моделей користувачів та умов збирання. Це може впливати на загальну узагальненість результатів для інших сценаріїв використання.

Іншим важливим фактором є обчислювальні можливості мобільних пристроїв. Алгоритми, такі як згорткові нейронні мережі (CNN), потребують значних апаратних ресурсів для обробки даних у реальному часі, що може бути проблематичним для пристроїв зі слабкою апаратною базою. У цьому контексті результати дослідження можуть бути непридатними для малопотужних мобільних пристроїв або таких, що мають обмежений час роботи від батареї.

Також слід зазначити, що алгоритми, які розглядаються в дослідженні, розроблені для роботи за стандартних умов. Зміни в умовах зовнішнього середовища, такі як температура, вологість чи наявність шуму в сенсорних даних, можуть призводити до зниження точності роботи алгоритмів. Крім того, певні рухи або активності користувачів можуть бути важкими для класифікації через їхню складну динаміку або схожість із іншими рухами.

Ще одним обмеженням є відсутність врахування індивідуальних фізичних особливостей користувачів. Наприклад, різні статура, вік або стиль рухів можуть впливати на якість роботи алгоритмів. Наразі дослідження не враховує ці особливості через складність масштабування моделі для широкого діапазону користувачів.

Окремо варто згадати вплив вимог до енергоефективності. Хоча дослідження охоплює цей критерій, досягнення компромісу між точністю алгоритмів та енергоспоживанням залишається викликом, особливо у випадках використання складних моделей, які забезпечують високу точність.

Таким чином, результати дослідження потребують обережного тлумачення та адаптації до конкретних умов застосування. Для подальших досліджень рекомендується розширити базу даних, враховувати індивідуальні особливості користувачів і проводити тести за різних умов експлуатації мобільних пристроїв.

4. ОПИС ТЕОРЕТИЧНОГО ДОСЛІДЖЕННЯ

4.1 Огляд основних підходів

У цьому розділі буде розглянуто п'ять основних підходів до розпізнавання рухів: Decision Trees, Random Forest, Convolutional Neural Networks, Hidden Markov Models та Support Vector Machines.

Вибір саме цих алгоритмів зумовлений їхньою ефективністю та адаптивністю у задачах аналізу фізичної активності користувачів, а також здатністю враховувати багатокритеріальні аспекти таких задач, як точність класифікації, енергоспоживання, швидкість обробки та масштабованість. Кожен із цих підходів базується на різних принципах, що забезпечує їхній внесок у дослідження різних аспектів ефективності алгоритмів.

4.1.1 Decision Trees

Один із найпопулярніших алгоритмів машинного навчання, що базується на побудові деревоподібної моделі для прийняття рішень. Алгоритм працює за принципом розділення даних на основі певних правил або умов, представлених у вузлах дерева. Кожен вузол відповідає за перевірку певної ознаки, а кожна гілка дерева — за результат перевірки. В кінцевих вузлах (листяках) визначається клас або значення, яке відповідає цьому шляху.

На кожному етапі розділення обирається ознака, яка мінімізує критерій нечіткості (наприклад, ентропія або індекс Джині):

Ентропія:

$$H(D) = - \sum_{i=1}^k p_i \log_2(p_i) \quad (4.1)$$

де $H(D)$ – ентропія набору даних D ;

p_i – ймовірність належності об'єкта до класу i .

Індекс Джині:

$$G(D) = 1 - \sum_{i=1}^k p_i^2 \quad (4.2)$$

Ці критерії визначають, наскільки добре обраний параметр розділяє дані на підгрупи.

Переваги Decision Trees

- простота реалізації та візуалізації;
- інтуїтивно зрозумілий процес прийняття рішень;
- швидкість навчання та передбачення на невеликих наборах даних;
- не потребує масштабування ознак (наприклад, нормалізації чи стандартизації);
- добре працює із категорійними та числовими даними.

Недоліки

- схильність до перенавчання (overfitting) на невеликих наборах даних;
- менша точність у порівнянні з ансамблевими методами, такими як Random Forest;
- чутливість до змін у даних;
- ігнорування можливих залежностей між ознаками, якщо вони не враховані на етапі розділенн.

Алгоритм Decision Trees може бути застосований для класифікації типів рухів, наприклад, розпізнавання ходьби, бігу або стрибків на основі даних із сенсорів мобільного пристрою (акселерометра або гіроскопа). Для кожного руху алгоритм аналізує характеристики, такі як прискорення, зміни кута нахилу або періодичність рухів, і на основі цих даних визначає клас.

Decision Trees є ефективним рішенням для завдань, що потребують швидкого прийняття рішень і пояснюваних моделей, але можуть бути менш точними для складних даних із великою кількістю шумів.

4.1.2 Random Forest

Ансамблевий метод машинного навчання, що базується на поєднанні великої кількості дерев рішень. Алгоритм використовує принцип "мудрості натовпу": кожне дерево рішень у лісі голосує за певний клас, а фінальне рішення приймається шляхом голосування більшості. Для створення різноманітних дерев використовуються випадкові вибірки даних (метод бутстрепінгу) та підмножини ознак.

Кінцеве передбачення в Random Forest визначається як середнє (для задач регресії) або найпоширеніший клас (для задач класифікації) серед усіх дерев:

$$\hat{y} = \text{Mode}(T_1(x), T_2(x), \dots, T_n(x)) \quad (4.3)$$

де $T_i(x)$ – прогноз i -го дерева для зразка x .

Переваги Random Forest

- висока точність;
- стійкість до шуму;
- універсальність;
- менша схильність до перенавчання.

Недоліки

- обчислювальна складність;
- втрата інтерпретованості.

Random Forest підходить для розпізнавання фізичної активності, оскільки здатен працювати з великою кількістю ознак, таких як зміни прискорення, кут нахилу та частота рухів. Наприклад, для розпізнавання бігу чи ходьби алгоритм може використовувати сукупність даних із сенсорів, забезпечуючи точну класифікацію навіть за наявності шуму.

4.1.3 Convolutional Neural Networks (CNN)

Спеціалізовані нейронні мережі, які здебільшого використовуються для обробки зображень і відео, але також ефективні в інших типах даних, таких як аудіо

та текст. Основною ідеєю є автоматичне виявлення ознак через застосування операцій згортки (convolution).

Операція згортка:

$$f(x, y) = \sum_{i, j} I(i, j)K(x - i, y - j) \quad (4.4)$$

де: $I(i, j)$ – піксельні значення вхідного зображення;

$K(x-i, y-j)$ – фільтр або ядро, яке ковзає по зображенню.

Переваги CNN

- автоматичне виявлення ознак;
- параметрична ефективність;
- CNN здатні навчатися на різних рівнях складності від простих до складних ознак.

Недоліки

- висока обчислювальна складність
- неінтерпретованість.

У мобільному додатку для відстеження фізичної активності, дані про рухи з акселерометра можуть бути передані до моделі CNN, яка аналізує їх і класифікує тип руху. Наприклад, якщо користувач рухається швидко, модель може визначити, що це біг, і на основі цього надати відповідні рекомендації чи статистику про тренування

4.1.4 Hidden Markov Models (HMM)

Статистичний метод, який використовується для моделювання систем, де спостережувані дані залежні від невидимих (скритих) станів. Вони особливо корисні для задач, де важливу роль відіграє часовий аспект, наприклад, для розпізнавання послідовностей чи прогнозування. Такий підхід дає змогу моделювати процеси, які мають внутрішню структуру, але їх стани не можна безпосередньо спостерігати. Наприклад, у задачах розпізнавання рухів або аналізу мови, де поведінка людини чи система може бути представлена як серія скритих

станів (ходьба, біг, сидіння), а спостережувані дані (акселерометричні або гіроскопічні вимірювання) залежатимуть від цих станів.

Модель (НММ):

$$P(O | \lambda) = \sum_S P(O, S | \lambda), \quad (4.5)$$

де: λ – піксельні значення вхідного зображення.

Переваги НММ

- моделювання часових залежностей;
- гнучкість;
- вбудовані алгоритми для навчання та передбачення.

Недоліки

- обмеження на кількість станів;
- якщо моделі не вистачає даних для навчання або дані не відображають усіх важливих аспектів процесу, прогнози можуть бути неточними;
- висока чутливість до початкових умов.

Під час тренування НММ можна використовувати дані про різні рухи (ходьба, біг, стояння) для визначення ймовірностей переходів між станами та ймовірностей спостережень для кожного стану. При отриманні нових даних сенсорів, модель може оцінити ймовірні стани і класифікувати поточну активність.

4.1.5 Support Vector Machines (SVM)

Алгоритм машинного навчання для класифікації та регресії, який знаходить оптимальну гіперплощину, що розділяє різні класи з максимальною маржею. Це особливо ефективно для лінійно неочевидних задач, де необхідно використовувати ядра для трансформації даних у більш високий вимір.

Лінійний SVM: Задача полягає в пошуку гіперплощини $w \cdot x + b = 0$, яка максимізує відстань між двома класами.

Мінімізуємо функцію:

$$\min_{w,b} \frac{1}{2} \|w\|^2 \quad (4.6)$$

за умови, що для кожної точки i виконується нерівність:

$$y_i(w \cdot x_i + b) \geq 1 \quad (4.7)$$

де: x_i – це вектор ознак;

y_i – мітка класу (1 або -1).

Переваги SVM

- Мінімізація помилок;
- Ефективність на малих вибірках;
- Гнучкість.

Недоліки

- Обчислювальна складність;
- Параметри ядра;
- Погана масштабованість.

Може бути використаний для класифікації різних типів рухів, таких як ходьба, біг, стрибки тощо, на основі даних акселерометра та гіроскопа.

4.2. Критерії оцінки моделей

Критерії оцінки моделей мають вирішальне значення для визначення ефективності алгоритмів машинного навчання в задачах розпізнавання рухів користувачів. У дослідженні розглядаються основні метрики, які дозволяють оцінити не лише точність, але й зручність використання та адаптацію алгоритмів на мобільних пристроях.

Алгоритми повинні задовольняти вимоги щодо точності, швидкості, енергоефективності та масштабованості, щоб забезпечити комфорт і ефективність для користувачів.

У рамках дослідження для задачі багатокритеріального вибору було розроблено кілька шкал для оцінки різних алгоритмів розпізнавання рухів. Кожна з цих шкал дозволяє комплексно оцінити модель за кількома критеріями, що є

важливими для ефективної роботи мобільних застосунків для відстеження фізичної активності.

Задача багатокритеріального вибору полягає в необхідності оцінити кожен алгоритм за різними факторами, де кожен критерій має свою шкалу оцінки. Це дозволяє приймати збалансовані рішення з урахуванням всіх важливих аспектів

4.2.1 Точність розпізнавання

Точність є однією з найпростіших і найбільш часто використовуваних метрик для оцінки класифікації. Вона показує, яка частка всіх передбачень моделі є правильними. Оцінка точності дозволяє зрозуміти, наскільки добре алгоритм виконує своє завдання при класифікації рухів користувача.

Шкала оцінки точності:

- 0-50% (1 бал) – дуже низька точність, алгоритм часто помиляється;
- 51-70% (2 бали) – низька точність, помилки все ще поширені;
- 71-85% (3 бали) – середня точність, прийнятна для базових застосунків;
- 86-95% (4 бали) – висока точність, рідкісні помилки;
- 96-100% (5 балів) – дуже висока точність, практично ідеальний результат.

Точність є хорошою метрикою для збалансованих наборів даних, де кількість позитивних та негативних прикладів є приблизно рівною. Однак при дисбалансі класів ця метрика може бути оманливою, адже висока точність може досягатися шляхом простого ігнорування рідкісного класу. Тому, для оцінки моделей на таких даних, слід використовувати додаткові метрики, такі як повнота чи F1-міра.

4.2.2 Швидкість роботи

Швидкість роботи є важливим критерієм для оцінки ефективності алгоритмів, особливо у реальному часі, де мінімальні затримки можуть значно покращити користувацький досвід. Цей критерій оцінює час, необхідний для обробки одного фрагмента даних (затримка) і дозволяє зрозуміти, наскільки швидко алгоритм реагує на вхідні дані.

Шкала оцінки швидкості роботи:

- >500 мс (1 бал) – дуже повільна обробка, не придатна для реального часу;
- 300-500 мс (2 бали) – повільна обробка, затримки відчутні;
- 200-300 мс (3 бали) – середня швидкість, помітно, але прийнятно;
- 100-200 мс (4 бали) – висока швидкість, майже миттєва;
- <100 мс (5 балів) – дуже висока швидкість, без затримок.

Ця шкала допомагає оцінити, наскільки алгоритм здатний обробляти дані в режимі реального часу, що є критично важливим для мобільних застосунків, де користувач очікує швидку реакцію системи..

4.2.3 Енергоспоживання

Енергоспоживання оцінює, наскільки алгоритм впливає на заряд батареї мобільного пристрою. У реальних умовах, де мобільні пристрої обмежені в енергетичних ресурсах, важливо, щоб алгоритм не виснажував батарею занадто швидко, зберігаючи баланс між ефективністю та енергоефективністю.

- 1 бал – дуже високе споживання, батарея розряджається дуже швидко.
- 2 бали – високе споживання, помітний вплив на тривалість роботи пристрою.
- 3 бали – середнє споживання, допустиме для використання.
- 4 бали – низьке споживання, незначно впливає на батарею.
- 5 балів – дуже низьке споживання, практично непомітний вплив.

Ця шкала дозволяє оцінити, наскільки оптимізовані алгоритми для роботи з обмеженими енергетичними ресурсами, що критично для мобільних застосунків.

4.2.4 Адаптивність до нових типів рухів

Адаптивність до нових типів рухів оцінює, наскільки алгоритм можна легко модифікувати або розширювати для підтримки нових типів рухів, без значних витрат часу чи ресурсів. Це важливий критерій для застосунків, які повинні підтримувати різні типи фізичної активності та адаптуватися до змінюваних умов.

Шкала оцінки адаптивності до нових типів рухів:

- 1 бал – немає можливості додавати нові типи рухів.
- 2 бали – додавання нових рухів вимагає значної перебудови алгоритму.
- 3 бали – можливе додавання, але з високими витратами часу та ресурсів.
- 4 бали – додавання нових рухів можливе із середніми витратами.
- 5 балів – просте та швидке додавання нових типів рухів.

Ця шкала дозволяє оцінити, наскільки гнучким є алгоритм в плані розширення функціоналу для підтримки нових рухів, що критично важливо для забезпечення масштабованості і адаптації системи до нових вимог користувачів.

4.2.5 Масштабованість

Масштабованість оцінює, наскільки алгоритм здатний ефективно працювати на різних пристроях, від простих до більш потужних. Це важливий критерій, особливо для мобільних застосунків, де пристрої можуть сильно відрізнятися за характеристиками, і алгоритм повинен бути здатним адаптуватися до різних умов.

Шкала оцінки масштабованості:

- 1 бал – працює лише на вузькому наборі пристроїв із високими характеристиками.
- 2 бали – потребує значної адаптації для різних пристроїв.
- 3 бали – працює на багатьох пристроях, але потребує оптимізації.
- 4 бали – легко переноситься на більшість пристроїв.
- 5 балів – ідеально підходить для всіх типів пристроїв, мінімальні вимоги.

Ця шкала дозволяє оцінити, наскільки добре алгоритм може масштабуватися для різних рівнів апаратного забезпечення, забезпечуючи стабільну роботу без зайвих витрат на оптимізацію чи адаптацію.

4.3 Обґрунтування вибору загорткової моделі

Для вирішення задачі вибору оптимального алгоритму для мобільних застосунків з відстеження фізичної активності було обрано загорткову модель з

ваговими коефіцієнтами. Це рішення ґрунтується на кількох ключових аспектах, які відповідають вимогам і характеристикам поставленої задачі:

- багатокритеріальність задачі: задача включає оцінку альтернатив за п'ятьма критеріями: точність розпізнавання, швидкість роботи, енергоспоживання, адаптивність до нових типів рухів і масштабованість. Кожен із цих критеріїв має різну важливість залежно від контексту використання мобільного застосунку. Наприклад, для застосунків, що працюють у реальному часі, критичним є критерій швидкості, а для застосунків з довготривалим використанням на одному заряді — енергоспоживання і точність. Загорткова модель дозволяє коректно врахувати ці відмінності і інтегрувати всі критерії в єдину оцінку, що значно полегшує прийняття рішення.
- потреба в агрегованій оцінці: завдяки здатності об'єднувати різноманітні критерії в єдине числове значення, загорткова модель значно спрощує процес порівняння альтернатив. Це дозволяє швидко отримати зважену оцінку кожного варіанту, що вкрай важливо для вибору оптимального алгоритму, особливо за складних умов множинних факторів.
- гнучкість: вибір загорткової моделі з ваговими коефіцієнтами забезпечує високий рівень гнучкості. Кожному з критеріїв можна надавати різні ваги, що дозволяє адаптувати модель під конкретні вимоги застосунку. Наприклад, для одного сценарію може бути важлива висока точність, а для іншого — мінімальний вплив на енергоспоживання. Це забезпечує можливість точного налаштування моделі для різних варіантів використання.
- можливість нормалізації: один з важливих аспектів задачі — робота з даними різних масштабів. Критерії, такі як точність і швидкість роботи, мають різні одиниці вимірювання та діапазони значень. Загорткова модель дозволяє легко нормалізувати ці критерії, що дає можливість ефективно їх поєднувати без необхідності враховувати їх еталонні значення чи масштаби.

- простота реалізації: загорткова модель має прозору математичну формулу, що спрощує процес її реалізації та адаптації до конкретної задачі. Її обчислення не вимагає складних алгоритмів або великих витрат ресурсів, що робить її ідеальним вибором для впровадження в мобільних застосунках, де обмеження на швидкість обробки та ресурси можуть бути критичними.

Таким чином, загорткова модель з ваговими коефіцієнтами є оптимальним вибором для цієї задачі, оскільки забезпечує ефективне і гнучке поєднання багатьох критеріїв, дозволяючи отримати об'єктивну та зважену оцінку кожного алгоритму з урахуванням всіх важливих факторів.

4.4 Опис задачі багатокритеріального вибору

Нам потрібно оцінити та обрати найкращий алгоритм для розпізнавання рухів, який буде інтегрований у мобільні застосунки, з урахуванням технічних, функціональних та користувацьких вимог.

4.3.1 Векторний опис альтернатив за критеріями

Для кожного критерію визначено 5-бальну шкалу оцінювання, де 1 бал відповідає найнижчій ефективності, а 5 балів – найвищій.

Таблиця 4.1 – Векторний опис альтернатив за критеріями (таблиця виконана самостійно)

Алгоритм	Точність розпізнавання	Швидкість роботи	Енергоспоживання	Адаптивність до нових типів рухів	Масштабованість
Decision Trees	3	5	4	2	4
Random Forest	4	4	2	3	4
CNN	5	2	2	5	3
HMM	3	3	4	3	2
SVM	4	3	3	3	4

Для оцінювання алгоритмів за п'ятьма критеріями було використано наукові статті та інші джерела, які забезпечують достовірну інформацію про їхню продуктивність у контексті розпізнавання рухів.

4.3.2 Перетворення векторного опису

В задачі багатокритеріального вибору для алгоритмів розпізнавання рухів важливо привести всі критерії до єдиного принципу оптимальності. Оскільки деякі критерії мають оптимізацію «за мінімумом» (наприклад, енергоспоживання), їх потрібно перетворити, щоб вони відповідали принципу «за максимумом». Це дозволяє коректно порівнювати всі альтернативи і уникнути перекручування результатів.

Для цього застосовується наступна формула:

$$X_{max} = X_{норм} = \frac{X_{max} - X_{поточне}}{X_{max} - X_{min}} \quad (4.8)$$

де: X_{max} – максимальне значення шкали критерію;
 $X_{поточне}$ – поточне значення критерію.

У таблиці 4.2 наводиться векторний опис перетворення критеріїв, де всі шкали приведені до принципу оптимальності «за максимумом».

Таблиця 4.2 – Векторний опис перетворення критеріїв до принципу оптимальності «за максимумом» (таблиця виконана самостійно)

Алгоритм	Точність	Швидкість	Енергоспоживання	Адаптивність	Масштабованість
Decision Trees	0	1	0	0	1
Random Forest	0.5	0.67	1	0.3	1
CNN	1	0	1	1	0.5
HMM	0	0.3	0	0.3	0
SVM	0.5	0.3	0.5	0.3	1

4.3.3 Альтернатив принципу Парето

Принцип Парето, або метод Парето-оптимальності, застосовується для вибору найбільш ефективних альтернатив у задачах з кількома критеріями. У цьому контексті альтернативи, які не можна покращити за жодним з критеріїв без погіршення за іншими критеріями, вважаються Парето-оптимальними. Таблиця 3 надає порівняння алгоритмів на основі п'яти критеріальних шкал (точність розпізнавання, швидкість роботи, енергоспоживання, адаптивність до нових типів рухів і масштабованість).

Таблиця 4.3 – Альтернатив принципу Парето (таблиця виконана самостійно)

Алгоритм	Точність	Швидкість	Енергоспоживання	Адаптивність	Масштабованість
Decision Trees	0	1	0	0	1
Random Forest	0.5	0.67	1	0.3	1
CNN	1	0	1	1	0.5
SVM	0.5	0.3	0.5	0.3	1

У даній таблиці для кожного алгоритму наведено його оцінки за критеріями, що дозволяє оцінити, які з них є оптимальними за принципом Парето. Це дає можливість побудувати модель, яка максимально задовольняє вимоги до алгоритмів з урахуванням множинних факторів.

4.3.4 Опис загорткової моделі

Для порівняння та оцінки алгоритмів розпізнавання рухів було використано згорткову модель. Вона забезпечує інтеграцію множини критеріїв у єдиний підсумковий показник Z^* , що характеризує кожну альтернативу.

Формула моделі:

$$Z^* = \sum_{j=1}^n a_j \cdot a_{ij}, \quad (4.9)$$

де: Z^* – підсумкова оцінка для алгоритму i ;

a_j – ваговий коефіцієнт для критерію j ;

a_{ij} – нормалізоване значення критерію j для алгоритму i ;

Цей підхід дозволяє врахувати різні аспекти ефективності алгоритмів (точність, швидкість, енергоспоживання тощо), визначивши їхній відносний вплив на загальну оцінку.

Таблиця 4.4 – Результати згорткової моделі (таблиця виконана самостійно)

Алгоритм	Точність	Швидкість	Енергоспоживання	Адаптивність	Масштабованість	Z^*
Decision Trees	0	1	0	0	1	0.35
Random Forest	0.5	0.67	1	0.3	1	0.695
CNN	1	0	1	1	0.5	0.725
SVM	0.5	0.3	0.5	0.3	1	0.53
Вагових коефіцієнт	0.3	0.25	0.2	0.15	0.1	

Згідно з розрахунками, алгоритми CNN і Random Forest показали найкращі результати. CNN забезпечує найвищу точність і адаптивність, тоді як Random Forest балансує між точністю, швидкістю та енергоефективністю.

4.3.5 Розрахунки корисності

Для оцінки загальної корисності кожної альтернативи за обраною згортковою моделлю з ваговими коефіцієнтами використовується наступна формула:

$$U_i = \sum_{j=1}^n w_j \cdot x_{ij} \quad (4.10)$$

де: U_i – загальна корисність альтернативи i

w_j – ваговий коефіцієнт для критерію j

x_{ij} – нормоване значення альтернативи i за критерієм j

Використовуючи цю формулу, ми можемо оцінити загальну корисність кожної альтернативи за усіма п'ятьма критеріями: точність розпізнавання, швидкість роботи, енергоспоживання, адаптивність до нових типів рухів та масштабованість. Для кожного критерію застосовуються відповідні вагові коефіцієнти, що дозволяє врахувати важливість кожного параметра при розрахунку корисності.

Таблиця 4.5 – Розрахунки корисність (таблиця виконана самостійно)

Алгоритми	Корисність
Decision Trees	0.35
Random Forest	0.6625
Convolutional Neural Networks (CNN)	0.7
Support Vector Machines (SVM)	0.47

У таблиці 4.5 наведено розраховану загальну корисність для кожної з альтернатив. Вищі значення корисності вказують на більш оптимальні варіанти для використання в мобільних застосунках для розпізнавання рухів, з урахуванням усіх критеріїв.

Найбільшу корисність серед альтернатив має CNN ($U = 0.7$), що робить її оптимальним вибором згідно із заданими ваговими коефіцієнтами та критеріями. Обраний підхід до вирішення задачі на основі згорткової моделі з ваговими коефіцієнтами дозволив чітко оцінити кожну з альтернатив і знайти найкраще рішення.

5 ОПИС ЕКСПЕРИМЕНТА ДОСЛІДЖЕННЯ

Цей розділ містить порівняння ефективності двох алгоритмів розпізнавання рухів — Random Forest та Convolutional Neural Networks з метою визначення найбільш оптимального для використання в мобільних застосунках, призначених для відстеження фізичної активності користувачів. Згідно з розрахунками загальної корисності (табл. 5.5), найвищу оцінку отримав CNN, що свідчить про його перевагу за сукупністю критеріїв, таких як точність, швидкодія та придатність для мобільного середовища. Алгоритм Random Forest показав нижчу корисність, однак залишається конкурентним варіантом.

5.1 Опис експерименту

Експеримент полягав у тестуванні алгоритмів на основі даних, зібраних від сенсорів під час виконання користувачем типових фізичних активностей, таких як ходьба, стрибки та стояння на місці. Для навчання та оцінки алгоритмів було використано набір даних, що містив приблизно 1000 зразків сенсорних показників, зібраних від 5 добровольців. Кожен зразок включав часові послідовності прискорення та кутових швидкостей, які слугували вхідними даними для класифікації типів рухів.

Обсяг даних був обмежений через апаратні можливості типового смартфона середнього класу та часові рамки дослідження.

5.2 Технології та програмні інструменти

Розроблений Android-застосунок реалізовано з використанням мови програмування Kotlin та бібліотеки TensorFlow Lite для обробки сенсорних даних. Застосунок здійснює збір, обробку та класифікацію даних з акселерометра для визначення типу фізичної активності користувача. Кожен алгоритм було інтегровано в застосунок у вигляді модуля, що дозволило проводити класифікацію рухів у реальному часі та записувати результати для подальшого аналізу.

Ефективність алгоритмів оцінювалася за п'ятьма критеріями: точність розпізнавання, швидкість роботи, придатність для мобільного середовища до нових типів рухів.

5.3 Підхід до розробки

Для реалізації мобільного застосунку, що розпізнає фізичну активність користувача, було реалізовано повний цикл розробки моделі машинного навчання від збору даних до інтеграції навченої моделі в Android-додаток.

На першому етапі реалізовано Android-додаток на мові Kotlin, який зчитує дані з акселерометра смартфона (див. рис. 5.1).

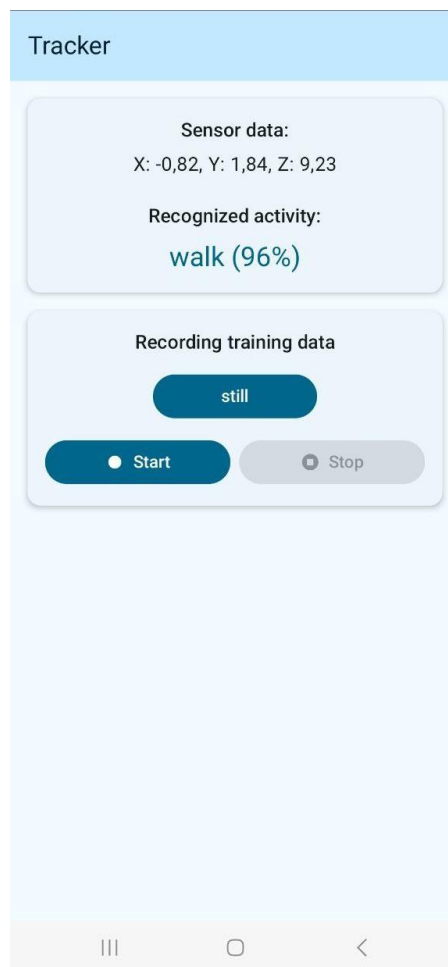


Рисунок 5.1 – Android-додаток (рисунок виконан самостійно)

Дані збираються у вигляді вікон по 50 зразків, що відповідає частоті 50 Гц (1 секунда активності). Для кожного типу активності користувача (ходьба, стрибки,

спокій) дані записуються у формат .csv. Ці дані складають тренувальний набір, який далі використовується для навчання моделей.

На основі зібраних CSV-файлів у середовищі Python реалізовано обробку даних та тренування моделей. Було розроблено скрипт для алгоритму Random Forest з подальшою оцінкою точності.

Розроблено скрипт для Random Forest:

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import classification_report

# === 1. Загрузка и обработка CSV-файлов ===
def extract_features_from_window(window):
    # window - это (50, 3)
    features = []
    for i in range(3): # по каждой оси (X, Y, Z)
        axis = window[:, i]
        features.extend([
            np.mean(axis),
            np.std(axis),
            np.min(axis),
            np.max(axis),
            np.median(axis),
        ])
    return features

def load_and_extract(csv_path, label, window_size=50):
    df = pd.read_csv(csv_path)
    data = df[['accX', 'accY', 'accZ']].values
    features = []
    labels = []
    for start in range(0, len(data) - window_size + 1, window_size):
        window = data[start:start + window_size]
        if window.shape[0] == window_size:
            feature_vector = extract_features_from_window(window)
```

```

        features.append(feature_vector)
        labels.append(label)
    return features, labels

# === 2. Загрузка всех файлов ===
files_and_labels = [
    ('/mnt/data/sensor_data_still_1749261305599.csv', 0),
    ('/mnt/data/sensor_data_walk_1749261422496.csv', 1),
    ('/mnt/data/sensor_data_jump_1749261491814.csv', 2),
]

X, y = [], []
for file_path, label in files_and_labels:
    features, labels = load_and_extract(file_path, label)
    X.extend(features)
    y.extend(labels)

X = np.array(X)
y = np.array(y)

# === 3. Разделение и обучение Random Forest ===
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42, stratify=y)

clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train, y_train)

# === 4. Оценка качества ===
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2)
model = RandomForestClassifier(n_estimators=100)
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
print(classification_report(y_test, y_pred))

```

На основі зібраних CSV-файлів реалізовано також скрипт для згорткової нейронної мережі (CNN), яка була натренована у середовищі TensorFlow Keras.

Модель призначена для розпізнавання фізичної активності користувача на основі сегментів даних з акселерометра.

Розроблено скрипт для CNN:

```
import pandas as pd
import numpy as np
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv1D, MaxPooling1D, Flatten,
Dense
from tensorflow.keras.utils import to_categorical
from sklearn.model_selection import train_test_split

# === 1. Загрузка и обработка CSV-файлов ===
def load_and_segment(csv_path, label, window_size=50):
    df = pd.read_csv(csv_path)
    data = df[['accX', 'accY', 'accZ']].values
    segments = []
    labels = []
    for start in range(0, len(data) - window_size + 1, window_size):
        window = data[start:start + window_size]
        if window.shape[0] == window_size:
            segments.append(window)
            labels.append(label)
    return segments, labels

# Пути к файлам
files_and_labels = [
    ('/mnt/data/sensor_data_still_1749261305599.csv', 0),
    ('/mnt/data/sensor_data_walk_1749261422496.csv', 1),
    ('/mnt/data/sensor_data_jump_1749261491814.csv', 2)
]

X, y = [], []
for file_path, label in files_and_labels:
    segments, labels = load_and_segment(file_path, label)
    X.extend(segments)
```

```

    y.extend(labels)

X = np.array(X)
y = np.array(y)

# === 2. Подготовка данных ===
num_classes = 3
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2, random_state=42, stratify=y)
y_train_cat = to_categorical(y_train, num_classes=num_classes)
y_test_cat = to_categorical(y_test, num_classes=num_classes)

# === 3. Построение и обучение модели ===
model = Sequential([
    Conv1D(32, kernel_size=3, activation='relu', input_shape=(50, 3),
padding='same'),
    MaxPooling1D(2),
    Conv1D(64, kernel_size=3, activation='relu', padding='same'),
    MaxPooling1D(2),
    Flatten(),
    Dense(64, activation='relu'),
    Dense(num_classes, activation='softmax')
])

model.compile(optimizer='adam', loss='categorical_crossentropy',
metrics=['accuracy'])
model.fit(X_train, y_train_cat, epochs=20, batch_size=16,
validation_data=(X_test, y_test_cat))

# === 4. Сохранение Keras модели ===
model.save("fitness_model.h5")
print("Модель сохранена в fitness_model.h5")

# === 4. Оценка качества ===

X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.2)
y_train_cat = to_categorical(y_train)

```

```

y_test_cat = to_categorical(y_test)

model = Sequential([
    Conv1D(32, kernel_size=3, activation='relu', input_shape=(50,
3)),
    Conv1D(64, kernel_size=3, activation='relu'),
    GlobalAveragePooling1D(),
    Dense(3, activation='softmax')
])
model.compile(optimizer='adam', loss='categorical_crossentropy',
metrics=['accuracy'])
model.fit(X_train, y_train_cat, epochs=10, batch_size=32)

```

Готова tflite-модель була інтегрована у мобільний застосунок. У додатку реалізовано повну обробку сигналів акселерометра: дані формуються у вікна, подаються в інтерпретатор TensorFlow Lite, де відбувається передбачення поточної активності. Результати класифікації виводяться користувачу в режимі реального часу [13].

Увесь процес від збору даних до мобільної класифікації реалізовано локально на пристрої. Це дозволяє проводити експерименти з різними моделями машинного навчання.

5.4 Результати експерименту

Після тестування на даних від 5 добровольців у реальних умовах отримано результати, представлені в таблиці 5.1 (оцінка за шкалою від 0 до 1, де 1 — найкращий результат)

Таблиця 5.1 – Результати тестування (таблиця виконана самостійно)

Алгоритм	Точність	Швидкість	Придатність для мобільного середовища	Z
Random Forest	0.88	1	1	0.957
CNN	0.94	0.87	0.87	0.883

Оцінки швидкості та придатності для мобільного середовища були сформовані на основі тестування на реальних пристроях, а також аналізу обчислювальних характеристик моделей.

Алгоритм Random Forest продемонстрував високу швидкодію як на етапі тренування, так і під час передбачення. Завдяки обмеженій кількості ознак (15 статистичних характеристик на кожне вікно), передбачення одного фрейму виконується за кілька мілісекунд, навіть на малопотужних процесорах мобільних пристроїв. Алгоритм не потребує попередньої нормалізації даних або складної обробки сигналів. Модель має незначний обсяг (кілька мегабайт або менше), не вимагає додаткових бібліотек чи спеціалізованого апаратного прискорення. Завдяки цьому вона легко інтегрується в Android-додаток, не створює надмірного навантаження на акумулятор і добре підходить для задач в умовах обмежених ресурсів.

Модель CNN, хоча й оптимізована у форматі TensorFlow Lite, залишається складнішою за своєю структурою, ніж дерева рішень. Навіть за ефективного використання обчислювального графа, час передбачення може коливатися від 5 до 20 мс на середньостатистичному смартфоні. На менш продуктивних пристроях це може призводити до затримок у режимі реального часу, що дещо знижує загальну ефективність. Крім того, для роботи моделі потрібне встановлення інтерпретатора TensorFlow Lite, а також вона може інтенсивно навантажувати центральний або графічний процесор під час обробки сигналів, що веде до підвищеного енергоспоживання. Більш складна архітектура CNN займає більше пам'яті, порівняно з Random Forest. Проте після конвертації у формат .tflite модель стає значно доступнішою для мобільного середовища, що дозволяє її використання з певними обмеженнями.

У результаті, Random Forest ($Z = 0.957$) виявився найбільш збалансованим за сукупністю характеристик, тоді як CNN ($Z = 0.883$), незважаючи на вищу точність, виявився менш практичним через вищу ресурсоємність.

ВИСНОВКИ

У ході дослідження було проведено комплексний аналіз алгоритмів розпізнавання рухів для їхньої інтеграції в мобільні застосунки, спрямовані на відстеження фізичної активності користувачів. Основна увага приділялася оцінці п'яти ключових алгоритмів: Decision Trees, Random Forest, Convolutional Neural Networks (CNN), Hidden Markov Models (HMM) та Support Vector Machines (SVM). Для порівняння було сформовано множину критеріїв, які найбільш повно відображають вимоги до алгоритмів у контексті їхнього використання в мобільних технологіях.

З метою інтегральної оцінки ефективності кожного алгоритму була застосована згорткова модель, що дозволила об'єктивно врахувати вплив усіх критеріїв: точність розпізнавання, швидкість роботи, енергоспоживання, адаптивність до нових типів рухів та масштабованість. Визначення вагових коефіцієнтів для критеріїв здійснювалося за допомогою методу аналізу ієрархій (АНР), що забезпечило обґрунтований розподіл пріоритетів між різними характеристиками.

Результати розрахунків показали, що найвищу загальну корисність має алгоритм CNN ($U = 0.7$), який демонструє найкращі показники точності та адаптивності. Алгоритм Random Forest ($U = 0.6625$) зайняв друге місце, забезпечуючи збалансовану продуктивність за всіма критеріями, особливо відзначившись швидкістю роботи. Support Vector Machines (SVM) та Decision Trees виявилися менш ефективними через обмежену адаптивність та високий рівень енергоспоживання у порівнянні з лідерами.

Обраний підхід до вирішення задачі на основі згорткової моделі дозволив провести систематичний аналіз та знайти найоптимальніший варіант для використання в мобільних застосунках. CNN є найбільш перспективним алгоритмом для задач розпізнавання рухів, враховуючи його здатність працювати з великими обсягами даних та забезпечувати високу точність. Однак Random Forest

залишається хорошим вибором у сценаріях, де важливіша швидкість обробки та енергоефективність.

Практична частина роботи, зокрема розробка та тестування Android-застосунку, підтвердила теоретичні висновки та дала змогу оцінити алгоритми в реальних умовах. Було створено програмне рішення, яке інтегрувало алгоритми, а їхня продуктивність була протестована на даних від сенсорів смартфона. Це дозволило не лише перевірити теоретичні припущення, а й отримати цінні практичні дані про поведінку алгоритмів на мобільних пристроях із обмеженими ресурсами.

Результати цього дослідження можуть бути використані як основа для розробки сучасних мобільних застосунків для відстеження фізичної активності, а також для подальшого вдосконалення алгоритмів у цій галузі.

Результати роботи апробовані на VIII Всеукраїнській студентській науковій конференції «Експериментальні та теоретичні дослідження в контексті сучасної науки» (див. додаток А) [14].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Андріяш С. Проектування користувацького інтерфейсу мобільного додатку для моніторингу здоров'я та лікування дефіциту вітамінів групи D [Електронний ресурс] // DSpace at ZNU. – URL: <https://dspace.znu.edu.ua/jspui/bitstream/12345/12816/1/Андріяш%20Софія%20ПЗ.pdf> (дата звернення: 16.01.2025).
2. Точність носимих фітнес-пристроїв: Що потрібно знати [Електронний ресурс] // Bimedis. – URL: <https://bimedis.net/latest-news/browse/730/tochnist-nosimih-fitness-pristroyiv-sho-potribno-znati> (дата звернення: 16.01.2025).
3. Функції та налаштування розумного годинника [Електронний ресурс] // Gelius.ua. – URL: <https://gelius.ua/blog/kak-rabotaut-smart-chasy/> (дата звернення: 16.01.2025).
4. 10 найкращих інструментів для тренувань AI (січень 2025) [Електронний ресурс] // Unite.ai. – URL: <https://www.unite.ai/uk/найкращі-інструменти-для-тренувань-штучного-інтелекту/> (дата звернення: 16.01.2025).
5. Lara O.D., Labrador M.A. A Survey on Human Activity Recognition using Wearable Sensors [Електронний ресурс] // IEEE Communications Surveys & Tutorials. – 2013. – URL: <https://ieeexplore.ieee.org/document/6372049> (дата звернення: 16.01.2025).
6. Андронов В. А., Калугін В. Д., Левтеров О. А., Тютюник В. В. Науково-технічні основи синтезу системи моніторингу надзвичайних ситуацій різного характеру за основними характеристиками технічних засобів реєстрації факторів небезпек. Прикладная радиоэлектроника. 2016. Т. 15, № 4. С. 327–333.
7. Шумілін В.О., Лєсна Н.С. Результати дослідження ефективності методів та засобів розробки бессерверних додатків для обробки зображень//Наука онлайн: міжнародний електронний науковий журнал. - 2021. - №5.
8. Ревенчук І.А., Д. С. Бабуріна. Архітектурні підходи для задачі динамічної обробки зображень// Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління : тез. доп. 12-ої міжнар. наук.-

техніч. конф., 27–28.04.2022 р.– Т. 2. Баку–Харків–Жиліна. С.128.

9. Кобзєв В. Г., Яковлєв С. В., Назаров О. С., Назарова Н. В., Горішня К. О. Модифікації методу опорних векторів для класифікації та виявлення аномалій у задачах обробки зображень. *Міжнародна науково-практична конференція «Застосування інформаційних технологій у підготовці та діяльності сил охорони правопорядку»*. Харків, 2024. С. 45–50.

10. Advances in Spatio-Temporal Segmentation of Visual Data / editors : Mashtalir V., Ruban I., Levashenko V. 2020. 274 p."A Method of Combining Hidden Markov Model and Convolutional Neural Network for Text Classification." MDPI. URL: <https://www.mdpi.com/2076-3417/11/14/6350> (дата звернення: 16.01.2025).

11. "Random Forest/Hidden Markov Model for Transportation Mode Recognition Using Smartphone Sensors." ResearchGate. URL: https://www.researchgate.net/publication/359500979_RANDOM_FORESTHIDDEN_MARKOV_MODEL_FOR_TRANSPORTATION_MODE_RECOGNITION_USING_SMARTPHONE_SENSORS (дата звернення: 16.01.2025).

12. "A Performance Study of Hidden Markov Model and Random Forest in Internet Traffic Classification." ResearchGate. URL: https://www.researchgate.net/publication/298363955_A_Performance_Study_of_Hidden_Markov_Model_and_Random_Forest_in_Internet_Traffic_Classification (дата звернення: 16.01.2025).

13. GitHub. – Tracker. *GitHub*. URL: <https://github.com/vovaroma/Tracker.git> (дата звернення: 14.06.2025).

14. Ромашов В.В. Програмна система для надання та отримання навчальних послуг. VIII Всеукраїнська мультидисциплінарна студентська наукова конференція «Експериментальні та теоретичні дослідження в контексті сучасної науки». м. Львів. 2025. – С. 108-111.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ ЗА НАУКОВИМИ НАПРЯМАМИ КЕРІВНИКА ТА НАУКОВЦІВ КАФЕДРИ ПРОГРАМНОЇ ІНЖЕНЕРІЇ

7. Шумілін В.О., Лесна Н.С. Результати дослідження ефективності методів та засобів розробки бессерверних додатків для обробки зображень//Наука онлайн: міжнародний електронний науковий журнал. - 2021. - №5.

8. Ревенчук І.А., Д. С. Бабурина. Архітектурні підходи для задачі динамічної обробки зображень// Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління : тез. доп. 12-ої міжнар. наук.-техніч. конф., 27–28.04.2022 р.– Т. 2. Баку–Харків–Жиліна. С.128.

9. Кобзєв В. Г., Яковлєв С. В., Назаров О. С., Назарова Н. В., Горішня К. О. Модифікації методу опорних векторів для класифікації та виявлення аномалій у задачах обробки зображень. *Міжнародна науково-практична конференція «Застосування інформаційних технологій у підготовці та діяльності сил охорони правопорядку»*. Харків, 2024. С. 45–50.