МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

НЕСКОРОДЄВА АНАСТАСІЯ РОМАНІВНА

Допускається до захисту:
в.о. завідувача кафедри
інформаційних технологій
к.т.н., доцент
_____ О.В. Зелінська
«____» _____2024

ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ПОСЛІДОВНОСТЕЙ КАДРІВ ГІМНАСТИЧНИХ ПОЗ

Спеціальність 122 «Комп'ютерні науки»

Кваліфікаційна (магістерська) робота

Науковий керівник:
Федоров Є.Є., професор кафедри
інформаційних технологій,
д.т.н., професор
_____
(підпис)

Оцінка:_____/ _____/_____
(бали за шкалою ECTS /за національною шкалою)

Голова ЕК: _____
(підпис)

Вінниця 2024

АНОТАЦІЯ

Нескородєва А. Р. Інтелектуальний аналіз послідовностей кадрів гімнастичних поз. Спеціальність 122 «Комп'ютерні науки». Освітня програма "Комп'ютерні технології обробки даних (Data Science)". Донецький національний університет імені Василя Стуса, Вінниця, 2024.

У кваліфікаційній (магістерській) роботі досліджено актуальну задачу інтелектуального аналізу послідовностей кадрів гімнастичних поз. Сформовано набір даних прикладів виконання гімнастичних елементів. Побудовано інтелектуальні моделі на основі згорткових та рекурентних нейромереж для класифікації елементів художньої гімнастики на основі зображень. Виконано ідентифікацію, програмну реалізацію та чисельне дослідження нейромережевої класифікації послідовностей кадрів гімнастичних поз.

Ключові слова: інтелектуальний аналіз, послідовність кадрів гімнастичних поз, набір даних, нейромережа, CNN, LSTM.

80 с., 3 табл., 19 рис., 51 джерела.

ABSTRACT

Neskorodieva A. Intellectual analysis of frames sequences of gymnastic poses. Specialty 122 "Computer Science". Educational program "Computer technologies of data processing (Data Science)". Vasyl' Stus Donetsk National University, Vinnytsia, 2024.

In the qualification (master's) thesis the actual problem of intellectual analysis of frames sequences of gymnastic poses is investigated. A data set of examples performing gymnastic elements has been formed. Intelligent models based on convolutional and recurrent neural networks are built to classify rhythmic gymnastics elements based on images. The identification, software implementation and numerical study of neural network classification of frames sequences of gymnastic poses are performed.

80 p., 19 figures., 3 tab., 51 ref.

Keywords: intellectual analysis, sequence of frames of gymnastic poses, data set, neural network, CNN, LSTM.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

CV – computer vision, (комп’ютерний зір)

CNN - convolutional neural network, (згорткові нейронні мережі)

HAR - human action recognition, (розпізнавання дій людини)

LSTM – Long Short-Term Memory, (довготривала короткочасна пам’ять)

LRCN – Long-term Recurrent Convolutional Networks, (довготривала рекурентно-згорткова мережа)

ConvLSTM – Convolutional LSTM, (згорткові LSTM)

TimeDistributed – Time Distributed Layer, (шар розподіленого часу)

MaxPooling2D – Max Pooling 2D Layer, (двовимірний шар максимального згортання)

ВСТУП

Спорт завжди був важливим елементом суспільства, що не тільки сприяє фізичному розвитку, а й формує волю, витривалість та прагнення до досягнення високих результатів. Протягом століть спорт розвивався, набуваючи нових форм та технік, і сьогодні у світі існує велика різноманітність дисциплін, кожна з яких має свої особливості. Художня гімнастика – це один із найяскравіших видів спорту, який поєднує елементи танцю, акробатики та фізичної підготовки, створюючи унікальне поєднання сили, гнучкості та грації. Цей вид спорту вимагає високої майстерності від спортсменів і особливої точності від суддів, які мають оцінювати виступи за складними критеріями.

Сучасна художня гімнастика як вид спорту розвивалася на перетині мистецтва і фізичної активності, відображаючи естетичні вимоги та спортивну підготовку спортсменів. На відміну від багатьох інших видів спорту, де результати можна виміряти конкретними цифрами (як-от швидкість, час або вага), художня гімнастика залежить від комплексного оцінювання багатьох аспектів виступу – складність тіла, складність предмету, артистизму та виконання, що вимагає від суддів надзвичайної уваги до деталей. Цей процес оцінювання є суб'єктивним і залежить від численних факторів, які впливають на суддівське рішення, що інколи може спричиняти дискусії та питання щодо об’єктивності підсумкових оцінок.

З розвитком технологій, потреба в автоматизації та забезпеченні більшої точності оцінювання стає все більш актуальною. Використання алгоритмів, здатних аналізувати послідовності кадрів, надає можливість об'єктивного визначення точності виконання спортивних елементів, що є важливим для оцінювання виступів у художній гімнастиці. Технології можуть не тільки підвищити точність суддівства, а й сприяти більш прозорому процесу оцінювання, що забезпечить рівні можливості для всіх спортсменів.

Об’єкт дослідження - автоматизований аналіз послідовностей кадрів гімнастичних поз.

Предмет дослідження – моделі та методи інтелектуального аналізу послідовностей кадрів гімнастичних поз.

Мета дослідження – підвищення ефективності розпізнавання послідовностей кадрів гімнастичних елементів на підставі методів інтелектуального аналізу.

Завдання дослідження:

- виконати аналіз моделей, методів та технологій інтелектуального аналізу послідовностей кадрів дій людини, зокрема у спорті;
- дослідити існуючи набори даних дій спортсменів для інтелектуального аналізу;
- створити набір даних для аналізу послідовностей кадрів гімнастичних поз;
- виконати вибір інтелектуальних моделей для аналізу послідовностей кадрів гімнастичних поз;
- виконати структурну та параметричну ідентифікацію обраної інтелектуальної моделі;
- провести чисельне дослідження моделей інтелектуального аналізу послідовностей кадрів гімнастичних поз;
- виконати програмну реалізацію інформаційної технології інтелектуального аналізу послідовностей кадрів гімнастичних поз.

Методи дослідження: аналіз при дослідженні наборів даних, моделей, методів та технології розпізнавання людських дій у спорті; методи структурної та параметричної ідентифікації нейромережевих моделей; методи об’єктного моделювання при розробці програмного забезпечення; методи чисельного дослідження нейромережевих моделей.

Наукова новизна:

- удосконалена структура нейромережевої моделі аналізу послідовностей кадрів гімнастичних поз;
- отримала подальший розвиток інформаційна технологія аналізу послідовностей кадрів гімнастичних поз.

Практична значимість полягає у створені нового набору даних, що враховує чотири поширених елементів художньої гімнастики з різною складністю розпізнавання; розробці алгоритму параметричної ідентифікації моделей; створені програмної реалізації інформаційної технології аналізу послідовностей кадрів гімнастичних поз.

Таким чином розроблені моделі можуть у майбутньому стати базовою частиною систем для об'єктивного оцінювання елементів художньої гімнастики. Впровадження таких систем має потенціал змінити підхід до суддівства у художній гімнастиці, забезпечивши вищий рівень справедливості та прозорості у процесі оцінювання, а також надасть спортсменам і тренерам новий інструмент для вдосконалення техніки й підвищення професійного рівня. В подальшому, розроблені методи можна адаптувати та застосувати для аналізу медичних або військових зображень, зберігаючи при цьому конфіденційність персональних даних.

У медицині такі технології можуть застосовуватися для контролю рухової активності пацієнтів під час реабілітації або для моніторингу прогресу фізичної терапії. У військовій справі подібні алгоритми здатні аналізувати рухи солдатів під час тренувань, що допоможе покращити тактичну підготовку та забезпечити стандарти виконання бойових вправ. Таким чином, розробка цих інструментів може відкрити широкі можливості для їх використання не лише у спорті, а й у суміжних галузях, де критично важливою є точність та об'єктивність аналізу рухів.

Результати роботи доповідались на наступних конференціях та конкурсах і опубліковані в роботах [1-4].

Структура магістерської роботи будується на трьох основних розділах, які послідовно висвітлюють проблематику, теоретичне обґрунтування, а також результати практичної реалізації дослідження. У першому розділі проводиться аналіз сучасного стану проблеми, пов'язаної з розпізнаванням та класифікацією дій у спортивному середовищі. Розглядаються існуючі моделі аналізу послідовностей кадрів та існуючі набори даних і їхня придатність для

поставленого завдання. У другому розділі викладено теоретичні аспекти моделей для аналізу послідовностей кадрів, зокрема вибір та обґрунтування використання обраних нейронних мереж. Третій розділ присвячений практичній реалізації обраний моделей. У ньому детально розглядається процес збору та обробки даних послідовностей кадрів гімнастичних поз. Описуються технічні аспекти розробки моделей, їх програмна реалізація, структурна і параметрична ідентифікація, та тестування. Представляються результати експериментів, оцінюється точність розпізнавання гімнастичних поз, порівнюються результати роботи розроблених моделі.

Кожен із розділів логічно пов'язаний, забезпечуючи цілісність і послідовність дослідження, що відповідає поставленій меті роботи.

РОЗДІЛ 1
ІНТЕЛЕКТУАЛЬНИЙ АНАЛІЗ ТА ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ОБРОБКИ ПОСЛІДОВНОСТЕЙ КАДРІВ

1.1 Основні поняття, завдання і проблеми інтелектуального аналізу дій людини

Штучний інтелект в основі якого лежить інтелектуальний аналіз даних - одна з найбільших галузей комп'ютерних наук, спрямована на створення систем, здатних виконувати завдання, для яких потрібен певний рівень інтелекту. Іншими словами, метою є створення машин, які можуть поводитися, як людина, мислити, як людина, і бути здатними приймати рішення самостійно [5].

Область ШІ, що займається обробкою візуальних даних, таких як зображення і відео, так само, як це робить людина, витягаючи з них корисну інформацію і розуміючи їх зміст, називається комп'ютерним зором [6]. Комп'ютерний зір постійно зростає завдяки великій кількості отриманих візуальних даних, таких як системи безпеки, дорожні камери та люди, які щодня завантажують їх в Інтернет [7].

Одними з основних завдань CV є розпізнавання того, чи є об'єкт або дія на зображенні чи відео, до якої категорії він належить, де він розташований і які пікселі належать об'єкту [8].

Завдання інтелектуального аналізу дій полягає в тому, щоб розпізнати, які людські дії виконуються, в якій послідовності кадрів, в якому часовому інтервалі і де в сцені знаходиться людина, що діє.

У роботі [9] автори зазначають, що термін "дія людини" в дослідженнях CV "варіюється від простого руху кінцівки до спільного складного руху декількох кінцівок і тіла людини". Крім того, вони вважають цей процес динамічним і вказують, що його зазвичай передають у відеоролику тривалістю кілька секунд.

У [10] автори називають дію "простими руховими патернами, які зазвичай виконуються однією людиною і, як правило, тривають протягом

короткого часу".

У деяких випадках для виконання завдання потрібно використовувати предмет або взаємодіяти з іншими людьми. Тому існує різний рівень складності дій. Деякі прості дії можна розпізнати за одним зображенням або кадром. Деякі складні дії відбуваються протягом більш тривалого періоду, і для їх розпізнавання потрібна послідовність кадрів. Наприклад, просту дію, таку як стояння або підняття руки, можна розпізнати за допомогою одного кадру. Довша послідовність відеокадрів потрібна, щоб визначити, яка дія відбулася для більш складної дії, наприклад, стрибка в довжину, що складається з різних фізичних рухів, таких як біг, стрибки, приземлення та вставання, які тривають протягом певного часу. Складність дії може також зростати, якщо відбувається взаємодія з одним або кількома об'єктами, наприклад, удар по м'ячу бейсбольною битою, коли людина використовує один об'єкт, бейсбольну биту, для виконання дії над іншим об'єктом, м'ячем.

Розпізнавання людських дій (Human Action Recognition, HAR) - це складне завдання, яке використовується в таких видах спорту, як волейбол, баскетбол, футбол і теніс, для виявлення гравців і розпізнавання їхніх дій та дій команд під час тренувань, матчів, розминок або змагань. Розпізнавання людських дій має на меті виявити особу, яка виконує дію, або декількох осіб, які виконують дії на невідомому відео послідовності, визначити тривалість дії та ідентифікувати її тип. Щоб зробити все це можливим, HAR знаходиться на перетині різних галузей ШІ, таких як CV та машинне навчання, разом з обробкою зображень (рис. 1.1).

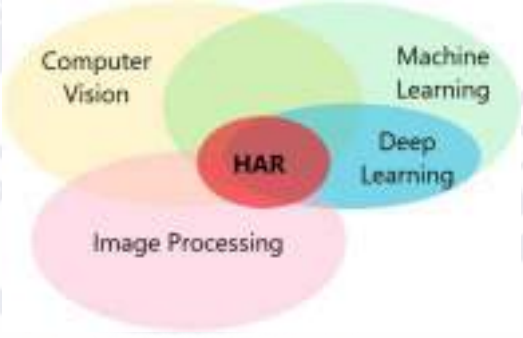

Рисунок 1.1 Розпізнавання людських дій на перетині різних галузей комп'ютерних наук

Це комплексне завдання, яке включає в себе виявлення людини на зображенні або відео, визначення місця дії в часі і просторі на відео послідовності, і, нарешті, розпізнавання дії.

HAR все більше представлений у спорті і стає все більш спрямованим на розпізнавання подібних дій у конкретній спортивній галузі.

Проблеми HAR можуть бути дуже різними, враховуючи різні застосування і різноманітний вибір даних, тому жоден підхід не підходить для вирішення всіх проблем, з якими можна зіткнутися. Більше того, можливі застосування належать до багатьох сфер. Наприклад, програми можна використовувати для аналізу одужання пацієнта або виявлення несподіваних подій, таких як знепритомніння літньої людини в будинку, або виявлення підозрілої поведінки під час відеоспостереження, аналізу поведінки гравців для різних взаємодій між людиною і комп'ютером тощо.

Багато робіт присвячено дослідженню HAR, здебільшого зосереджуються на повсякденних діях, простих діях з різних видів спорту та інших реалізаціях. Повсякденні дії включають, наприклад, ходьбу, рукостискання, сидіння, їжу тощо, тоді як прості дії з різних видів спорту включають їзду на велосипеді, верхову їзду, біг тощо.

У [11] автори представляють огляд розробок у напрямку розпізнавання людських дій, від обговорення класичних методів, які використовують ручні ознаки, до методів, орієнтованих на глибоке навчання.

У [12] автори класифікували методи HAR на дві основні категорії, унімодальні (наприклад, стохастичні, засновані на правилах) та мультимодальні (наприклад, поведінкові, афективні), відповідно до природи сенсорних даних, які вони використовують, а потім проаналізували їх.

У [9] автори розглянули розпізнавання для деяких загальних дій і обговорили різні проблеми, такі як внутрішні і міжкласові варіації, захаращеність фону і рух камери, недостатня кількість анотованих даних, словниковий запас дій і нерівномірна передбачуваність. Вони використовують методи, спільні для різних типів дій, такі як декомпозиція складних дій на

високих рівнях ієрархії на комбінацію дій на нижчих рівнях і боротьба з нерівномірною передбачуваністю, оскільки деякі дії миттєво передбачувані, а інші потребують більше кадрів для спостереження.

Ці проблеми ще більш виражені у спорті, тому автори часто намагаються вдосконалити існуючі методи або розробити нові, щоб зменшити та подолати їх у контексті спортивних дій. Крім того, у спорті є й інші виклики, такі як більш складні сцени, кілька гравців на полі, оклюзія, дії, які дуже мало відрізняються, тощо.

Наприклад, в [13] автори надали огляд існуючих комерційних технологій для спортивної сфери. Проте вони дійшли висновку, що не існує інструменту, який може автоматично розпізнавати і класифікувати спортивну діяльність для змістовного аналізу. Вони також обговорили впровадження глибокого навчання для спортивних дій, але в їхньому огляді не так багато робіт, що зосереджуються на конкретних діях для певного виду спорту, оскільки дослідження більше зосереджені на розумінні ролі глибокого навчання.

Розпізнавання людських дій у спорті - це використання методів комп'ютерного зору для виявлення гравців або розпізнавання дій чи діяльності спортсменів. Діяльність гравців або спортсменів можна відстежувати під час розминки та фітнес-тренуваннях, спеціальних тренувань для певного виду спорту, матчів або змагань.

Таким чином, HAR можна використовувати для вирішення завдань моніторингу продуктивності спортсменів (наприклад, біг), порівняння різних дій, що виконуються різними гравцями, або декількох дій одного гравця (наприклад, удар ззаду в тенісі, подача у волейболі), щоб допомогти відпрацювати техніку, або поліпшити стиль тощо.

Крім того, за допомогою HAR можна провести автоматизований статистичний аналіз спортивного матчу або виступів окремих спортсменів. Як правило, спортивні сцени - це динамічні сцени, в яких спортсмени виконують кілька дій, використовуючи різні техніки, рухаючись і змінюючи позиції, тому

розпізнавання дій спортсмена є складним завданням.

У командних видах спорту сцена ще складніша, оскільки на полі багато гравців, часто з команд-суперників, за якими потрібно паралельно стежити і розпізнавати їхні дії. Крім того, гравці можуть виконувати різні дії одночасно, але з часовим лагом, так що кожен спортсмен виконує окрему дію в певний момент певним чином. Часто гравці змінюють своє положення, відстань і кут нахилу до камери, прикривають один одного, входять і виходять з поля зору камери тощо, що викликає такі проблеми, як розпізнавання декількох об'єктів з оклюзією і захаращеність сцен [14].

Таким чином, існує велике різноманіття людських дій і умов їх спостереження, які підлягають аналізу по відео. В той же час не існує універсальних методів, моделей і інформаційних технологій щодо їх аналізу. Розглянемо існуючи підходи до класифікації спортивних дій.

1.2. Підходи до класифікації спортивних дій

Розпізнавання людських дій базується на класифікації спортивних дій. Розглянемо існуючи підходи до класифікації.

Згідно з [15], існує чотири рівні складності людської діяльності: жести, дії, взаємодії та групова діяльність.

Жестом вважається базовий рух, що включає деякі рухи рук, пальців або частини тіла, наприклад, витягування руки.

На відміну від жестів, які описуються простими рухами, дія складається з декількох рухів, організованих у часі. Їй притаманна певна мета, яку потрібно досягти, наприклад, для того, щоб ходити, потрібно піднімати ноги і рухати руками, щоб утримувати рівновагу і рухатися.

Взаємодія відбувається за участі двох або більше осіб, а іноді й предметів, наприклад, коли двоє людей б'ються або баскетболіст кидає м'яч.

Нарешті, групові дії виконуються групами, що складаються з кількох людей, які використовують предмети, наприклад, група людей, які біжать.

Запропонований у [16] метод категоризації спортивних дій розроблений більш детально і явно модифікований для застосування HAR у спорті. Крім

того, дії та види діяльності розмежовуються таким чином, що дії включають одного гравця, а види діяльності - кількох гравців. Певні правила визначають дії та діяльність у спорті. Таким чином, для того, щоб класифікувати дію або діяльність як дію, потрібен експерт у певній галузі зі знанням заздалегідь визначеного набору фізичних рухів, які повинні бути виконані. Звичайно, можуть бути відмінності у виконанні дії або активності, які відображають стиль гравця. Проте, як тільки попередньо визначені рухи виконані, дії та види діяльності можна класифікувати та розпізнати, незалежно від того, наскільки широко гравці рухають руками, чи роблять вони більші кроки або стрибають вище за інших.

Також автори пропонують всеохоплюючу систематизацію людських дій, пов'язаних зі спортом, яка складається з трьох основних категорій: індивідуальні дії, спільні дії та командні дії. Ці дії та активності можуть бути навмисними та ненавмисними. Наприклад, передача м'яча у футболі, удар у рамку воріт, випадкове падіння або контакт, удар ногою, гра рукою у футболі тощо. Обидва типи можуть бути розпізнані в HAR у спорті, де для першого з них метою є визначення того, яка дія була виконана і чи була вона виконана правильно, а для другого - чи відбулася неочікувана подія або інцидент. Аналізуючи ненавмисні дії та заходи, було помічено, що це здебільшого індивідуальні дії та спільні заходи. Однак є також приклади ненавмисних командних дій, таких як штраф у хокеї, коли команда має більше, ніж дозволено, гравців на льоду одночасно.

Коли в дії бере участь лише один учасник, це визначається як індивідуальна дія. Вона поділяється на кілька підкатегорій, залежно від складності й участі додаткових об'єктів. Найпростіша форма таких дій — це жести, або одиничні рухи певних частин тіла, найчастіше рук, що використовуються для передачі сигналів. Вони поширені як серед суддів, які, наприклад, можуть сигналізувати фоли або порушення, так і серед гравців, які, вказуючи жестом, можуть просити «тайм-аут». Ці рухи є найбільш базовою формою комунікації без слів і відповідають категорії жестів, визначеній у

попередніх дослідженнях.

Прості дії є складнішими, ніж жести, оскільки залучають різні частини тіла, що рухаються одночасно. Ці рухи поділяються на дії, виконувані на місці, і дії з переміщенням. До простих дій на місці належать такі рухи, як присідання або пліє в балеті, а також ухиляння у боксі. Такі дії не передбачають переміщення по полю, але вимагають точної координації рухів. Прості дії з переміщенням включають послідовні повторення рухів під час пересування, наприклад, кроки або біг, які формують ритмічну послідовність дій.

Дії, які включають взаємодію з об'єктами, охоплюють рухи з участю предметів, як-от кидання або ловля м'яча. Така форма взаємодії дозволяє більш різноманітні й складні рухи, оскільки об'єкт стає частиною дії, що підвищує вимоги до координації. Хоча в попередніх дослідженнях взаємодія часто охоплює як об'єкти, так і інші особи, цей вид рухів тут розглядається окремо, зосереджуючись саме на залученні предметів.

Складні дії є найбільш комплексною категорією індивідуальних дій, що включають поєднання простих рухів і взаємодій з об'єктами. Наприклад, у баскетболі кидок у стрибку складається з вертикального стрибка, під час якого гравець утримує м’яч однією рукою, і кінцевого руху другою рукою, спрямованого на завершення кидка в кошик. Цей рух може бути також виконаний під час дриблінгу, що ускладнює дію, оскільки вимагає координації рухів під час переміщення. Іншим прикладом складної дії є подача в тенісі, що охоплює два послідовні рухи з м’ячем: гравець спершу підкидає м’яч однією рукою, а потім наносить удар по ньому ракеткою, додаючи обертання і швидкість.

Дії, що включають щонайменше двох гравців, але не всю команду, називаються спільною діяльністю. Такі дії передбачають поєднання однієї або кількох простих дій, взаємодій з об'єктами або складних дій, що здійснюються учасниками одночасно або послідовно. У класифікаціях спортивних дій спільна діяльність найбільш відповідає категорії взаємодії, однак в цьому підході взаємодію з об'єктами виділили окремо, як індивідуальну дію, що

відображає взаємодії гравців зі спортивними предметами. Дослідження підтверджують, що комбінація простих і складних дій із залученням об'єктів формує унікальні форми взаємодії між спортсменами, підвищуючи координацію та динаміку командного процесу.

Коли дії залучають всю команду і мають стратегічний характер, їх визначають як командну діяльність. Командна активність аналогічна категорії групової діяльності, але в даній класифікації вона більш детально поділяється за типами спортивних дій та рівнем участі. Командна діяльність може включати різні комбінації індивідуальних і спільних дій, часто відображаючи узгоджену роботу групи з конкретною метою. Наприклад, атакувальні дії у футболі передбачають одночасне пересування гравців до воріт суперника з передачею м’яча, що охоплює індивідуальні й командні дії, зокрема завершальний удар по воротах.

Особливою відмінністю наведеної схеми класифікації спортивних дій є врахування складності дій не тільки через залучення об'єктів, але і через кількість гравців, що беруть участь. Традиційна класифікація не завжди чітко враховує, як кількість учасників впливає на рівень складності виконуваних дій. Автори зазначають, що кількість спортсменів у дії значно ускладнює структуру дій, особливо у командних видах спорту, де одночасно можуть брати участь до 6-15 гравців.

Таким чином, запропонована класифікація враховує, що чим більше гравців виконує дію, тим складнішим стає завдання для аналізу й вивчення динаміки взаємодії, особливо в контексті командних видів спорту. Такий поділ індивідуальних дій дозволяє детальніше розглянути кожен тип рухів, що допомагає при навчанні алгоритмів розпізнавання, а також є основою для подальших досліджень у сфері аналізу поведінки й динаміки рухів.

Для індивідуальних видів спорту індивідуальні дії є типовими і часто взаємодіють з предметами, наприклад, у художній гімнастиці відбувається взаємодія з обручем, м’ячем, булавами або стрічкою. У командних видах спорту можуть бути задіяні всі комбінації дій та видів діяльності.

В даній роботі визначимо, гімнастичний елемент це заздалегідь визначений набір фізичних рухів, які спортсмен виконує за певний час для отримання балів.

Таким чином, бачимо значне різноманіття у підходах до класифікацій дій у спорті за наборами ознак, що потребує відповідних методів і моделей їх розпізнавання.

1.3 Аналіз застосування методів і моделей інтелектуального аналізу дій спортсменів

HAR у спорті є завданням керованого навчання на спортивних даних. Воно починається зі збору та анотування даних для завдання, попередньої обробки, наприклад, видалення цифрового шуму, та виокремлення ознак.

Етап виокремлення ознак в HAR може дозволити ідентифікувати рух об'єкта в сцені, але ці дескриптори не пояснюють дії. Тому класифікація за допомогою методів ML, таких як машина опорних векторів (SVM), використовується в багатьох випадках завдяки її швидкому виконанню [17]. Інші методи ML, які автори використовують рідше, це алгоритм K-найближчого сусіда [18], алгоритм K-середніх [19] тощо.

Дії в HAR також можуть бути класифіковані за допомогою різних методів на основі DL, включаючи автоматичне вилучення ознак із зображень, опис і класифікацію.

На сьогодні найбільшого поширення набули методи на основі DL [20]. Хоча DL усуває фазу ручного вилучення ознак, все одно потрібен великий набір даних для вивчення великої кількості параметрів прихованих шарів, а також більше обчислювальних ресурсів і прискорювачів, таких як графічні процесори.

У роботі [21] дослідники вивчали можливості використання згорткових нейронних мереж для розпізнавання типу тенісних ударів. Вони порівнювали мережі Inception-v3 та MobileNet. Основна увага була зосереджена на розпізнаванні таких дій: фаза підготовки до удару ззаду, удар ззаду, фаза підготовки до удару спереду, удар спереду і відсутність удару. Мережа

MobileNet досягла кращих результатів.

У роботі [22] завданням було розпізнати удар в бадмінтоні на відеозаписах, що транслювалися, за допомогою попередньо навчених методів CNN. Вивчалися удари та інші дії в бадмінтоні. Перший експеримент вивчав продуктивність чотирьох існуючих попередньо навчених моделей: AlexNet, GoogleNet, VGGNet-16 і VGGNet-19, розпізнаючи п'ять дій. Результати показують, що попередньо навчена модель AlexNet має найвищу точність розпізнавання. Другий експеримент - це дослідження продуктивності двох різних попередньо навчених моделей: AlexNet і GoogleNet, для розпізнавання тільки ударних і не ударних дій. Результати показують, що попередньо навчена модель GoogleNet демонструє найкращі показники при розпізнаванні ударів.

У роботі [23] використовуючи той самий набір даних із бадмінтону для вилучення ознак використано нейронні мережі AlexNet і GoogLeNet, щоб порівняти продуктивність цих двох моделей. Потім ознаки були класифіковані за допомогою SVM. Результати показали, що екстрактор ознак, який використовує модель AlexNet CNN, має найкращу продуктивність.

Згорткові нейронні мережі застосовуються не лише для класифікації зображень, але й для багатьох інших задач: сегментації, розпізнавання об'єктів, генерації зображень, аналізу відео, медичної діагностики і навіть автономного водіння. Їхня гнучкість і здатність виявляти складні залежності роблять згорткові нейронні мережі важливим інструментом у багатьох галузях науки і техніки.

Останнім часом значну популярність також здобули моделі трансформерів, які демонструють високу ефективність у завданнях обробки послідовностей даних, зокрема, в обробці природної мови та комп'ютерному зорі. Ці моделі здатні краще враховувати довгострокові залежності та паралельно обробляти всі елементи послідовності.

Крім того, для підвищення точності та ефективності моделей часто застосовуються методи навчання та до налаштування попередньо

натренованих моделей на специфічних наборах даних. Це дозволяє ефективно використовувати знання, отримані на великих загальних наборах даних, і адаптувати їх до певної предметної області.

Ще одним важливим аспектом є інтерпретованість та зрозумілість моделей машинного навчання, особливо в критично важливих сферах, таких як медицина. Тому активно розвиваються методи візуалізації та пояснення рішень моделей, що дозволяє краще зрозуміти їх поведінку та обґрунтованість прийнятих рішень.

Однією з таких мереж, яка стала проривом у класифікації зображень і основою нових методів глибокого навчання, такі як VGGNet [24] і AlexNet [25]. Перші спроби використання таких мереж датуються 1980-ми роками. Yann LeCun разом з колегами розробив першу архітектуру CNN, яка отримала назву LeNet, для розпізнавання рукописних цифр на зображеннях. Після цього, протягом двох десятиліть CNN не отримували великого розвитку через обмеження обчислювальних ресурсів.

Також виокремимо ще одну множину інтелектуальних моделей які використовуються при розпізнаванні спортивних дій, це моделі на основі нейромереж довгої-короткої пам’яті - LSTM.

Результати представлені в роботі [26] спрямовані на розпізнавання дрібнозернистих дій у тенісі за допомогою спеціалізованої глибокої нейронної архітектури. У своїй моделі автори представляли відео у вигляді послідовностей ознак, які отримували за допомогою нейронної мережі Inception, попередньо навченої на незалежному наборі даних. Отримані ознаки використовувались як вхід для трирівневої мережі LSTM, яку автори застосовували для класифікації тенісних дій. Експерименти проводилися на наборі даних THETIS, що містить 12 основних тенісних ударів, що дозволило оцінити точність розпізнавання тенісних дій.

Розширення методів розпізнавання тенісних ударів було запропоновано в роботі [26], де для задачі класифікації 3D-ударів використовувалася зважена LSTM. У цій моделі також застосовувалась нейронна мережа Inception для

витягнення локальних просторових ознак з кожного кадру відео. Далі до моделі вводився зважений LSTM-декодер для обробки послідовної історичної інформації вихід якої потім переходить до повністю пов'язаного шару з класифікатором SoftMax, який присвоює ймовірності виконаним діям. Оцінка ефективності також проводилася на наборі даних THETIS, і результати підтвердили високу продуктивність методу при розпізнаванні складних спортивних дій у тенісі.

У роботі [27] автори запропонували підхід до розпізнавання тенісних дій, який використовує глибоке навчання з модулями канальної та просторової уваги, що значно підвищує точність аналізу послідовностей кадрів. Модель використовує CNN для вилучення візуальних ознак з відео, які потім проходять через спеціальні модулі уваги, що підсилюють найбільш релевантні ознаки. Потім перетворені карти ознак передаються в двонаправлену LSTM-мережу, вихідні дані якої обробляються повністю зв'язаним шаром з класифікатором SoftMax для визначення ймовірності виконуваної дії. Цей підхід був перевірений на шести діях тенісистів з набору даних THETIS, де отримав багатообіцяючі результати.

Робота дослідників у спортивних дисциплінах, таких як теніс, бадмінтон та гандбол, показує, що успішне розпізнавання спортивних рухів потребує різних підходів до обробки відеопослідовностей, залежно від типу дій, кількості залучених об’єктів та умов освітлення.

Наприклад, у тенісі та бадмінтоні (роботи [21, 22, 28]) моделі на основі згорткових нейронних мереж (CNN), таких як Inception та MobileNet, продемонстрували ефективність у вилученні локальних просторових ознак, особливо для розпізнавання чітко виражених рухів, таких як удари. Для художньої гімнастики, де рухи є складнішими, застосування моделей, які поєднують CNN та рекурентні мережі LSTM (як у [27]), є перспективним. CNN добре витягують просторові ознаки, тоді як LSTM або двонаправлені LSTM, що враховують послідовність кадрів, дозволяють відстежувати часове розгортання рухів, як, наприклад, у випадках з дриблінгом або передачами

м'яча. Моделі зі зваженим LSTM підвищують точність, оскільки враховують історичну інформацію дії, що може бути важливим для розпізнавання послідовностей плавних та безперервних рухів у гімнастиці.

Для роботи з художньою гімнастикою важливо розробити підхід, що ефективно об’єднує просторові ознаки рухів з їхньою часовою послідовністю. Це включає роботу зі складними індивідуальними діями, що виконуються із взаємодією з об’єктами. Як показують приклади інших спортивних дисциплін, застосування багаторівневих моделей на основі CNN-LSTM або з компонентами просторової уваги може підвищити точність класифікації, що є необхідним для обробки таких детальних і варіативних дій, як у художній гімнастиці.

1.4 Огляд наборів даних дій спортсменів для інтелектуального аналізу

Для навчання та оцінки методів HAR необхідні розмічені дані. Для задач HAR загалом традиційно доступними публічними наборами даних є KTH [29] та Weizmann [30], які містять різноманітні дії, деякі з яких можуть бути пов’язані зі спортивною сферою. Вони були представлені на початку 21-го століття.

Порівняно з сучасними наборами даних, вони містять невелику кількість класів дій і скромну кількість зразків, записаних у лабораторних умовах. Набір даних KTH складається лише з шести класів (ходьба, біг, бокс, махання руками і плескання в долоні), які виконують 25 осіб кілька разів за чотирма різними сценаріями.

На противагу цьому, такі набори, як HACS [31] і Kinetics 700-2020 [32] мають значно більше класів, значно більше даних і були записані в реалістичних умовах. Наприклад, набір Kinetics – це великий набір даних, з 400/600/700 класами людських дій, залежно від версії, що містить вручну позначені відео, завантажені з YouTube. HACS – це великий набір даних для ідентифікації та часової локалізації людських дій, зібраних з веб-відео. Для HAR у спортивній сфері існує шість наборів даних публічних дій, які

включають різні дії з різних видів спорту: UCF Sports Action Data Set [33], Olympic Sports Dataset [34], Sports-1M [35], FineGym [36], SVW [37] та MultiSports [38].

Набір даних спортивних дій UCF, показаний на рисунку 1.2, складається з різних маркованих дій, зібраних з різних видів спорту, які зазвичай демонструються на телевізійних каналах, разом з анотаціями людей, показаних жовтим кольором у рамочці. Набір даних містить 150 послідовностей з роздільною здатністю 720 × 480 і частотою кадрів 10 кадрів в секунду. Загальна тривалість набору даних становить 958 с, а середня довжина послідовності - близько 6,39 с. Набір даних включає десять дій у різних середовищах (спортивний зал, спортивне поле, природа тощо): Пірнання, замах у гольфу, удар ногою, підйом, їзда на коні, біг, катання на скейтборді, вправи на спортивному коні, вправи на різновисоких брусах та ходьба. Кількість послідовностей не однакова для кожного класу, наприклад, дія піднімання містить мінімум 6 послідовностей, а дія ходьби - максимум 22 послідовності. Крім того, окремі дії є короткими, наприклад, удари ногами, порівняно з ходьбою або бігом, які є відносно довшими.

Рисунок 1.2. Приклад відеокадрів різних дій з набору даних UCF Sports Action Data Set

Набір даних про олімпійські види спорту у поточній версії містить відеозаписи спортсменів, які займаються 16 різними видами спорту. Він містить 50 відео з кожного з 16 видів спорту: стрибки у висоту, стрибки в довжину, потрійні стрибки, стрибки з жердиною, метання диска, метання молота, метання списа, штовхання ядра, баскетбольний кидок, боулінг, подачі

у тенісі, стрибки у воду з вишки, стрибки у воду з трампліну, важка атлетика, поштовх та гімнастичні стрибки. Відео у наборі даних взяті з YouTube і містять реалістичні сцени. Відео анотовано за допомогою Amazon Mechanical Turk, а кадри характеризуються рухом камери, зміною відстані об'єкта від камери та сильними оклюзіями.

На рисунку 1.3 показано приклади кадрів декількох класів дій з набору даних, з яких видно, що дія відбувається в спортивному залі або на спортивних майданчиках, залежно від того, яка дія спостерігається. З точки зору складності, всі дії в наборі є індивідуальними діями. Точніше, є взаємодії, такі як подача у тенісі та боулінг, і складні дії, такі як стрибки у висоту, довжину, потрійний стрибок і т.д. Наприклад, послідовності зі стрибків у довжину показують, як спортсмен спочатку стоїть на місці, готуючись до стрибка, потім біжить, стрибає, приземляється і, нарешті, встає.

Рисунок 1.3. Приклад відеокадрів з набору даних про олімпійський спорт

Набір даних Sports-1M містить 1 133 158 URL-адрес відео на YouTube, автоматично анотованих 487 спортивними мітками за допомогою YouTube Topics API. Класи впорядковані в таксономії, яка містить внутрішні вузли, такі як "Водні види спорту", "Командні види спорту", "Зимові види спорту", "Спортивні ігри з м'ячем", "Бойові види спорту", "Спортивні ігри з тваринами", і загалом стає дрібнозернистою на рівні листків. Дії відбуваються у спортзалах, басейнах, спортивних залах, на спортивних майданчиках, дорогах, у лісі, на льодових ковзанках, гірськолижних схилах, у небі, у парках та інших місцях, з якими люди стикаються щодня. Деякі приклади спортивних сцен та відповідні їм позначення показані на рисунку 1.4. Основна увага приділяється розрізненню таких видів спорту, як теніс, хокей, плавання, водне

поло, катання на лижах тощо.

Рисунок 1.4. Приклад відеокадрів з набору даних Sports-1M

Набір даних "Спортивне відео в реальному житті" - це складний набір даних, знятий користувачами-аматорами мобільного додатку Coach's Eye за допомогою своїх смартфонів під час занять спортом або перегляду гри. Набір даних містить 4100 відео і складається з 30 спортивних категорій та 44 видів спорту, таких як хокей, стрибки в довжину, стрибки з жердиною, веслування, біг, штовхання ядра, фігурне катання, лижі, футбол, плавання, теніс, волейбол та інші, рисунок 1.5.

Рисунок 1.5. Приклад відеокадрів, що містять різні категорії з набору даних Sports Videos in The Wild

Через непрофесійну зйомку відео виникають проблеми з вібрацією та рухом камери, оклюзією, зміною точки зору та поганим освітленням. Крім

того, через недосконалу практику гравців-аматорів спостерігається різне виконання дії в нерепрезентативному середовищі, як правило, за межами спортивного майданчика, фонове захаращення.

MultiSport - це новий реалістичний багато персональний набір даних про просторово-часові локалізовані спортивні дії. Дані зібрані для чотирьох різних видів спорту: аеробної гімнастики, баскетболу, футболу та волейболу, де дії кожного виду спорту детально проаналізовані та анотовані, що дає 66 класів дій, таких як пас, ведення та дриблінг у баскетболі. З 3200 відеокліпів 37701 екземплярів дій прокоментовано точними і щільними анотаціями в просторовій і часовій областях, що представляють 902 тис. обмежувальних рамок навколо діючих гравців. Набір даних є масштабним і якісним, але водночас дуже складним через велику кількість гравців у сцені та дуже схожі між собою дрібнодисперсні дії, а фон є набагато менш характерним і не може надати достатньої інформації для розпізнавання дрібнодисперсних дій.

Незважаючи на те, що ці шість наборів даних використовуються для розпізнавання різних спортивних дій, таких як боулінг, плавання або біг, існує безліч відео зі спортивними сценами, матчами та змаганнями, які можна завантажити з веб-сайту. Однак вони здебільшого не позначені тегами і не підготовлені для машинного навчання, тому підготовка відповідних спортивних сцен для машинного навчання займає багато часу. З цієї причини, щоб навчити модель HAR для певного виду спорту, дослідники дуже часто змушені створювати власні набори даних, оскільки не існує репрезентативного набору даних, який можна було б використовувати як еталон.

У роботі [39] дослідники створили спеціалізований набір даних, що складається із записів тренувань з гандболу, проведених як у закритих приміщеннях, так і на відкритих майданчиках. Для зйомок використовували камери з роздільною здатністю Full HD, розташовані на різних висотах і позиціях по периметру спортивного майданчика, що забезпечувало різноманітність перспектив і дозволяло захопити детальні просторово-часові

ознаки дій спортсменів. У цьому наборі даних представлено понад 700 послідовностей, що включають як індивідуальні дії (наприклад, біг — базова дія або кидок у ворота як взаємодія з об'єктом), так і складні та спільні дії, такі як дриблінг або передача м’яча між двома гравцями.

Набір даних характеризується високою складністю через низку зовнішніх чинників. Так, на відкритих майданчиках дослідникам довелося враховувати природне освітлення та сонячне світло, що створювало різноманітні тіні і відблиски, тоді як записи у приміщеннях велися за умов штучного освітлення, яке вимагало додаткових корекцій і налаштувань камер. Крім того, на фоні присутні інші гравці та елементи, що ускладнюють завдання розпізнавання дій, роблячи цей набір цінним ресурсом для моделювання алгоритмів комп’ютерного зору та обробки спортивних відеоданих у складних умовах.

Набір даних FineGym створено на основі 708 годин відеороликів спортивної гімнастики, розміщених на YouTube, де понад 95% з них мають високу роздільну здатність, 720 і 1080 пікселів. Усі відео у FineGym - це офіційні записи змагань найвищого рівня з жіночої спортивної гімнастики, де дії багаті та різноманітні з точки зору точок зору та поз. Крім того, на відміну від інших наборів даних, де фон може допомогти розрізнити дії, всі приклади у FineGym мають відносно однакове тло. Цей набір даних відрізняється від вищезгаданих наборів тим, що надає часові анотації як на рівні дій, так і на рівні піддій з трирівневою семантичною ієрархією: подія, набір та елемент, рисунок 1.6.

Автори проаналізували чотири гімнастичні вправи: вправи на колоді, брусах, опорному стрибку та вільні вправи. Ці вправи ідентифіковані в кожному відео, і вони розкладаються за допомогою вручну побудованих дерев рішень на під дії. Наприклад, вправа на колоді анотується як послідовність елементарних піддій, отриманих з п'яти наборів, де під дія в кожному наборі додатково анотується за допомогою чітко визначених міток класу.

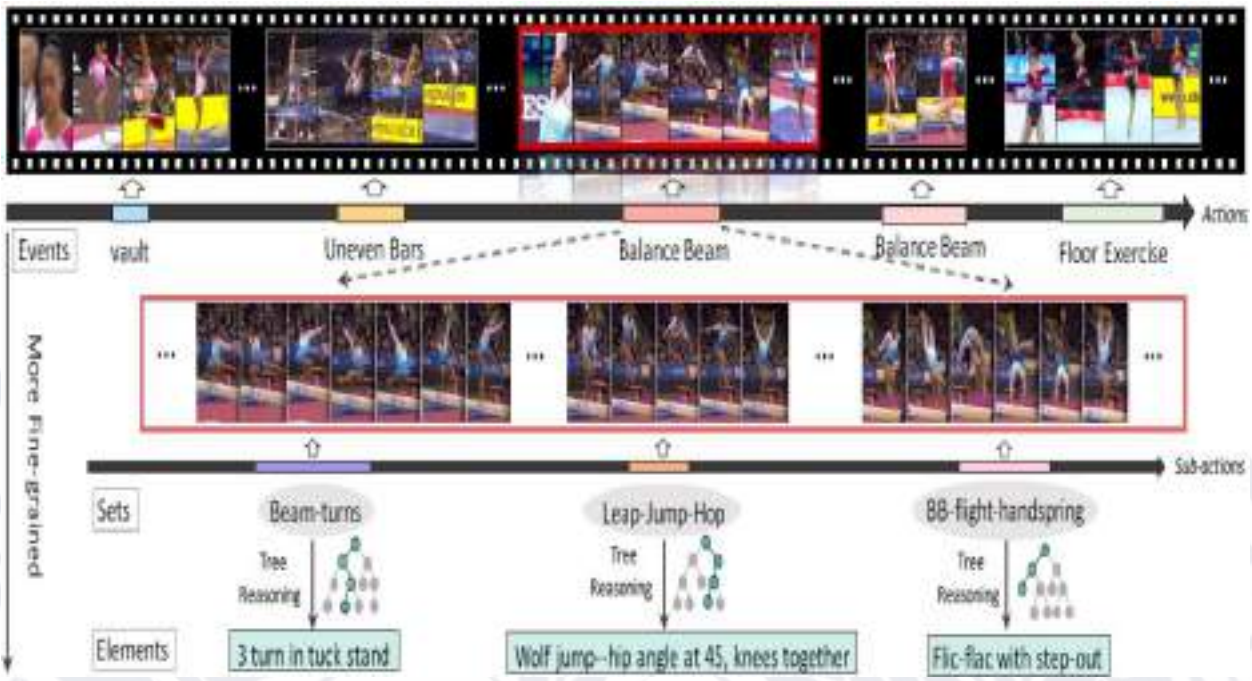

Рисунок 1.6. Трирівнева семантична ієрархія у FineGym.

Для того, щоб врахувати всі комбінації категорій елементів і полегшити поповнення бази даних, було визначено 530 категорій, з яких 354 мають принаймні один екземпляр. Кількість екземплярів у кожній категорії елементів коливається від 1 до 1 648, а загальна кількість зразків з урахуванням усіх класів сягає 32 697. Через відсутність екземплярів для деяких категорій елементів доцільніше використовувати версію Gym288 з 288 класами або більш збалансовану версію Gym99 з 99 класами.

Слід підкреслити, що для практичної оцінки методів HAR у спорті необхідно використовувати набір даних, який реалістично описує дії в цьому виді спорту з усіма можливими труднощами і є достатньо великим, щоб вивчити параметри методу і побудувати надійну модель. Однак, якщо такого набору немає у відкритому доступі, підготовка такого набору для навчання є трудомістким, дорогим і виснажливим процесом, оскільки він вимагає запису в природних умовах, а також подальшого маркування та підготовки записів.

З аналізу доступних наборів даних для вивчення спортивних дій можна зробити висновок, що хоча вони забезпечують значну кількість даних та охоплюють широкий спектр видів діяльності, їхня придатність для завдань, пов'язаних із аналізом послідовностей кадрів гімнастичних поз у художній

гімнастиці, є обмеженою. Набори даних, такі як KTH та Weizmann, хоч і є історично важливими для розвитку галузі розпізнавання людських дій, не відповідають сучасним вимогам, оскільки вони містять малу кількість класів, обмежений обсяг зразків і не включають специфічних гімнастичних поз. Їхній фокус переважно зосереджений на простих діях, таких як ходьба чи біг, які суттєво відрізняються від складних і витончених рухів у художній гімнастиці. Сучасніші набори даних, як-от HACS та Kinetics, пропонують більше різноманіття та реалістичність завдяки великій кількості класів і прикладів. Проте їхня основна мета — охопити широкий спектр загальнолюдських дій, часто без конкретного акценту на деталізованих спортивних рухах або специфічних позах. У художній гімнастиці, де ключове значення має аналіз складних рухів і балансів, таких як елементи рівноваги та обертання, ці набори даних є недостатніми через відсутність релевантних прикладів. Хоча спеціалізовані набори, такі як UCF Sports Action та Olympic Sports Dataset включають дані з різних видів спорту, вони, як правило, фокусуються на більш загальних аспектах дій спортсменів, а не на деталізованих рухах і позах. Набір даних FineGym фокусуються лише на елементах одного виду спорту, а саме на спортивній гімнастиці. Це робить їх лише частково корисними для завдань художньої гімнастики. У підсумку, ці публічні набори даних можуть бути лише початковою базою, але для точного аналізу гімнастичних поз у художній гімнастиці необхідно створювати спеціалізовані набори даних, які б враховували специфіку цього виду спорту.

1.5 Інтелектуальні технології обробки послідовностей кадрів

Покращення апаратного забезпечення з більшою обчислювальною потужністю і легкодоступними алгоритмами машинного навчання з відкритим кодом, протестованими і застосованими в багатьох додатках, сприяє розвитку CV. Завдяки таким популярним бібліотекам, як Tensorflow [40] і OpenCV [41], більше не потрібно починати проекти CV з нуля, що, в свою чергу, призводить до зменшення часу, витраченого на реалізацію.

Важливою концепцією в галузі обробки послідовностей кадрів є поняття оптичного потоку, яке описує зміну положення об'єктів між кадрами. Оптичний потік можна використовувати для відстеження руху об'єктів, виявлення змін у сцені та розпізнавання дій.

Обробка та аналіз послідовностей кадрів є важливим завданням у галузі комп'ютерного зору та машинного навчання. Воно знаходить застосування в різноманітних сферах, таких як відеоспостереження, розпізнавання дій, аналіз руху, медична діагностика та багато інших. У контексті даної роботи, акцент буде зроблено на застосуванні цих технологій для аналізу гімнастичних поз на послідовностях кадрів.

Інтелектуальні технології обробки послідовностей кадрів спираються на методи машинного навчання та глибокого навчання, які дозволяють автоматично виявляти закономірності та ознаки в даних. Ці методи здатні навчатися на великих наборах даних і виявляти складні залежності, що важко або неможливо описати аналітично.

Одним з ключових завдань при обробці послідовностей кадрів є виокремлення ключових кадрів, які містять найбільш важливу інформацію про події або явища, що відбуваються. Це дозволяє значно скоротити обсяг даних для подальшого аналізу та підвищити ефективність обробки.

Розвиток інтелектуальних технологій обробки послідовностей кадрів тісно пов'язаний з доступністю обчислювальних ресурсів та наявністю якісних наборів даних для навчання моделей. Поява потужних графічних процесорів та хмарних обчислювальних ресурсів значно прискорила прогрес у цій галузі, роблячи можливим навчання складних моделей на величезних наборах даних.

Таким чином, оцінювання виконання елементів у художній гімнастиці є складним та комплексним завданням. Кожен елемент, починаючи від артистизму до елементів тіла та роботи з предметом, піддається суворому аналізу. Складність полягає в тому, що оцінка подібних елементів піддається суб'єктивним інтерпретаціям і вимагає від експертів високого рівня професіоналізму.

У попередній роботі [3] виконано дослідження проблем та класифікації гімнастичних елементів рівноваги, що виконуються спортсменками з художньої гімнастики, на підставі кадрів.

У цій роботі розглядається класифікація гімнастичних елементів повороту в динаміці на підставі аналізу послідовності кадрів. У цьому контексті розробка інтелектуальних методів класифікації гімнастичних елементів по відео може значно покращити об'єктивність та ефективність процесу оцінювання у художній гімнастиці.

Сформулюємо деталізовані завдання дослідження наступним чином:

- виконати збір відео прикладів виконання гімнастичних елементів та провести попередню обробку для формування набору даних для аналізу послідовностей кадрів гімнастичних поз;
- виконати вибір нейромережевих моделей типу CNN, LSTM, ConvLSTM і LRCN для аналізу послідовностей кадрів гімнастичних поз;
- виконати структурну та параметричну ідентифікацію нейромережевих моделей типу CNN, LSTM, ConvLSTM і LRCN;
- провести чисельне дослідження обраних нейромережевих моделей аналізу послідовностей кадрів гімнастичних поз;
- виконати програмну реалізацію інформаційної технології інтелектуального аналізу послідовностей кадрів гімнастичних поз.

1.6 Висновки за розділом 1

1. Виконано огляд основних понять, завдань і проблем інтелектуального аналізу послідовностей кадрів дій людини. Зроблено висновок, що існує велике різноманіття людських дій і умов їх спостереження, які підлягають аналізу по відео. В той же час не існує універсальних методів, моделей і інформаційних технологій щодо їх аналізу.

2. Встановлено значне різноманіття у підходах до класифікацій дій у спорті за наборами ознак, що потребує відповідних методів і моделей аналізу.

3. Зроблено припущення, що комбінація CNN, для вилучення ознак, і LSTM, для послідовної обробки кадрів, може бути ефективним у аналізі послідовностей кадрів гімнастичних елементів.

4. З аналізу доступних наборів даних для вивчення спортивних дій можна зробити висновок, що хоча вони забезпечують значну кількість даних та охоплюють широкий спектр видів діяльності, їхня придатність для завдань, пов'язаних із аналізом послідовностей кадрів гімнастичних поз у художній гімнастиці, є обмеженою.

5. Таким чином, оцінювання виконання елементів у художній гімнастиці є складним та комплексним завданням. Кожен елемент, починаючи від артистизму до елементів тіла та роботи з предметом, піддається суворому аналізу. Складність полягає в тому, що оцінка подібних елементів піддається суб'єктивним інтерпретаціям і вимагає від експертів високого рівня професіоналізму.

РОЗДІЛ 2
МЕТОД НЕЙРОМЕРЕЖЕВОЇ КЛАСИФІКАЦІЇ ПОСЛІДОВНОСТЕЙ КАДРІВ ГІМНАСТИЧНИХ ПОЗ

2.1 Визначення архітектури для аналізу послідовностей кадрів гімнастичних поз на основі згорткових нейронних мереж

Згорткові нейронні мережі є однією з найпопулярніших і найефективніших архітектур для обробки зображень та відео. Вони були спеціально розроблені для розпізнавання образів і відмінно працюють з даними, що мають дво- або тривимірну структуру, як-от зображення чи відео. На початку мережі згорткові шари використовуються для вилучення ознак, а в кінці повністю з'єднані шари - для класифікації.

Найпопулярнішими моделями, що використовуються для реалізації HAR у спорті, є згорткові нейронні мережі (Convolution Neural Networks, CNN), а також довга короткочасна пам'ять (Long Short-Term Memory, LSTM) [42].

Шари згортки є основою згорткових нейронних мереж. У цих шарах фільтри, або їх ще називають ядра, проходять по зображенню, здійснюючи операцію згортки, тобто скалярне множення фільтру на невелику ділянку зображення. Це дозволяє виділяти важливіші характеристики, наприклад, краї або текстури. Один із ключових аспектів згорткових нейронних мереж — це спільне використання ваг у фільтрах, що значно зменшує кількість параметрів мережі.

Шари об'єднання використовуються для зменшення розмірності простору ознак, що дозволяє зменшити обчислювальні витрати. Під час об’єднання найчастіше обирають функцію максимального об’єднання, яка вибирє максимальне значення серед групи пікселів. Це дозволяє зберігати найбільш релевантні ознаки, зменшуючи при цьому кількість обчислень і запобігаючи перенавчанню.

Після кожного шару згортки та об'єднання зазвичай застосовують нелінійні активаційні функції, такі як зрізаний лінійний вузол (Rectified Linear

Unit, ReLU), що допомагають нейронній мережі навчитися моделювати складні залежності.

Зрізаний лінійний вузол особливо популярний завдяки своїй простоті та ефективності, оскільки дозволяє моделі швидко сходитися.

Згорткові нейронні мережі навчаються методом зворотного поширення помилки, де ваги кожного фільтру оновлюються залежно від градієнтів помилок. Використання градієнтного спуску дозволяє поступово мінімізувати функцію втрат, покращуючи точність мережі.

Для ефективного навчання згорткових нейронних мереж зазвичай використовують великі набори даних, оскільки великі моделі можуть швидко перенавчитися. Для уникнення проблеми перенавчання у згорткових нейронних мережах використовують такі техніки, як регуляризація, метод випадкового вимикання нейронів і розширення даних, яке включає додаткові трансформації вхідних зображень, такі як обертання або масштабування. Архітектура простої CNN представлена на рисунку 2.1.

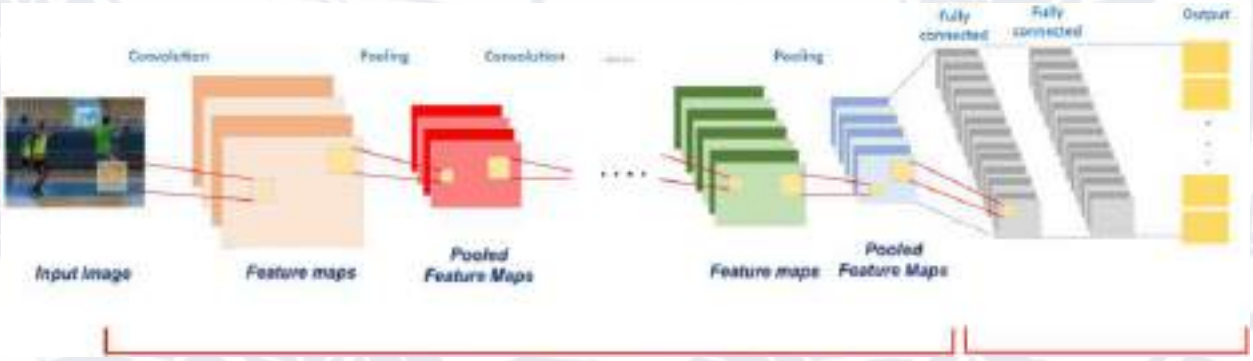

Рисунок 2.1. Проста архітектура CNN.

Ця мережа на рисунку 2.1 є прикладом двовимірної згортки, але тривимірна згортка часто може бути використана для виокремлення ознак при розпізнаванні дій [43]. Одна 3D-згортка може одночасно фіксувати просторово-часову інформацію про поведінку дій на відео, тоді як одна 2D-згортка потребує певної допомоги для фіксації часової інформації [44]. Якщо вхідні дані для дії складаються з декількох кадрів, 2D згортка накопичує всі результати з різних відеокадрів. Вона виводить одне зображення навпроти 3D-згортки, яка зберігає часову інформацію у всіх різних відеокадрах і генерує

вихідний об'єм, колекцію кадрів [45], як показано на рисунку 2.2. Подібно до об'єднання шарів у 2D- ШНМ, 3D-об'єднання створює один вихідний піксель для пікселя того самого кольору, який виконується разом із сусідніми відеокадрами.

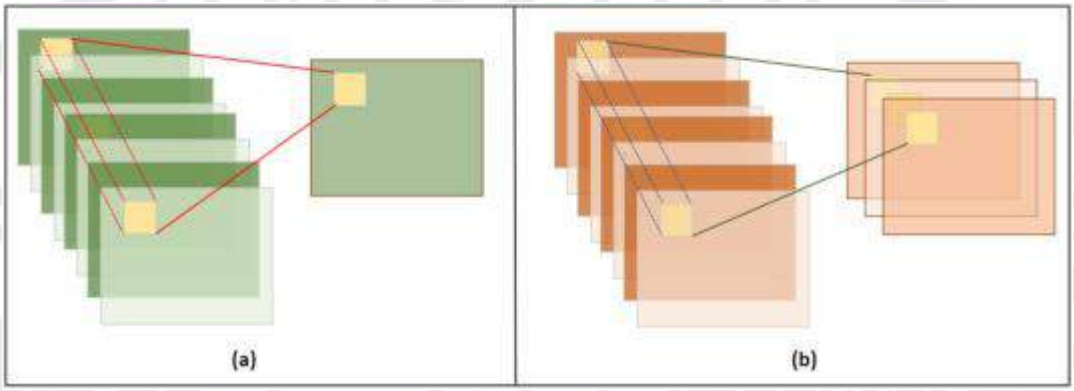

Рисунок 2.2. Порівняння 2D-згортки (а) та 3D-згортки (б).

Далі наведено приклад обрахунку вхідного та вихідного сигналу згорткового шару при зменшені масштабу у два рази. Нехай ν — позиція в області з'єднання, $\nu = (\nu_x, \nu_y)$, K_I , — кількість площин клітин у вхідному шарі I (для RGB-зображень 3), K_{s_l} — кількість площин клітин у шарі зменшення дискретизації S_l, K_{c_l} — кількість площин клітин у згортковому шарі C_l, A_l — площа зв'язування площини шару S_l, $\widehat{L}$ — кількість шарів згортки (або зменшення дискретизації), $\breve{L}$ — кількість повністю з'єднаних шарів .

1. $l = 1.$

2. Розрахунок вихідного сигналу для згорткового шару:

$$u_{c_l}(m,i) = f_{c_l}(h_{c_l}(m,i)),\ m \in \{1,\dots,N_{c_l}\}^2,\ i \in \overline{1,K_{c_l}},$$

$$h_{c_l}(m,i) = \begin{cases} b_{c_1}(i) + \sum_{k=1}^{K_I} \sum_{\nu \in A_1} w_{c_1}(\nu,k,i)x(m+\nu,k), & l=1 \\ b_{c_l}(i) + \sum_{k=1}^{K_{s_{l-1}}} \sum_{\nu \in A_{l-1}} w_{c_l}(\nu,k,i)u_{s_{l-1}}(m+\nu,k), & l>1 \end{cases},$$

де $w_{c_1}(\nu,k,i)$ - вага зв'язку від ν-го положення в області k-ї площини клітин вхідного шару I до *i*-ї площини клітин згорткового шару C_1, $w_{c_l}(\nu,k,i)$ - вага зв'язку від ν-го положення в області k-ї площини клітин що знижує дискретизацію шару S_{l-1} до *i*-ї площини клітин згорткового шару C_l, $u_{c_l}(m,i)$ - вихід клітини в m-ю позицію в *i-ї* площині клітин згорткового шару C_l, f_{c_l} - є функцією активації нейронів згорткового шару C_l.

3. Обчислення вихідного сигналу для шару пониження дискретизації зі зменшенням масштабу в два рази:

$$u_{s_l}(m,k)=\frac{1}{4}\sum_{\upsilon\in\{0,1\}^2}u_{c_l}(2m+\upsilon,k),\ m\in\{1,\dots,N_{s_l}\}^2,\ k\in\overline{1,K_{s_l}},$$

де $w_{s_l}(k,k)$ – ваги зв'язків від *k*-ї площини клітин згортокового шару C_l до *k*-ї площини клітин для шару пониження дискретизації S_l, $u_{s_l}(m,k)$ – вихід клітини у m-ї позиції у *k*-ї площині клітин для шару пониження дискретизації S_l.

2.2 Визначення архітектури для аналізу послідовностей кадрів гімнастичних поз на основі рекурентних нейронних мереж

Довга короткочасна пам'ять (англ. long short-term memory, LSTM) — це різновид рекурентних нейронних мереж, які здатні зберігати та опрацьовувати довготривалі залежності в послідовних даних[46]. На відміну від стандартних рекурентних нейронних мереж, які мають труднощі з передачею інформації через багато кроків часу, модель LSTM може ефективно працювати з тривалими послідовностями завдяки спеціальній структурі внутрішніх осередків пам'яті. Архітектура LSTM була запропонована у 1997 році Sepp Hochreiter і Jürgen Schmidhuber, як рішення для проблеми зникнення градієнтів у стандартних рекурентних нейронних мережех. Це стало можливим завдяки впровадженню механізму комірок пам'яті та спеціальних воріт, які дозволяють моделі контролювати потік інформації.

LSTM складаються зі спеціальних блоків, які називаються блоками пам'яті, що розташовані в рекурентних прихованих шарах і містять комірки пам'яті. Комірки пам'яті зберігають часовий стан мережі завдяки самозв'язкам. Для керування потоком інформації використовуються різні типи мультиплікативних блоків, які називаються вентилями. Вхідні вентилі контролюють потік вхідних активацій в комірку пам'яті, а вихідні вентилі контролюють вихідний потік активацій комірки в решту мережі. Вентиль забування викликає адаптивне забування або скидання пам'яті комірки шляхом масштабування її внутрішнього стану перед додаванням його як вхідного сигналу до комірки через самоповторюване з'єднання комірки [47], рисунок 2.3. Це забезпечує високий рівень контролю над потоком інформації в мережі та дозволяє ефективно працювати з даними на різних часових шкалах. Однак, такий послідовний потік не дуже добре використовує сучасні графічні процесори, призначені для паралельних обчислень.

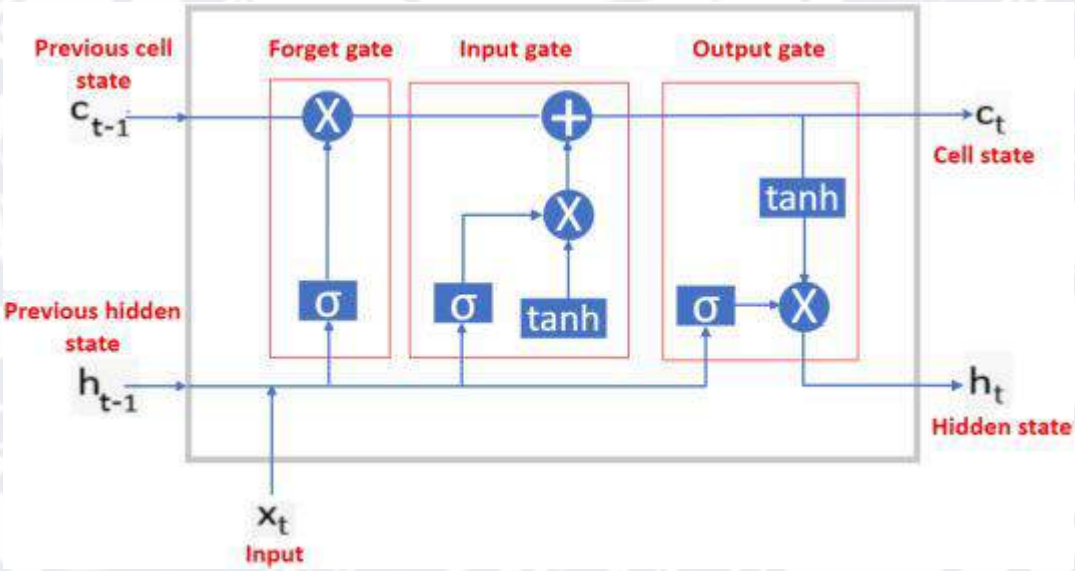

Рисунок 2.3. Зображення простого LSTM.

Як і інші нейронні мережі, моделі LSTM навчаються за допомогою зворотного поширення помилки, але з урахуванням часових залежностей. Їхня здатність уникати проблеми згасання градієнтів робить їх надзвичайно ефективними для задач, де важливо враховувати контекст у часі, наприклад, у задачах обробки природної мови або аналізу часових рядів. Одна з головних проблем стандартних рекурентних нейронних мереж — це проблема згасання

або вибуху градієнтів під час навчання на довгих послідовностях. Градієнти, які використовуються для оновлення ваг мережі, можуть ставати надто малими або занадто великими, що ускладнює коректне навчання моделі. Це робить стандартні рекурентні нейронні мережі погано придатними для задач, де необхідно враховувати довготривалі залежності в послідовностях, наприклад, в аналізі тексту чи відео. Саме тому модель LSTM була розроблена як рішення цієї проблеми.

Оскільки модель LSTM може обробляти послідовні дані, їх часто застосовують для прогнозування часових рядів, де важливо враховувати як короткотривалі, так і довготривалі залежності між подіями. Наприклад, LSTM моделі широко використовуються для фінансових прогнозів, аналізу кліматичних змін і в медицині для прогнозування стану пацієнтів на основі історичних даних. Моделі LSTM показують чудові результати в задачах обробки природної мови, таких як машинний переклад, розпізнавання мовлення та генерація тексту. Вони здатні "запам'ятовувати" контекст у тексті на різних відрізках часу, що особливо важливо для завдань, де попередні слова чи фрази можуть впливати на сенс наступних. Крім базової версії моделі LSTM, було запропоновано багато покращень, які дозволяють краще вирішувати різноманітні задачі. Наприклад, закриті рекурентні одиниці (Gated Recurrent Units) є спрощеною версією LSTM, де було зменшено кількість воріт, що робить їх менш обчислювально інтенсивними. Також є різні модифікації LSTM для роботи з багатовимірними даними або для паралельної обробки послідовностей.

Модель LSTM, як і інші глибокі нейронні мережі, потребують великих обсягів даних для ефективного навчання. Це пов’язано з тим, що для генералізації потрібна значна кількість прикладів різних сценаріїв. У таких задачах, як обробка відео або мовлення, де послідовності можуть бути дуже довгими, важливо мати доступ до великих і різноманітних наборів даних, щоб забезпечити високоякісні прогнози.

Навчання моделі LSTM є ресурсозатратним процесом, особливо коли

мова йде про довгі послідовності. Для прискорення навчання застосовують кілька технік, таких як паралельне навчання на кількох графічних процесорах, а також використання попередньо натренованих моделей, які можна доопрацювати під конкретні задачі. Іншим підходом є використання оптимізованих бібліотек та фреймворків для роботи з нейронними мережами.

У наступних формулах наведено алгоритм обрахунків під час прогнозування моделі LSTM:

$$y_i^{in}(n-1) = x_i,$$

$$y_j^{gin}(n) = f\left(b_j^{gin} + \sum_{i=1}^{N^{(0)}} w_{ij}^{in-gin} y_i^{in}(n-1)\right),\ j \in \overline{1, N^{(1)}},$$

$$y_j^{g\varphi}(n) = f\left(b_j^{g\varphi} + \sum_{i=1}^{N^{(0)}} w_{ij}^{g\varphi} y_i^{in}(n-1)\right),\ j \in \overline{1, N^{(1)}},$$

$$\tilde{s}_{jv}^{c}(n) = g\left(b_{jv}^{s} + \sum_{i=1}^{N^{(0)}} w_{ijv}^{in-s} y_i^{in}(n-1)\right),\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(1)}},$$

$$s_{jv}^{c}(n) = y_j^{\varphi}(n) s_{jv}^{c}(n-1) + y_j^{gin}(n) \tilde{s}_{jv}^{c}(n),\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(1)}},$$

$$y_j^{gout}(n) = f\left(b_j^{gout} + \sum_{i=1}^{N^{(0)}} w_{ij}^{in-gout} y_i^{in}(n-1)\right),\ j \in \overline{1, N^{(1)}},$$

$$y_{jv}^{c}(n) = y_j^{gout}(n) h(s_{jv}^{c}(n)),\ v \in \overline{1, S_j},\ j \in \overline{1, N^{(1)}},$$

$$y_j^{out}(n) = f\left(b_j^{out} + \sum_{i=1}^{N^{(1)}} \sum_{v=1}^{S_i} w_{ivj}^{c-out} y_{iv}^{c}(n)\right),\ j \in \overline{1, N^{(2)}}.$$

Одна з проблем моделі LSTM, як і інших нейронних мереж, полягає в складності інтерпретації того, що відбувається всередині мережі. Хоча модель може давати точні прогнози, розуміння того, як саме LSTM приймає свої рішення, є непростим завданням.

2.3 Поєднання згорткових та рекурентних нейронних мереж для аналізу послідовностей кадрів гімнастичних поз

Ефективною є гібридна архітектура, де згорткові нейронні мережі використовуються для вилучення ознак із зображень або відео, а модель LSTM - для моделювання залежностей у часі. Така комбінація є надзвичайно ефективною в задачах відео аналізу або розпізнавання дій на послідовностях кадрів. Згорткова нейронна мережа виділяє просторові ознаки, а LSTM дозволяє виявляти часові залежності між ними. У цій роботі буде розглянуто два варіанти поєднання згорткових та рекурентних мереж де на зібраному наборі даних навчаються обидві мережі. Третім підходом буде використання поєднання попередньо навченої моделі MobileNetV2, для вилучення ключових точок скелету, з рекурентною нейронною мережею.

Перша архітектура складається з трьох послідовних шарів, кожен з яких є варіантом нейронної мережі типу ConvLSTM. Ці шари забезпечують здатність моделі не тільки вчитися з просторових ознак, але й зберігати та обробляти тимчасові залежності між кадрами. Блок-схема, що описує архітектуру глибокої нейронної мережі для обробки послідовностей даних зображена на рисунку 2.4.

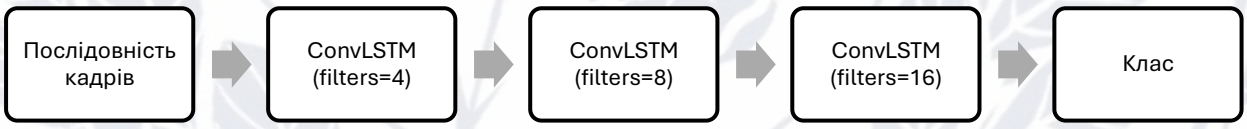

Рисунок 2.4. Блок-схема архітектури мережі на основі ConvLSTM

Кількість фільтрів з кожним шаром збільшується для того щоб вилучити як базові просторові ознаки, та і складніші залежності набору даних для виконання класиіфікації. Така структура дозволить ефективно обробляти послідовності кадрів гімнастичних поз, де важливі як просторові, так і часові закономірності.

Другий варіант моделі складається з послідовності з трьох шарів Conv2D та одного шару LSTM, після чого слідує шар для класифікації (рис. 2.5). Перші три шари є двовимірними згортковими шарами, що дозволить моделі ефективно витягувати просторові ознаки з кожного окремого кадру послідовності. Кожен з цих шарів має фільтри, які визначають різні просторові залежность, такі як краї,

текстури та інші просторові елементи, вхідних зображень. Ці фільтри застосовуються до кожного кадру незалежно, що дозволяє виділяти найбільш значущі просторові ознаки в межах кожного окремого кадру.

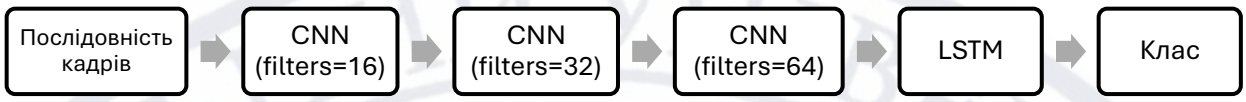

Рисунок 2.5. Блок-схема архітектури мережі на основі LRCN

Таким чином, Conv2D відповідає за обробку просторових характеристик даних, але не враховує часову залежність між кадрами.

Після трьох згорткових шарів, результат передається у LSTM шар, який спеціалізується на обробці часових залежностей. Завдяки цьому, модель може виявляти закономірності і взаємозв'язки, що простягаються на кілька кадрів вперед або назад.

Як і в першій архітектурі, після шару LSTM йде шар, який виконує класифікацію на основі отриманих ознак, що дає змогу моделі здійснювати класифікацію.

Третя варіація архітектури відрізняються тим що під час попередньої обробки набору даних використовується попередньо навчена модель MobileNetV2 для знаходження ключових точок скелету. У якості вхідних даних для навчання моделі LSTM використовується послідовність скелетів спортсмена, де кожна послідовність скелетів відповідає послідовності кадрів у наборі даних (рис.2.6).

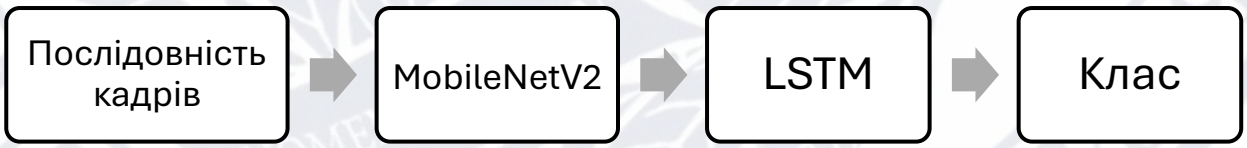

Рис. 2.6. Блок-схема архітектури мережі на основі LSTM

Ключові точки зберігають лише найбільш значущі просторові ознаки, зменшуючи обсяг даних і зосереджуючи модель на найважливіших аспектах. Після отримання послідовності ключових точок, вони передаються в LSTM шар, що дозволяє моделі вивчати часову залежність між ними. Враховуючи, що ключові точки є компактним представленням даних, використання LSTM дає змогу виявляти закономірності не тільки в межах одного кадру, але й по всій

послідовності, аналізуючи зміну цих точок у часі.

Як і в попередніх архітектурах, на кінцевому етапі мережі застосовується шар для класифікації, який здійснює кінцевий прогноз основі навченої інформації про часову та просторову залежність між ключовими точками.

Основна різниця між першою і другою моделлю полягає в тому, як обробляються просторові та часові залежності. У першій моделі використовуються шари ConvLSTM, що є комбінованими шарами для обробки як просторових, так і часових характеристик одночасно. Це дозволяє моделі більш ефективно захоплювати взаємозв'язки між кадрами, застосовуючи згортки і рекурентні механізми в одному кроку.

У другій моделі обробка просторових ознак і часових залежностей розділена: спочатку використовуються Conv2D шари для обробки просторових аспектів кожного окремого кадру, після чого результати передаються до шару LSTM, який обробляє часову залежність між кадрами. Такий підхід може бути менш ефективним у порівнянні з комбінованим ConvLSTM, оскільки кожен кадр обробляється ізольовано, і модель не має змоги ефективно враховувати просторово-часові залежності на ранніх етапах навчання.

Основною відмінністю третьої моделі є спосіб представлення та обробки вхідних даних. На відміну від попередніх моделей, де використовувались початкові кадри, в цій архітектурі акцент робиться на використанні послідовності ключових точок, що суттєво змінює підхід до моделювання просторових і часових залежностей. Замість роботи з усім зображенням, модель фокусується на найважливіших характеристиках, що знижує обсяг даних, роблячи модель більш ефективною в задачах, де важливо розпізнавати зміну положення тіла з плином часу.

Таким чином, третя архітектура є більш спеціалізованою порівняно з першою та другою моделями, оскільки використовує інформацію про ключові точки замість повних кадрів. Це може призвести до покращення ефективності, зокрема для задач, де важлива не стільки детальна текстура зображень, скільки їхня зміна у часі, зокрема в задачах відстеження або розпізнавання руху. Водночас така модель може бути менш універсальною в задачах, що вимагають

глибокого аналізу всього зображення без явних виділених ключових точок.

Таким чином, перша модель з ConvLSTM може бути більш ефективною для задач, де важливі як просторові, так і часові залежності, оскільки ці два аспекти обробляються одночасно. Натомість друга модель, з окремими шарами Conv2D та LSTM, може бути менш оптимальною в контексті обробки складних послідовностей кадрів, оскільки обробка просторових і часових аспектів здійснюється окремо, що може ускладнити ефективне моделювання їх взаємодії.

2.4 Висновки за розділом 2

1. У другому розділі було розглянуто та обґрунтовано різні архітектури нейронних мереж для задачі аналізу послідовностей кадрів гімнастичних поз.

2. Визначено підходи на основі згорткових нейронних мереж, які дозволяють ефективно витягувати просторові ознаки з окремих кадрів.

3. Проаналізовано можливості використання рекурентних нейронних мереж для моделювання часових залежностей у послідовностях, що забезпечує розуміння динаміки виконання гімнастичних елементів.

4. Виокремлено підхід на основі інтеграції згорткових і рекурентних підходів, що дозволяє враховувати як просторову, так і часову інформацію, забезпечуючи більш точне розпізнавання складних гімнастичних поз.

5. Отримані результати створюють основу для ефективного використання гібридних моделей у розв'язанні задачі інтелектуального аналізу кадрів у спортивному середовищі.

РОЗДІЛ 3
ІДЕНТИФІКАЦІЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ НЕЙРОМЕРЕЖЕВОЇ КЛАСИФІКАЦІЇ ПОСЛІДОВНОСТЕЙ КАДРІВ ГІМНАСТИЧНИХ ПОЗ

3.1 Збір та організація власного набору даних.

Для створення набору даних було використано відеозапис фінального етапу змагань з художньої гімнастики на XXXII Олімпійських іграх 2021 року, що проходили в Токіо [48]. Роздільна здатність відеозапису 1280 на 720 пікселів. Програма Vegas PRO [49] допомогла здійснити покадровий аналіз відео, що дозволило з високою точністю виділити необхідні фрагменти виконання елементів. Завдяки покадровому інтерфейсу програми, який відображає час у форматі години:хвилини:секунди:кадри, вдалося точно визначити початок і кінець кожного елемента. Це забезпечило контроль довжини відрізків виконання елементів для подальшого аналізу. Інтерфейс програми дозволяє відстежувати кожен кадр окремо, забезпечуючи необхідну точність для вирізання та категоризації рухів у рамках набору даних. На рисунку 3.1 наведено приклад виокремлення у програмі Vegas PRO виконання елемента який відноситься до 1 класу. У нижній лівій частині екрану можна побачити час початку виконання елементу, у правому нижньому куті довжину виконання елементу.

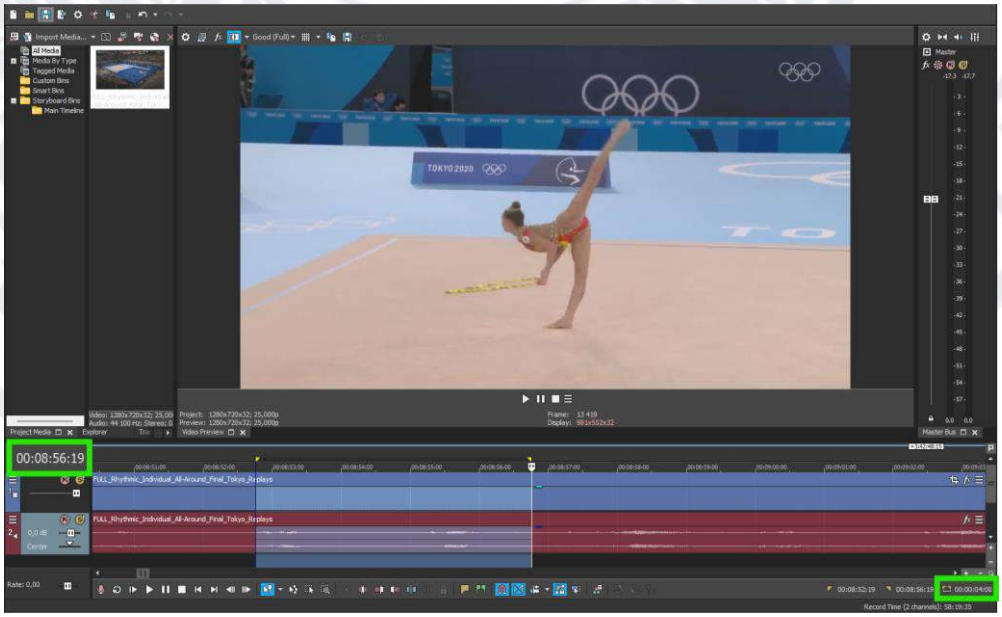

Рисунок 3.1. Приклад виокремлення у програмі Vegas PRO

Для створення набору даних було обрано наступні чотири види оцінюваних елементів, зображені на рисунку 3.2, які затверджені правилами міжнародної федерації гімнастики та дозволені для виконання:

- Клас 1 - поворот в положенні шпагат без використання рук, з горизонтальним розташуванням тулуба;
- Клас 2 - поворот в положенні шпагат з використанням рук, з вертикальним розташуванням тулуба;
- Клас 3 - елемент рівноваги в положенні шпагат без використання рук, з горизонтальним розташуванням тулуба;
- Клас 4 - стрибок з обертом в положенні шпагат без використання рук, з горизонтальним розташуванням тулуба.

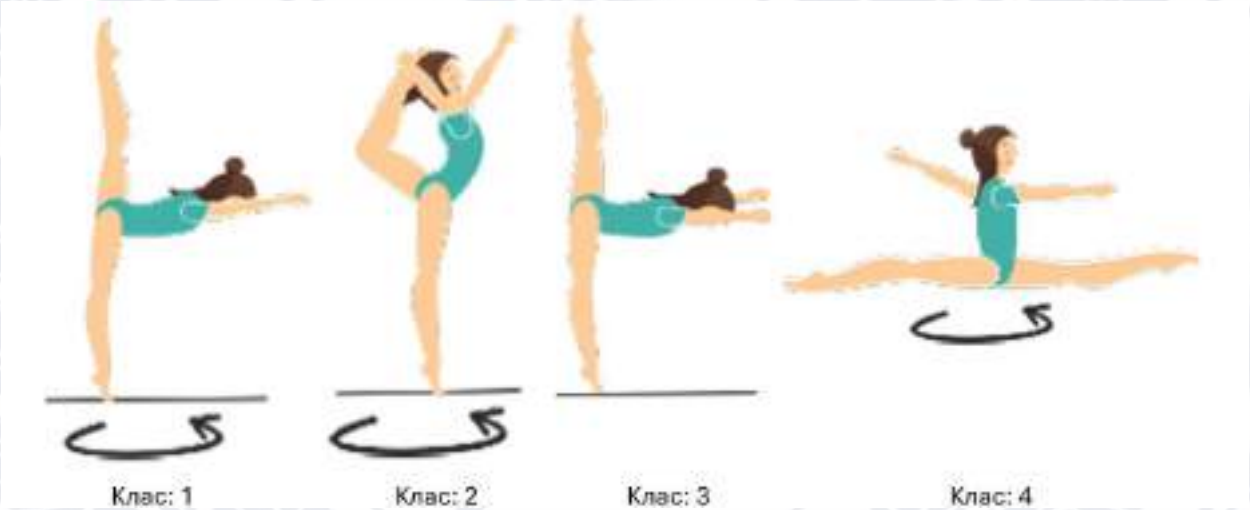

Рисунок 3.2. Схематичне зображення положення тіла при виконанні елемента кожного класу.

Відео-відрізки прикладів виконання елементів кожного класу описуються у окремому текстовому файлі. У кожному рядку файла записано час початку виконання елемента та час упродовж якого спортсменка виконується гімнастичний елемент одного класу. Один рядок описує один приклад виконання елемента класу. Тобто, кожен клас має відповідний текстовий файл з прикладами виконання цього класу. В результаті, набір даних складається з одного відео змагального дня та 4 текстових файлів для кожного класу. Такий формат збереження інформації дозволяє гнучко та оптимально, особливо за обсягом пам’яті, зберігати інформацію. Через те, що

зберігання окремих кадрів для виконання кожного елементу займає гігабайти пам’яті. Наприклад, зберігання кожного кадру кожного прикладу виконання елемента 1 класу займає близько 15 Гб. Використаний підхід дозволяє за необхідності з повного відео виступів вибирати інформацію про класи у необхідному для конкретної задачі форматі: відео відрізки, покадрово або кожен другий чи шостий кадр прикладу виконання елемента.

На рисунку 3.3 наведено приклад послідовностей кадрів виконання одного елемента обертання який відноситься до першого класу.

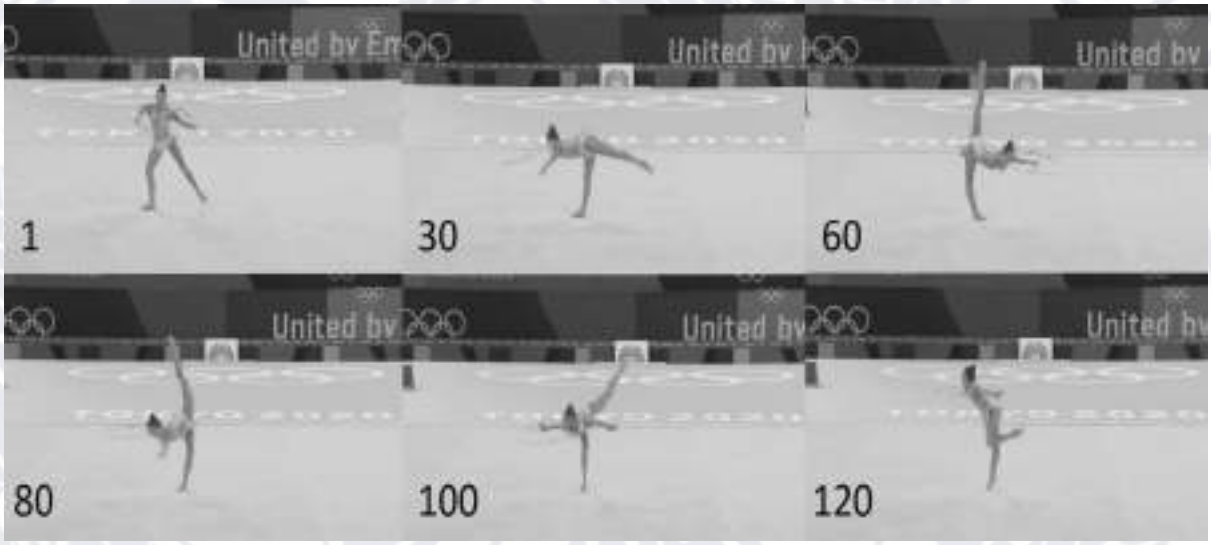

Рисунок. 3.3. На кадрах приклад зображення положення тіла під час виконання елемента повороту у шпагаті без використання рук

Вирізання прикладів елементів було реалізовано мовою програмування python з допомогою модуля os, який дозволяє організовувати роботу з файловою системою для обробки відео. Даний підхід також забезпечує гнучкість у виборі частоти зберігання кадрів. За допомогою параметрів можна вказати, які саме кадри слід витягувати: усі, через кожен другий або навіть шостий кадр. Це дозволяє значно зменшити обсяг необхідної пам'яті для зберігання даних, при цьому не втрачаючи важливої інформації для аналізу. Наприклад, якщо елемент виконується тривало, можна використовувати кожен шостий кадр для економії пам'яті без втрати точності аналізу руху. Це значно скорочує час підготовки даних для тренування нейронних мереж та інших алгоритмів.

Також, важливо зазначити, що модуль os надає можливість управління файлами та каталогами: створення нових директорій для зберігання вирізаних кадрів, перейменування файлів, перевірка наявності даних або видалення зайвих фрагментів. Це особливо корисно при роботі з великими обсягами відеоматеріалів, що допомагає підтримувати порядок у файловій системі та уникати дублювання даних. Додатково, збереження кадрів у різних форматах, наприклад, JPEG або PNG дозволяє користувачам обирати найбільш відповідний формат для подальшого аналізу або зберігання. Формат файлу JPEG добре підходить для зменшення розміру файлів, тоді як PNG забезпечує високу якість зображення для більш точного аналізу.

Фрагмент коду який використовується для формування відео набору даних RG Elements:

```
import os
data = []
cls = "cls_panche_turn"
with open("{}.txt".format(cls), 'r') as file:
  for line in file:
    values = line.strip().split()
    if len(values) == 2:
      time_start, time_len = values
      data.append([time_start, time_len])
i = 0
for ts, tl in data:
  os.system("""ffmpeg -ss {} -i FULL_Rhythmic_Individual_All-Around_Final_Tokyo_Replays.mp4 -to {} -vf "crop=720:720" -c:v libx264 -preset fast -c:a copy {}/{}_{}.mp4""".format(ts, tl, cls, cls, i))
  i += 1
```

Цей фрагмент коду на Python призначений для автоматичного вирізання відео-фрагментів виконання гімнастичних елементів із повного відео на основі часу початку та тривалості, зазначених у текстовому файлі, та збереження їх у

нові файли. Спершу імпортується модуль os, що дозволяє взаємодіяти з операційною системою, зокрема для виконання команд оболонки, таких як виклик програми ffmpeg. Далі відбувається ініціалізація змінних для зберігання інформації з подальшою її обробкою. Змінна data — це порожній список, до якого будуть додаватися дані про час початку та тривалість відео-фрагментів, тоді як змінна cls містить ім'я класу (в даному випадку "class_1"). Ця змінна використовується як для позначення текстового файлу, де зберігаються дані про часові межі фрагментів, так і для генерації назв кінцевих відеофайлів.

Дані з текстового файлу зчитуються з використанням механізму відкриття файлу на читання. Файл, назва якого відповідає значенню змінної cls (наприклад, "class_1.txt"), містить у кожному рядку дві змінні — час початку (time_start) та тривалість (time_len) фрагмента. Кожен рядок розділяється на дві частини, і перевіряється, чи він дійсно складається з двох значень. Якщо умова виконується, дані додаються до списку data у вигляді підсписків. Цей підхід дозволяє гнучко зберігати та обробляти інформацію про часові параметри фрагментів.

Основний процес вирізання відеофрагментів реалізовано за допомогою програми ffmpeg. Для кожного елемента у списку data запускається цикл, у якому змінні ts та tl відповідають часу початку та тривалості відповідного фрагмента. Використовується команда os.system() для виклику ffmpeg, що вирізає відеофрагмент із використанням наступних параметрів:

- -ss {} визначає точний час початку вирізання;
- -i FULL_Rhythmic_Individual_All-Around_Final_Tokyo_Replays.mp4 вказує шлях до оригінального відеофайлу;
- -to {} встановлює тривалість кожного фрагмента;
- -vf "crop=720:720" обрізає кадр до розміру 720x720 пікселів для зручності подальшого аналізу;
- -c:v libx264 -preset fast виконує кодування за допомогою кодека H.264 з налаштуванням швидкого пресету, що забезпечує ефективне стиснення

без значної втрати якості.

- -c:a copy застосовується для копіювання аудіодоріжки без змін, що зберігає вихідну якість звуку.

Кінцевий файл зберігається у відповідному класовому каталозі (наприклад, "class_1"), а його назва автоматично формується за схемою "class_i.mp4", де "i" — індекс, що збільшується з кожною ітерацією циклу. Така система індексації гарантує унікальність кожного файлу та дозволяє легко ідентифікувати фрагменти. Таким чином, цей алгоритм не тільки автоматизує процес вирізання фрагментів із тривалої відеопослідовності, але й забезпечує точність і систематичність зберігання, що значно полегшує подальшу обробку даних для інтелектуального аналізу спортивних рухів.

Також, код легко можна модифікувати для роботи з іншими відеофайлами або класами, просто змінивши назви файлів або параметри. Цей підхід дозволяє легко витягувати та обробляти відео з використанням стандартних бібліотек і інструментів.

Таким чином, реалізація вирізання прикладів гімнастичних елементів із відео на Python з використанням модулів os є ефективним підходом для обробки великих обсягів відеоданих, забезпечуючи при цьому гнучкість і масштабованість для подальших досліджень і навчання моделей машинного навчання.

В результаті, набір даних RG Elements включає 240 прикладів виконання чотирьох різних видів гімнастичних елементів. Ці елементи виконуються десятьма різними гімнастками, кожна з яких використовує один із чотирьох типів снарядів, таких як стрічка, обруч, м'яч або булави. Це надає різноманіття даним, оскільки кожна гімнастка має унікальний стиль виконання елементів і взаємодії зі снарядом. Всі приклади мають певну варіативність залежно від індивідуальної техніки спортсмена, тривалості виконання, а також особливостей роботи з кожним конкретним снарядом.

Кожен приклад у наборі складається із відео фрагменту, який відображає виконання одного елементу. Тривалість кожного відео різниться, оскільки

вона залежить від часу, необхідного спортсмену для повного виконання цього елементу. Ця тривалість коливається від однієї секунди до шести секунд, що дає значну варіацію в розмірі і обсязі даних. Наприклад, короткі елементи можуть включати швидкі й динамічні рухи, тоді як довші фрагменти можуть містити більше плавних та розтягнутих рухів гімнастки. У середньому, тривалість одного відео-фрагменту складає приблизно три секунди.

Кожна секунда виконання елементу у відео описується за допомогою 30 кадрів. Таким чином, кожен відео фрагмент складається з набору окремих кадрів, які детально відображають зміни положення тіла спортсмена в просторі. Це дозволяє зберігати точну інформацію про траєкторію руху, положення кінцівок, голови, корпусу та снаряда під час виконання складних елементів. Дана структура дозволяє більш ефективно використовувати ці кадри для аналізу або навчання моделей машинного навчання.

Особливістю набору даних RG Elements є те, що він включає приклади виконання елементів з різними снарядами. Кожен снаряд вимагає специфічних навичок роботи від гімнастки і може впливати на траєкторію рухів та техніку виконання. Наприклад, робота зі стрічкою передбачає більш плавні і розтягнуті рухи, тоді як елементи з обручем часто є більш динамічними та точними. Булави і м'яч, у свою чергу, вимагають іншого підходу до контролю та балансу тіла. Це різноманіття рухів забезпечує багатий набір даних для подальшого аналізу.

Набір даних також дозволяє зберігати зміни в положенні тіла спортсменів при виконанні елементів, що є ключовим для детального вивчення рухів у художній гімнастиці. Кожен рух спортсмена під час виконання елементу аналізується кадр за кадром, що дає змогу точно фіксувати зміни в положенні тіла, напрямку руху або роботі зі снарядом. Така структура дає можливість використовувати різні підходи до роботи з даними: як аналіз усього відео, так і обробку окремих кадрів або фрагментів.

Таким чином, набір даних RG Elements може використовуватись для досліджень у галузі комп'ютерного зору та інтелектуального аналізу

спортивних рухів, оскільки він зберігає високу деталізацію виконання гімнастичних елементів і враховує різноманіття технік та снарядів.

3.2 Попередня обробка набору даних

Для проведення чисельного дослідження було використано власний набір даних RG Elements. Набір даних розділений на навчальний набір із 200 прикладів і тестовий набір із 40 прикладів виконання елемента. Для реалізації запропонованих нейронних мереж використовувався пакет tensorflow, у програмному середовищі Google Collaboratory.

Спочатку було виконано попередню обробку набору даних. Для цього було написано функцію frames_extraction, яка забезпечує підготовку даних для подальшого аналізу гімнастичних поз. Функція спочатку зчитує відеофайли з набору даних і змінює розмір кадрів відео до фіксованої ширини і висоти, щоб зменшити обчислення і нормалізувати дані до діапазону від нуля до одиниці шляхом ділення значень пікселів на 255, що прискорює збіжність під час навчання мережі. Було обрано змінити розмір зображення до 128 пікселів по ширині і висоті, що призводить до зменшення обчислювальних витрат. Кількість кадрів відео, які будуть подаватися у модель як одна послідовність, має бути однакова для кожного відео, тому було обрано 30 кадрів для кожної послідовності (SEQUENCE_LENGTH). Наступний рядок коду розраховує інтервал для вибірки кадрів:

```
skip_frames_window = max(int(video_frames_count/ SEQUENCE_LENGTH), 1)
```

Інтервал, через який будуть вибиратися кадри, обчислюється на основі загальної кількості кадрів і бажаної довжини послідовності. Ця функція є важливою частиною попередньої обробки даних і забезпечує стандартизацію відео для подальшого аналізу, зменшуючи розмір відео та нормалізуючи кадри для моделі машинного навчання.

Далі було створено функцію для створення набору даних create_dataset, яка ітераційно перебирає всі класи і викликатє функцію frame_extraction для кожного відеофайлу вибраних класів і повертатиме кадри - змінна features, індекс класу - змінна labels, і шлях до відеофайлу - змінна video_files_paths.

Після використання цієї функції необхідно перетворити індекс класу, змінна labels у вектори з однократним кодуванням. Для цього використаємо наступний фрагмент коду: one_hot_encoded_labels = to_categorical(labels).

Другий варіант попередньої обробки даних полягає у вилучені з послідовності кадрів ключових точок скелету спортсмена за допомогою попередньо навченої моделі. MediaPipe Pose Landmarker — це бібліотека, яка біла розроблена компанією Google для використання розробниками у сфері штучного інтелекту, що реалізує завдання визначення анатомічних орієнтирів людського тіла на зображеннях або відео. Вона дозволяє ідентифікувати ключові точки тіла[50]. Технологія базується на використанні моделей машинного навчання, які можуть працювати як зі статичними зображеннями, так і з потоковим відео.

Основна функція бібліотеки полягає у виявленні 33 ключових орієнтирів людського тіла, які схематично зображені на рисунку 3.4, що включають такі частини, як ніс, очі, плечі, лікті, зап’ястя, стегна, коліна, щиколотки та пальці. Для цього використовується двоетапна модель: спочатку визначається наявність людського тіла в кадрі, а потім визначається його анатомічна структура. Результатом є координати орієнтирів у нормалізованій площині зображення, а також у тривимірному просторі.

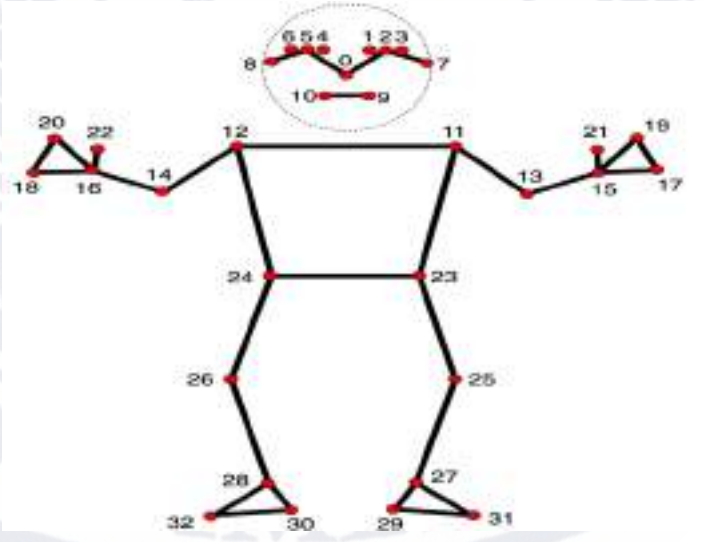

Рисунок 3.4. Схема розташування ключових орієнтирів людського тіла.

Бібліотека MediaPipe використовує оптимізовані нейронні мережі на основі MobileNetV2, що забезпечують ефективну обробку даних у режимі реального часу навіть на пристроях з обмеженими ресурсами.

Для тривимірного моделювання людського тіла застосовується архітектура BlazePose, яка інтегрує GHUM — сучасний підхід до моделювання форм людського тіла [51].

Бібліотека підтримує декілька режимів роботи, таких як обробка одиничних зображень, кадрів відео або потокового відео. Крім того, користувач може налаштовувати параметри, як-от мінімальний поріг довіри до результатів, максимальну кількість тіл у кадрі або наявність сегментаційної маски.

Особливістю MediaPipe також є простота її підключення у проект. Після встановлення бібілотеки потрібно ініціалізувати потрібний модуль та створити об’єкт класу який виконує обробку кадра для визначення ключових точок скелету людини. Приклад кода:

```
import mediapipe as mp
mp_pose = mp.solutions.pose
pose = mp_pose.Pose(static_image_mode=False, model_complexity=2, enable_segmentation=False)
```

Ця технологія знаходить застосування в різних галузях, зокрема для аналізу рухів, відстеження пози у фітнесі, а також створення інтелектуальних систем аналізу даних.

3.3 Ідентифікація та програмна реалізація моделі для аналізу послідовностей кадрів гімнастичних поз на основі архітектури ConvLSTM

На цьому кроці ми реалізуємо перший підхід, використовуючи комбінацію клітинок ConvLSTM. Клітинка ConvLSTM є варіантом мережі LSTM, яка містить операції згортки в мережі. Це LSTM із згорткою, вбудованою в архітектуру, що робить його здатним ідентифікувати просторові особливості даних, зберігаючи до уваги часові відношення.

Для класифікації відео цей підхід ефективно фіксує просторове співвідношення в окремих кадрах і часове співвідношення між різними кадрами. Завдяки такій структурі згортки ConvLSTM здатний отримувати тривимірні вхідні дані (ширина, висота, кількість_каналів), тоді як простий LSTM приймає лише одновимірні вхідні дані, тому LSTM несумісний для моделювання просторово-часових даних на своїх власні.

Для побудови моделі було використано повторювані шари з бібліотеки Keras ConvLSTM2D. Шар ConvLSTM2D також приймає кількість фільтрів і розмір ядра, необхідні для застосування згорткових операцій. Вихідні дані шарів у кінці вирівнюються та передаються на щільний шар із активацією softmax, яка виводить ймовірність кожної категорії дії. Також використоно шари MaxPooling3D, щоб зменшити розміри кадрів і уникнути непотрібних обчислень, а також шари Dropout, щоб запобігти переобладнанню моделі на даних. Архітектура проста і має невелику кількість параметрів, які можна навчити. Це пояснюється тим, що ми маємо справу лише з невеликою підмножиною набору даних, яка не потребує великомасштабної моделі.

Наступний фрагмент коду виконує структурну ідентифікацію моделі на основі підходу ConvLSTM:

```
def create_convlstm_model():
    model = Sequential()
    model.add(ConvLSTM2D(filters = 4, kernel_size = (3, 3), activation = 'tanh', data_format = "channels_last", recurrent_dropout=0.2, return_sequences = True, input_shape = (SEQUENCE_LENGTH,IMAGE_HEIGHT, IMAGE_WIDTH, 3)))
    model.add(MaxPooling3D(pool_size=(1, 2, 2), padding='same', data_format= 'channels_last'))
    model.add(TimeDistributed(Dropout(0.2)))
    model.add(ConvLSTM2D(filters = 8, kernel_size = (3, 3), activation = 'tanh', data_format = "channels_last", recurrent_dropout=0.2, return_sequences=True))
    model.add(MaxPooling3D(pool_size=(1, 2, 2), padding='same', data_format= 'channels_last'))
```

```
    model.add(TimeDistributed(Dropout(0.2)))
    model.add(ConvLSTM2D(filters = 14, kernel_size = (3, 3), activation = 'tanh', data_format = "channels_last", recurrent_dropout=0.2, return_sequences=True))
    model.add(MaxPooling3D(pool_size=(1, 2, 2), padding='same', data_format='channels_last'))
    model.add(TimeDistributed(Dropout(0.2)))
    model.add(ConvLSTM2D(filters = 16, kernel_size = (3, 3), activation = 'tanh', data_format = "channels_last", recurrent_dropout=0.2, return_sequences=True))
    model.add(MaxPooling3D(pool_size=(1, 2, 2), padding='same', data_format= 'channels_last'))
    model.add(Flatten())
    model.add(Dense(len(CLASSES_LIST), activation = "softmax"))
    model.summary()
    return model
```

Модель створюється за допомогою класу Sequential, що дозволяє додавати шари по черзі. Це зручний спосіб створення нейронних мереж, де кожен шар є наступним кроком після попереднього. Модель починається з шару ConvLSTM2D з 4 фільтрами та ядром розміром 3 на 3. Активаційна функція — tanh, яка часто використовується в рекурентних нейронних мережах для обробки часових послідовностей. Параметр recurrent_dropout=0.2 додає регуляризацію для запобігання перенавчанню. Параметр return_sequences = True вказує на те, що модель повинна повертати послідовність, оскільки кожен шар передає на вихід кадри для наступного шару. Вхідні дані шару мають форму (SEQUENCE_LENGTH, IMAGE_HEIGHT, IMAGE_WIDTH, 3), що відповідає послідовності кадрів з розміром зображень і трьома каналами.

Після шару ConvLSTM2D додається шар MaxPooling3D, який виконує операцію підвибірки для зменшення розміру просторових даних, зокрема у відео. Далі використовується TimeDistributed(Dropout(0.2)), який накладає регуляризацію шляхом відкидання 20% випадкових нейронів, що допомагає

уникати перенавчання. Шар TimeDistributed забезпечує застосування шару відсіву до кожного кадру окремо. Другий шар ConvLSTM2D має 8 фільтрів і подібну структуру. Він також використовує активацію tanh і recurrent_dropout=0.2. Шар return_sequences=True залишається активованим, щоб продовжувати передавати послідовність кадрів для подальшої обробки.

Наступним знову додається шар MaxPooling3D для зменшення розміру даних, а також шар відсіву, який працює через TimeDistributed, з метою регуляризації даних. Наступні шари мають 14 та 16 фільтрів, що поступово збільшує кількість фільтрів у моделі для більшої здатності розпізнавати складніші шаблони. Для обох шарів використовуються ті самі налаштування активації, рекурентного відсіву і передачі послідовностей. Останній шар MaxPooling3D використовується для зменшення розміру даних до більш компактного виду.

Шар Flatten перетворює багатовимірні дані на одновимірний вектор, який можна подати на вхід щільному шару. Щільний шар додається останнім, він відповідає за класифікацію. Функція активації softmax використовується для перетворення виходів у ймовірності, що підходить для багатокласової класифікації. Кількість нейронів цього шару відповідає кількості класів у наборі даних. Виклик функції model.summary показує детальну архітектуру моделі, включаючи кількість параметрів і розміри шарів. Функція завершується поверненням готової моделі, яку можна використовувати для навчання на послідовностях відеокадрів.

Архітектура моделі яка була побудована за використанням функції create_convlstm_model враховуючи параметри набору даних наведена у таблиці 3.1. Загальна кількість параметрів у побудованій моделі 127084.

Таблиця 3.1 - Архітектура моделі нейронної мережі

Шар	Вихідний розмір	Кількість параметрів
ConvLSTM2D	(None, 30, 126, 126, 4)	1024
MaxPooling3D	(None, 30, 63, 63, 4)	0

Продовження таблиці 3.1

TimeDistributed	(None, 30, 63, 63, 4)	0
ConvLSTM2D	(None, 30, 61, 61, 8)	3488
MaxPooling3D	(None, 30, 31, 31, 8)	0
TimeDistributed	(None, 30, 31, 31, 8)	0
ConvLSTM2D	(None, 30, 29, 29, 14)	11144
MaxPooling3D	(None, 30, 15, 15, 14)	0
TimeDistributed	(None, 30, 15, 15, 14)	0
ConvLSTM2D	(None, 30, 13, 13, 16)	17344
MaxPooling3D	(None, 30, 7, 7, 16)	0
Flatten	(None, 23520)	0
Dense	(None, 4)	94084

Розмір шару складається з наступних параметрів:

- Перший параметр — розмір пакету, який буде визначено під час навчання або передбачення.
- 30 — кількість кадрів у послідовності (довжина послідовності).
- 126x126 — просторовий розмір (висота і ширина зображення).
- Останній параметр — кількість каналів (наприклад, глибина фільтрів для кожного кадру).

Навчання моделі подібного розміру використовуючи ресурси TPU Google Colaboratory займає близко 15 хвилин. Команда convlstm_model.save('convlstm_model.h5') використовується для збереження навченої моделі у файл на диск. convlstm_model — це змінна, що містить створену та вже навчену ConvLSTM-модель, яку ми хочемо зберегти. За допомогою методу save класу Model в бібліотеці TensorFlow, виконується збереження архітектури моделі та її ваги. Файл буде збережений у форматі HDF5. Після виконання цієї команди, модель можна буде завантажити пізніше для подальшого використання без необхідності повторного навчання. Файл

міститиме не тільки структуру моделі, але й усі збережені ваги, що дозволяє відновити модель у тому ж стані, в якому вона була збережена.

Для оптимізації процесу навчання моделі було використано механізм ранньої зупинки. Цей підхід передбачає моніторинг значення обраного показника, в даному випадку, функції втрат на валідаційному наборі даних протягом навчання. У випадку, якщо показник не покращується протягом 10 епох, навчання припиняється. Вибір мінімального значення функції втрат і автоматичне відновлення ваг найкращої ітерації забезпечують ефективність навчання та зменшення ризику перенавчання.

Для компіляції моделі було визначено функцію втрат категоріальної крос-ентропії, яка підходить для задач багатокласової класифікації, а також метод оптимізації Adam, що забезпечує адаптивне налаштування швидкості навчання. Як метрику ефективності було обрано точність моделі, що дозволяє оцінювати якість класифікації на основі частки правильних передбачень.

Процес навчання відбуваються на протязі 25 епох із розміром партії даних, встановленим на 4, що дозволяє ефективно використовувати обчислювальні ресурси. Навчальні дані були розділені на навчальний і валідаційний під набори у співвідношенні 75% до 25%. Крім того, включення функції перемішування даних сприяє уникненню локальних мінімумів під час оптимізації. Навчання супроводжувалося збереженням історії, яка використовується для подальшого аналізу процесу навчання моделі.

На рисунку 3.5 представлено графіки залежність точності та втрат протягом процесу навчання побудованої моделі на тренувальному та валідаційному наборах даних моделі ConvLSTM.

Результати навчання моделі демонструють зростання точності як на тренувальних, так і на валідаційних даних, під час навчання, проте після 22-гої епохи точність валідаційних даних стабілізується. На тренувальній вибірці, точність досягає 96.52% на 23-й епосі, що означає високу точність навчання.

Оцінка продуктивності навчання на валідаційній вибірці показує складнішу динаміку. На ранніх епохах модель демонструє низьку точність і

високу втрату на валідаційних даних, однак із поступовим збільшенням кількості епох результати покращуються, досягаючи точності 70.37% із найменшими втратами 0.8034 на 15-й епосі. Це свідчить про хорошу здатність моделі до узагальнення на даному етапі навчання. Незважаючи на це, подальше збільшення кількості епох призводить до коливань показників на валідаційній вибірці. Спостерігається деяке зростання втрат і зниження точності в окремих точках навчання, що може свідчити про початок перенавчання.

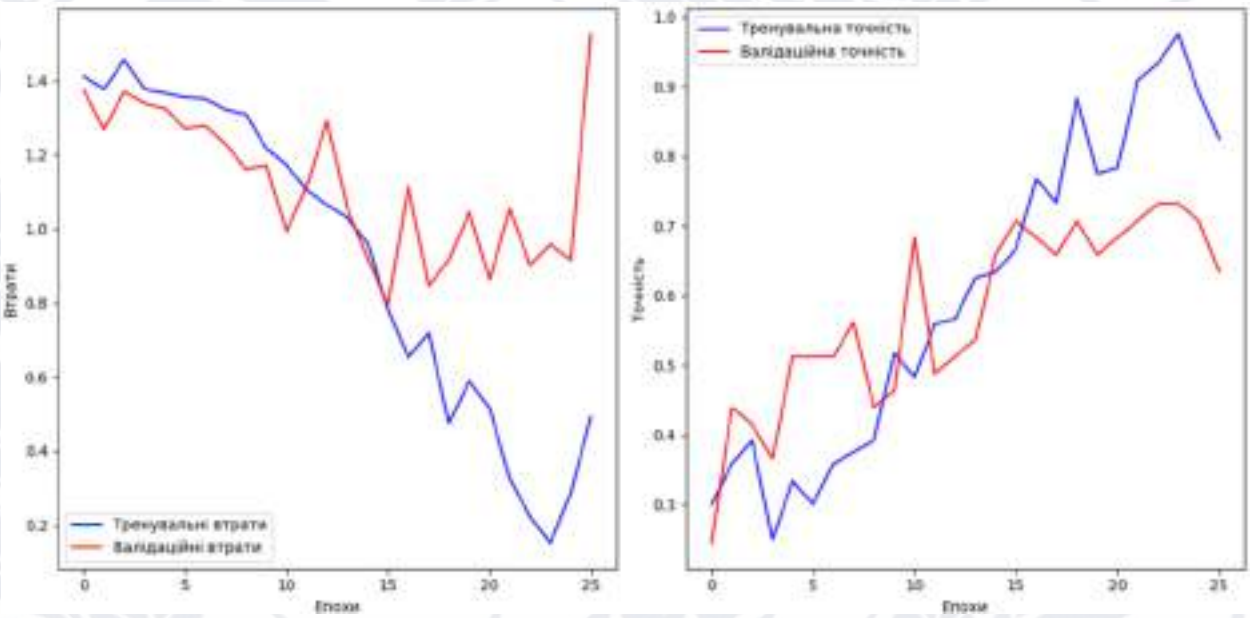

Рисунок 3.5. Залежність точності та втрат від кількості ітерацій моделі ConvLSTM.

Загалом модель демонструє точність 75,56% на тестовій вибірці даних, яку не бачила під час навчання. Такі результати свідчать про необхідність удосконалення процесу навчання, щоб уникнути перенавчання і покращити узагальнюючу здатність моделі.

3.4 Ідентифікація та програмна реалізація моделі для аналізу послідовностей кадрів гімнастичних поз на основі архітектури LRCN

На цьому етапі буде реалізовано підхід LRCN (Long-term Recurrent Convolutional Network), який об'єднує шари згорткових нейронних мереж і рекурентних нейронних мереж в єдину архітектуру. Згорткові шари відповідають за вилучення просторових характеристик з кожного кадру відео,

після чого ці характеристики передаються до LSTM на кожному часовому кроці для моделювання часової послідовності. Це дозволяє мережі вивчати просторово-часові характеристики під час наскрізного навчання, що сприяє побудові надійної моделі для розпізнавання дій. Крім того, використовується шар-обгортка TimeDistributed, який дозволяє застосовувати шар CNN до кожного кадру відео окремо. Це особливо корисно для вхідних даних у форматі (кількість кадрів, ширина, висота, кількість каналів), оскільки дозволяє передавати цілі відео послідовності через модель в одному виклику, обробляючи кадри незалежно.

Для реалізації LRCN архітектури будуть використані TimeDistributed-шари Conv2D, після яких йдуть шари MaxPooling2D і Dropout. Ознаки, вилучені з шарів Conv2D, згодом обробляються шаром Flatten, після чого передаються на вхід до рекурентного шару LSTM. Для класифікації дій використовується щільний шар із функцією активації softmax, який приймає вихід LSTM і передбачає виконувану дію. Наступний фрагмент коду виконує структурну ідентифікацію моделі на основі підходу LRCN:

```
def create_LRCN_model():
  model = Sequential()
  model.add(TimeDistributed(Conv2D(16,(3,3),padding='same',activation = 'relu'),
input_shape = (SEQUENCE_LENGTH, IMAGE_HEIGHT, IMAGE_WIDTH, 3)))
  model.add(TimeDistributed(MaxPooling2D((4, 4))))
  model.add(TimeDistributed(Dropout(0.25)))
  model.add(TimeDistributed(Conv2D(32,(3,3),padding='same',activation='relu')))
  model.add(TimeDistributed(MaxPooling2D((4, 4))))
  model.add(TimeDistributed(Dropout(0.25)))
  model.add(TimeDistributed(Conv2D(64,(3,3),padding='same',activation='relu')))
  model.add(TimeDistributed(MaxPooling2D((2, 2))))
  model.add(TimeDistributed(Dropout(0.25)))
  model.add(TimeDistributed(Conv2D(64,(3,3),padding='same',activation='relu')))
  model.add(TimeDistributed(MaxPooling2D((2, 2))))
```

```
model.add(TimeDistributed(Flatten()))
model.add(LSTM(32))
model.add(Dense(len(CLASSES_LIST), activation = 'softmax'))
model.summary()
return model
```

Для побудови моделі класифікації використовується клас Sequential. Спочатку модель застосовує обгорнуті в шар TimeDistributed згорткові шари Conv2D. На цьому етапі застосовуються згортки до кожного кадру відео послідовності виконання елементу окремо. Перший згортковий шар має 16 фільтрів із розміром ядра 3x3 і функцією активації ReLU, що допомагає вилучати базові просторові характеристики з кожного кадру виконання елементу. Після нього йде шар MaxPooling2D, який зменшує розмірність вхідних даних, і шар регуляризації Dropout, що запобігає перенавчанню, вимикаючи 25% нейронів під час навчання.

Другий і третій згорткові шари мають 32 та 64 фільтри відповідно, і для кожного з них також застосовуються шари MaxPooling2D та Dropout, що зменшують розмірність ознак і сприяють кращій генералізації. Після цих шарів додано ще один згортковий шар із 64 фільтрами, що дозволяє моделі глибше аналізувати просторові характеристики кадрів.

Після згорткових шарів додається шар Flatten, який перетворює багатовимірні дані у вектор для подальшого використання в рекурентному шарі LSTM. Рекурентний шар LSTM із 32 одиницями відповідає за обробку часових залежностей між кадрами у відео послідовності. Вихід LSTM надходить на щільний шар, де застосовується функція активації softmax для класифікації відео за кількістю класів. Модель завершується викликом методу summary(), який відображає загальну архітектуру моделі, після чого функція повертає побудовану модель.

Архітектура моделі яка була побудована за використанням функції create__LRCN_model враховуючи параметри набору даних наведена у таблиці 3.2. Загальна кількість параметрів у побудованій моделі 97636.

Таблиця 3.2 - Архітектура моделі нейронної мережі LRCN

Шар	Вихідна форма	Кількість параметрів
TimeDistributed	(None, 30, 128, 128, 16)	448
TimeDistributed	(None, 30, 32, 32, 16)	0
TimeDistributed	(None, 30, 32, 32, 16)	0
TimeDistributed	(None, 30, 32, 32, 32)	4640
TimeDistributed	(None, 30, 8, 8, 32)	0
TimeDistributed	(None, 30, 8, 8, 32)	0
TimeDistributed	(None, 30, 8, 8, 64)	18,496
TimeDistributed	(None, 30, 4, 4, 64)	0
TimeDistributed	(None, 30, 4, 4, 64)	0
TimeDistributed	(None, 30, 4, 4, 64)	36,928
TimeDistributed	(None, 30, 2, 2, 64)	0
TimeDistributed	(None, 30, 256)	0
LSTM	(None, 32)	36,992
Dense	(None, 4)	132

Таблиця демонструє структуру LRCN моделі, яка складається з 13 шарів, що включають комбінацію TimeDistributed шарів, рекурентного шару LSTM та фінального щільного шару для класифікації. TimeDistributed шари обробляють кадри відео окремо, застосовуючи операції згортки, підвибірки та регуляризації до кожного кадру. В результаті, модель може вилучати просторові характеристики для кожного кадру відео. LSTM шар обробляє послідовність цих ознак, дозволяючи моделі вивчати часові залежності між кадрами. Dense шар із функцією активації softmax на виході відповідає за класифікацію дії на 4 класи на основі отриманих часово-просторових характеристик.

Для оптимізації процесу навчання моделі LRCN було використано механізм ранньої зупинки. Підхід дозволяє відслідковувати значення функції втрат на валідаційному наборі даних під час навчання. Якщо функція втрат не

покращується протягом 15 послідовних епох, навчання припиняється. Режим мінімізації гарантує вибір найкращих ваг моделі, які були досягнуті протягом навчання, завдяки параметру. Модель була скомпільована з використанням категоріальної крос-ентропії як функції втрат, як оптимізатор було обрано метод Adam, як метрика для оцінки продуктивності моделі використовувалася точність.

Навчання моделі виконувалося протягом 48 епох з розміром партії даних чотири. Дані були поділені на навчальний та валідаційний під набори у співвідношенні 70% до 30%, також у процесі оптимізації використовувалося перемішування даних. Усі результати навчання, включно зі зміною функції втрат та точності для навчального та валідаційного під наборів, зберігалися у вигляді історії навчання для подальшого аналізу.

На рисунку 3.6 представлено графіки залежність точності та втрат протягом процесу навчання побудованої моделі на тренувальному та валідаційному наборах даних моделі LRCN.

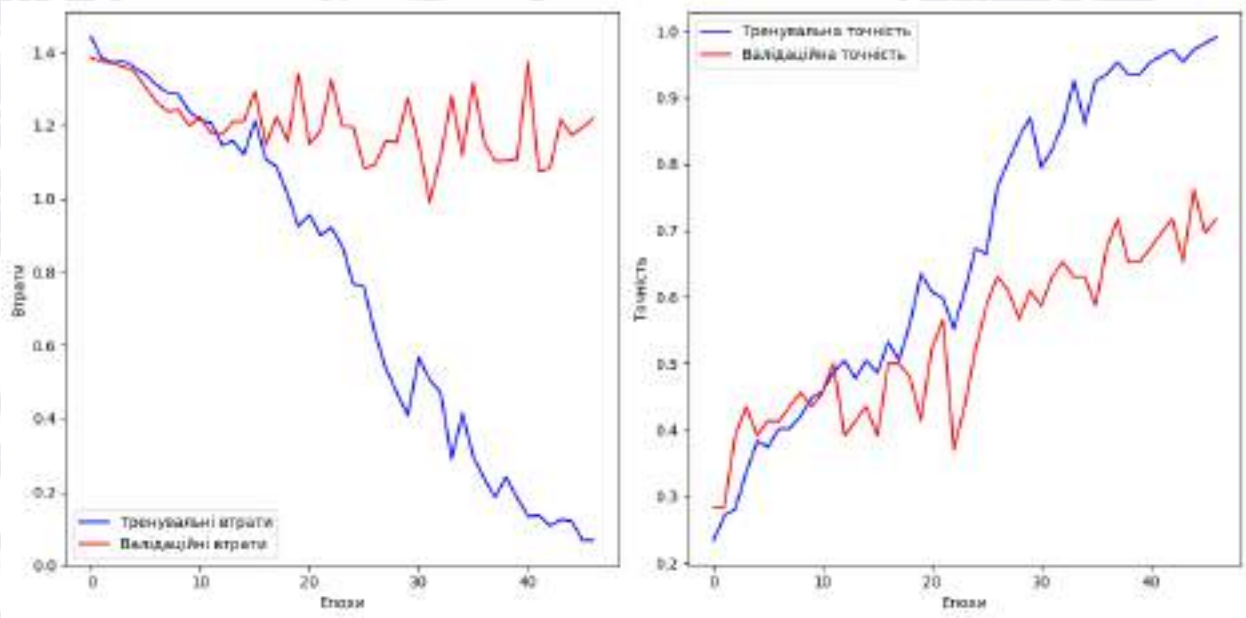

Рисунок 3.6. Залежність точності та втрат від кількості ітерацій моделі LRCN.

Історія навчання моделі демонструє поступове покращення її характеристик у перші епохи. Зменшення функції втрат для навчальної вибірки та валідаційного набору свідчить про те, що модель добре вивчає

закономірності у даних. На початкових етапах спостерігається зростання точності як для навчальної, так і для валідаційної вибірки, що вказує на адекватну здатність моделі до узагальнення.

Проте далі, після 12-тої епохи, функція втрат на валідаційному наборі починає коливатися без суттєвого покращення, хоча точність навчання продовжує зростати. Це є потенційною ознакою перенавчання, коли модель починає адаптуватися до специфічних характеристик навчальних даних і втрачає здатність узагальнювати на нові дані. Зокрема, на останніх епохах, валідаційна точність досягає значення 75.74%, тоді як навчальна точність сягає майже 99%, що підтверджує дисбаланс між навчальними та валідаційними результатами.

Хоча остаточна валідаційна точність є прийнятною, рівень функції втрат залишається високим. Це може вказувати на те, що модель не оптимально працює з деякими класами або складними прикладами з валідаційного набору. Параметри ранньої зупинки сприяли уникненню сильнішого перенавчання, але загалом слід розглянути додаткові заходи, такі як збільшення обсягу навчальних дани або модифікація архітектури моделі.

3.5 Ідентифікація та програмна реалізація моделі для аналізу послідовностей кадрів гімнастичних поз на основі архітектури LSTM

У цьому пункті представлено підхід, що поєднує використання бібліотеки MediaPipe для отримання ключових точок людського тіла з побудовою моделі LSTM для класифікації гімнастичних поз. Кожен кадр послідовності обробляється за допомогою MediaPipe Pose Landmarks, яка виконує детекцію 33 ключових точок скелету тіла спортсмена. Послідовність таких координат для всіх кадрів формується як вхідні дані для моделі LSTM. Ця модель, побудована з трьома рекурентними шарами, здатна аналізувати залежність між положеннями тіла на різних кадрах, що дозволяє їй визначати, до якої гімнастичної пози належить задана послідовність. Модифікована функція для виконання попередньої обробки набору даних:

def frames_extraction_with_landmarks(video_path):frame.

```
 landmarks_list = []
 video_reader = cv2.VideoCapture(video_path)
 video_frames_count = int(video_reader.get(cv2.CAP_PROP_FRAME_COUNT))
skip_frames_window = max(int(video_frames_count / SEQUENCE_LENGTH), 1)
  for frame_counter in range(SEQUENCE_LENGTH):
    video_reader.set(cv2.CAP_PROP_POS_FRAMES, frame_counter * skip_frames_window)
    success, frame = video_reader.read()
    if not success: break
    rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = pose.process(rgb_frame)
    if results.pose_landmarks:
      landmarks = [[lm.x, lm.y, lm.z] for lm in results.pose_landmarks.landmark]
      landmarks_list.append(landmarks)
    else: landmarks_list.append(np.zeros((33, 3)))
  video_reader.release()
  return np.array(landmarks_list)
```

Попередня обробка набору тепер включає знаходження ключових точок скелету спортсмена на кожному кадрі відео послідовності. Для цього реалізовано функцію extract_landmarks_from_video, яка виконує вилучення ключових точок тіла людини із заданого відео. Результатом функції є послідовність ключових точок, які репрезентують положення тіла на основі рівномірно вибраних кадрів для кожного відео прикладу набору даних.

Спершу функція ініціалізує порожній список landmarks_list для зберігання координат ключових точок. Потім за допомогою бібліотеки OpenCV відкривається відеофайл, і визначається загальна кількість кадрів за допомогою властивості cv2.CAP_PROP_FRAME_COUNT. Для вибору рівновіддалених кадрів розраховується крок між кадрами, що визначається як частка від поділу загальної кількості кадрів на константу SEQUENCE_LENGTH, яка задає кількість кадрів, необхідних для аналізу.

Якщо результат розрахунку виявляється меншим за одиницю, значення кроку встановлюється рівним одному. Далі у циклі виконується ітерація за кожним вибраним кадром. Позиція в відеопотоці встановлюється через cv2.CAP_PROP_POS_FRAMES, після чого кадр зчитується. Якщо зчитування виявляється невдалим (наприклад, досягнуто кінеця відео), цикл завершується. Зчитаний кадр конвертується у формат RGB, що необхідно для роботи з бібліотекою MediaPipe. Кадр передається у метод process, який повертає результати аналізу. Якщо модель успішно визначає ключові точки, їхні координати x, y та z зчитуються та додаються до списку landmarks_list. У випадку, коли ключові точки не виявлено, до списку додається масив із нулями розміром (33, 3), що відповідає кількості точок та їх координат. Такий підхід гарантує послідовність у форматі даних, навіть якщо об'єкт не був знайдений. Після завершення ітерації метод для зчитування відео закривається за допомогою методу release, а сформований масив ключових точок повертається у вигляді об'єкта. Його розмірність становить (SEQUENCE_LENGTH, 33, 3), де кожен елемент відповідає набору тривимірних координат для окремого кадру.

Наступний фрагмент коду виконує структурну ідентифікацію моделі на основі підходу LSTM:

```
def create_lstm_model():
    model = Sequential()
    model.add(LSTM(64, return_sequences=True, input_shape=(
SEQUENCE_LENGTH, 33 * 3)))
    model.add(Dropout(0.3))
    model.add(BatchNormalization())
    model.add(LSTM(128, return_sequences=True))
    model.add(Dropout(0.3))
    model.add(BatchNormalization())
    model.add(LSTM(256, return_sequences=False))
    model.add(Dropout(0.4))
```

```
    model.add(BatchNormalization())
    model.add(Dense(512, activation='relu'))
    model.add(Dropout(0.4))
    model.add(Dense(len(CLASSES_LIST), activation="softmax"))
    model.summary()
    return model
```

Функція create_lstm_model реалізує архітектуру нейронної мережі для класифікації послідовностей даних ключових точок, отриманих із бібліотеки MediaPipe. Модель побудована з використанням рекурентних шарів LSTM та доповнена шарами Dropout і BatchNormalization для покращення загальної продуктивності. Архітектура моделі базується на кількох послідовних шарах. Початковий шар LSTM із 64 нейронами відповідає за вилучення базових часових залежностей із вхідних даних. Вхідний розмір даних визначається як кількість кадрів у послідовності помножена на кількість координат ключових точок. Використання параметра return_sequences=True дозволяє передавати весь часовий контекст наступним LSTM-шарам. Для запобігання перенавчанню після кожного LSTM-шару застосовується шар Dropout, який випадковим чином відключає частину нейронів під час тренування, а також використовується шар BatchNormalization для стабілізації навчання та зменшення зміщення даних.

Другий LSTM-шар із 128 нейронами відповідає за вилучення більш складних залежностей. Використання параметра return_sequences=True зберігає часову структуру, що дозволяє наступному шару обробляти вже детальніший часовий контекст. Аналогічно, шари Dropout і BatchNormalization підвищують стабільність та узагальнення моделі. Третій шар LSTM має 256 нейронів і завершує вилучення часових ознак. У цьому шарі return_sequences=False використовується для передачі лише останнього стану LSTM до наступних шарів, перетворюючи часовий контекст у статичний вектор ознак. Завдяки щільно пов'язаному шару із 512 нейронами та активацією ReLU, модель створює компактне представлення даних, що

підвищує якість класифікації. Завершальний вихідний шар має стільки нейронів, скільки класів визначено у змінній CLASSES_LIST, і використовує активацію softmax для отримання ймовірностей належності до кожного класу. Модель, створена цією функцією, є інструментом для аналізу послідовностей даних ключових точок, що дозволяє виконувати класифікацію поз художніх гімнасток.

Архітектура моделі яка була побудована за використанням функції create_lstm_model враховуючи параметри набору даних наведена у таблиці 3.3.

Таблиця 3.3 - Архітектура моделі нейронної мережі LSTM

Шар	Вихідна форма	Кількість параметрів
LSTM	(None, 30, 64)	41,984
Dropout	(None, 30, 64)	0
BatchNormalization	(None, 30, 64)	256
LSTM	(None, 30, 128)	98,816
Dropout	(None, 30, 128)	0
BatchNormalization	(None, 30, 128)	512
LSTM	(None, 256)	394,240
Dropout	(None, 256)	0
BatchNormalization	(None, 256)	1,024
Dense	(None, 512)	131,584
Dropout	(None, 512)	0
Dense	(None, 4)	2,052

Загальна кількість параметрів 670468 збалансована для забезпечення ефективного навчання. Ця структура дозволяє моделі успішно класифікувати послідовності кадрів, використовуючи їхні просторово-часові характеристики. Завдяки переходу від аналізу необроблених зображень до використання координат ключових точок вдається значно зменшити обсяг оброблюваних даних та підвищити стійкість моделі до зовнішніх шумів і варіацій.

Процес навчання моделі LSTM був організований із застосуванням механізму ранньої зупинки. Параметри моніторингу були налаштовані на функцію втрат для валідаційного під набору даних, а мінімізація цієї функції використовувалася як критерій зупинки навчання. Якщо функція втрат не покращується протягом 10 послідовних епох, навчання припиняється. Після закінчення навчання відбувається збереження найкращих ваг моделі. Модель була скомпільована із застосуванням категоріальної крос-ентропії як функції втрат. Як оптимізатор використовувався метод Adam, а оцінка ефективності виконувалася за допомогою метрики точності. Навчання моделі проводилося упродовж 50 епох із розміром партії данихчотири. Дані було поділено на навчальний та валідаційний під набори у співвідношенні 75% до 25%. Результати навчання, включно з функцією втрат та точністю, зберігалися у вигляді історії для подальшого аналізу.

На рисунку 3.7 представлено графіки залежність точності та втрат протягом процесу навчання побудованої моделі на тренувальному та валідаційному наборах даних моделі LSTM.

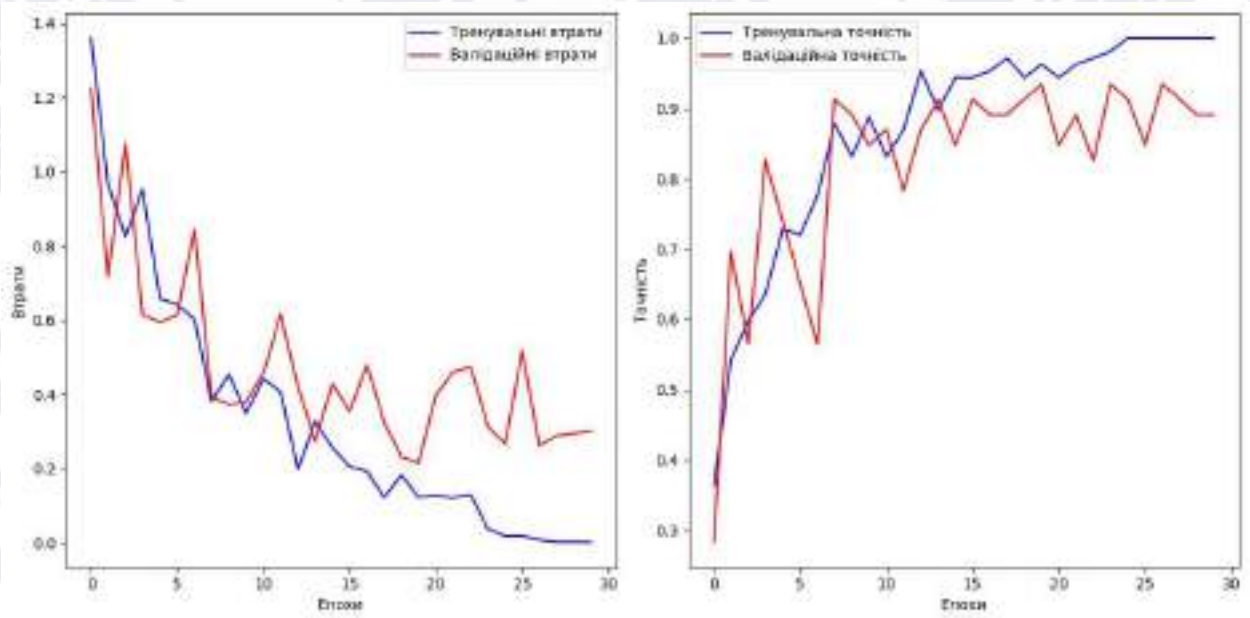

Рисунок 3.7. Залежність точності та втрат від кількості ітерацій моделі LSTM.

Результати навчання моделі свідчать про її значний потенціал у вирішенні поставленого завдання. На ранніх етапах модель демонструє

стабільне покращення як тренувальної, так і валідаційної точності, досягаючи високого показника 97.44% на 18-й епосі. Це підтверджує здатність моделі ефективно засвоювати складні патерни в даних і забезпечувати високу точність класифікації. Хоча на пізніших етапах спостерігаються певні коливання валідаційних втрат і точності, це не заважає моделі підтримувати високу продуктивність, що видно зі стабільних результатів у кількох ітераціях. Така динаміка може бути пов’язана з особливостями набору даних, де складні пози та межі між класами є викликом для класифікації. Таким чином, модель демонструє значний прогрес і перспективи.

3.6 Висновки за розділом 3

1. У процесі навчання трьох моделей були отримані результати, які відображають різні аспекти їхньої здатності до класифікації складних гімнастичних поз. Результати навчання моделей продемонстрували здатність обраних підходів ефективно класифікувати послідовності гімнастичних поз. Загальний аналіз демонструє, що всі три підходи мають потенціал для вирішення поставленого завдання, проте якість їхньої роботи залежить від архітектури моделі та параметрів навчання.

2. Модель ConvLSTM демонструє довший період адаптації в початкових епохах, із низькою точністю та значними втратами на навчальній і валідаційній вибірках. Лише на дев’ятій епосі спостерігається значне покращення точності у понад 50% і поступове зменшення валідаційних втрат. В подальшому навчанні модель демонструє зростання точності до понад 90%, хоча періодично виникають перепади валідаційної точності та втрат, що вказує на нестабільність адаптації до даних.

3. Модель LRCN показує поступове зниження тренувальних втрат і стабільне зростання тренувальної точності, що свідчить про ефективне навчання на початку тренування моделі. Проте динаміка втрат і точності свідчить про можливість перенавчання, після 18-тої епохи, що може обмежувати її здатність узагальнювати результати на нових даних.

4. Модель LSTM, що працювала виключно з послідовностями скелетних даних, відзначається кращим узгодженням між тренувальною та валідаційною точністю, що свідчить про її здатність до узагальнення та значущість ключових точок для класифікації рухів.

5. Загалом, високих результатів досягла тільки модель LSTM, яка виявилася найбільш збалансованою, демонструючи стійкість до перенавчання та високу узагальнюючу здатність. Це свідчить про ефективність використання складних архітектур для задач аналізу послідовностей кадрів гімнастичних поз, тоді як моделі ConvLSTM і LRCN потребують деякої додаткової оптимізації для більшої стійкості.

6. Загальний аналіз моделей засвідчив, що кількість кадрів і їхня якість є вагомими факторами для точності класифікації, особливо в умовах швидкої зміни поз у гімнастичних елементах. Запропоновані методи можуть бути адаптовані для інших видів спорту або задач комп'ютерного зору, що відкриває широкі можливості для подальшого використання і вдосконалення. З огляду на досягнуті результати, можна зробити висновок, що обраний підхід інтелектуального аналізу послідовностей кадрів є ефективним для задач розпізнавання динамічних рухів, і поставлені завдання роботи було виконано успішно.

ВИСНОВКИ

У рамках даної кваліфікаційної (магістерської) роботи отримано наступні результати.

1. Встановлено, що існує велике різноманіття людських дій і умов їх спостереження, які підлягають аналізу по відео. В той же час не існує універсальних методів, моделей і інформаційних технологій щодо їх аналізу.

2. Встановлено значне різноманіття у підходах до класифікацій дій у спорті за наборами ознак, що потребує відповідних методів і моделей аналізу.

3. Визначені перспективні моделі інтелектуального аналізу послідовностей кадрів гімнастичних поз – моделі класифікації на основі нейромереж типу ConvLSTM, LRCN, LSTM.

3. Зібрано набір даних, який включає чотири класи, кожен з яких представляє складну позу гімнастки під час виконання чотирьох різних елементів. Також у наборі містяться приклади виконання елемента повороту та рівноваги під час виконання яких спортсмен приймає однакову позу. Складність таких поз потребує точного розпізнавання з урахуванням кожної позиційної деталі, що відкриває додаткові можливості для застосування методів комп'ютерного зору та глибокого навчання у цьому дослідженні.

4. Було проведено аналіз існуючих методів і моделей комп’ютерного зору, зокрема згорткових та рекурентних нейронних мереж, для розпізнавання складних рухів та поз. У роботі представлено два різних підходи з поєднання згорткових та рекурентних нейронних мереж, які обробляють зображення, та підхід де послідовність скелетів спортсмена обробляється рекурентною мережею для класифікації гімнастичних елементів.

5. Виконана ідентифікація моделей для аналізу послідовностей кадрів гімнастичних поз на основі нейромереж: ConvLSTM, LRCN та LSTM впоєднанні з MobileNetV2.

6. Розроблено модель, яка продемонструвала високу точність і здатність ефективно розрізняти складні рухи, такі як повороти та елементи рівноваги, що підтверджує їхню відповідність вимогам завдання. Особливість

запропонованої моделі полягає в її здатності ідентифікувати складні пози, які виконує спортсмен та відрізняти елемент повороту від елементу рівноваги.

5. Виконана програмна реалізація та чисельне дослідження розроблених моделей аналізу послідовностей кадрів гімнастичних поз.

Узагальнюючи результати магістерської роботи, можна стверджувати, що дослідження спрямоване на вирішення актуальної задачі інтелектуального аналізу послідовностей кадрів складних поз на прикладі художньої гімнастики.

Практичне значення роботи полягає у створенні та тестуванні набору даних, адаптованого до задач розпізнавання гімнастичних поз, що містить складні та різноманітні пози спортсменів.

Отримані результати можуть слугувати основою для подальших досліджень у галузі спортивної аналітики та автоматизації суддівства.

Таким чином, магістерська робота робить внесок у розвиток методів аналізу спортивних дій, зокрема у художній гімнастиці, забезпечуючи нові можливості для практичного використання технологій машинного навчання у спортивній індустрії.

Завдання магістерської роботи виконані в повному обсязі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Neskorodieva A., Strutovskyi M., Baiev A., Vietrov O. Real-time Classification, Localization and Tracking System (Based on Rhythmic Gymnastics): in Proceedings of the IEEE 13-th International Conference on Electronics and Information Technologies, 14.11(2023). 2023. P.11–16. Doi:10.1109/elit61488.2023.10310664

2. Smirnov O., Fedorov E, Neskorodieva A., Neskorodieva T. Intellectual Classification method of Gymnastic Elements Based on Combinations of Descriptive and Generative Approach: in Proceedings of the 8-th international conference on computational linguistics and intelligent systems. Lviv, April 12–13 2024. CEUR-WS, 2024. Vol. 3664, P. 11–23. URL: https://ceur-ws.org/Vol-3664/paper2.pdf

3. Neskorodieva A. Classification of rhythmic gymnastics sport elements by video. Automation of Technological and Business Processes. 2024. 16(2), P. 103-112. URL: https://doi.org/10.15673/atbp.v16i2.2845

4. Нескородєва А. Нейромережеві методи автоматичного визначення пози людини, яка виконує вправи з художньої гімнастики Ukrainian Journal of Information Systems and Data Science. 2023. N. 1. P. 53-65. URL: https://doi.org/10.31558/2786-9482.2023.1.4

5. Liawatimena, S., Abdurahman, E., Trisetyarso, A., & Wibowo, A. (2023). Deteksi Objek dan Pengukuran Panjang serta Berat Ikan Menggunakan YOLOv3-ResNet18. *Prosiding Use Cases Artificial Intelligence Indonesia: Embracing Collaboration for Research and Industrial Innovation in Artificial Intelligence*. 2023. P. 195-223. URL: https://doi.org/10.55981/brin.668.c554

6. Prince S. J. D. Computer Vision: Models, Learning, and Inference. Cambridge: Cambridge University Press. 2012. URL: https://doi.org/10.1017/CBO9780511996504

7. What Is Computer Vision? IBM, 2021 URL: https://www.ibm.com/topics/computer-vision (дата звернення 27.11.2024)

8. Nixon M., Aguado A. Feature extraction and image processing for computer vision. Academic press. 2024. URL: https://doi.org/10.1016/B978-0-12-814976-8.00001-4

9. Kong Y., Fu Y. Human action recognition and prediction: A survey. International Journal of Computer Vision. V. 130(5). P. 1366-1401. URL: https://doi.org/10.48550/arXiv.1806.11230

10. Turaga P., Chellappa R., Subrahmanian V. S., Udrea O. Machine recognition of human activities: A survey. IEEE Transactions on Circuits and Systems for Video technology. N. 18(11), P. 1473-1488.
URL: https://doi.org/10.1109/TCSVT.2008.2005594

11. Herath S., Harandi M., Porikli F. Going deeper into action recognition: A survey. V. 60, P. 4-21. URL: https://doi.org/10.1016/j.imavis.2017.01.010

12. Vrigkas M., Nikou C., Kakadiaris I. A review of human activity recognition methods. Frontiers in Robotics and AI. V. 2. 28. URL: https://doi.org/10.3389/frobt.2015.00028
13. Rahmad N. A., As'Ari M. A., Ghazali N. F., Shahar N., Sufri, N. A. A survey of video based action recognition in sports. Indonesian Journal of Electrical Engineering and Computer Science. 2021. 11(3). P. 987-993. URL: https://doi.org/10.1016/j.imavis.2017.01.010
14. Host K., Ivasic-Kos M., Pobar M. Tracking Handball Players with the DeepSORT Algorithm. In ICPRAM. P. 593-599. URL: https://doi.org/10.5220/0009177605930599
15. Fan H., Ng H.-C., Liu S., Que Z, Niu X. Luk W. Reconfigurable Acceleration of 3D-CNNs for Human Action Recognition with Block Floating-Point Representation: in Proceedings of the 2018 28th International Conference on Field Programmable Logic and Applications (FPL), Dublin, Ireland, 2018. P. 287-2877, Doi: 10.1109/FPL.2018.00056
16. Host K., Ivašić-Kos M. An overview of Human Action Recognition in sports based on Computer Vision. Heliyon, 8(6). URL: https://doi.org/10.1016/j.fsidi.2019.200901
17. Ramasinghe S., Manosha Chathuramali K. G., Rodrigo R. Recognition of badminton strokes using dense trajectories: in Proceedings of the 7th International Conference on Information and Automation for Sustainability. Colombo. Sri Lanka. 2014. P. 1-6. Doi: 10.1109/ICIAFS.2014.7069620
18. Chen C., Shu Y., Shu K.-I., H. Zhang. WiTT:Modeling and the evaluation of table tennis actions based on WIFI signals): in Proceedings of the 24-th International Conference on Pattern Recognition (ICPR). Beijing, China. 2018. P. 3100-3107. Doi: 10.1109/ICPR.2018.8545854
19. Gupta A., Karel A., Sakthi Balan M. Discovering cricket stroke classes in trimmed telecast videos. In International Conference on Computer Vision and Image Processing. P. 509-520. Singapore: Springer Singapore. URL: https://doi.org/10.1007/978-981-15-4018-9_45
20. Al-Faris M, Chiverton J, Ndzi D, Ahmed AI. A Review on Computer Vision-Based Methods for Human Action Recognition. Journal of Imaging. 2020; 6(6):46. URL: https://doi.org/10.3390/jimaging6060046
21. Skublewska-Paszkowska, M., Lukasik E., Szydlowski B., Smolka J. Powroznik P. Recognition of tennis shots using convolutional neural networks based on three-dimensional data: in Proceedings of the 6-th International Conference on Man-Machine Interactions, ICMMI 2019, Cracow, Poland, October 2-3. 2019. P. 146-155. Springer International Publishing, Doi: 10.1007/978-3-030-31964-9_14
22. Rahmad N. A., As'Ari M. A., Soeed K., Zulkapri, I. Automated badminton smash recognition using convolutional neural network on the vision based data. In IOP Conference Series: Materials Science and Engineering. Vol. 884, No. 1, P. 012009. IOP Publishing, Doi: 10.1088/1757-899X/884/1/012009

23. Rahmad N. A., As'ari M. A. The new Convolutional Neural Network (CNN) local feature extractor for automated badminton action recognition on vision based data. In Journal of Physics: Conference Series. 2021. Vol. 1529, No. 2, P. 022021. IOP Publishing, Doi: 10.1088/1742-6596/1529/2/022021
24. Simonyan, K. Very deep convolutional networks for large-scale image recognition. 2015. URL: https://doi.org/10.48550/arXiv.1409.1556
25. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. ImageNet classification with deep convolutional neural networks. Commun. ACM 60, 6 (June 2017). 2017. P.84–90. URL: https://doi.org/10.1145/3065386
26. Cai, J., Tang, X. Rgb video based tennis action recognition using a deep historical long short-term memory. 2018. Doi: 10.48550/arXiv.1808.00845
27. Ullah M., Yamin M., Mohammed A. Attention-based lstm network for action recognition in sports: in Proc. IS&T Int'l. Symp. on Electronic Imaging: Intelligent Robotics and Industrial Applications using Computer Vision. 2021. P 302-1-302-6. URL: https://doi.org/10.2352/ISSN.2470-1173.2021.6.IRIACV-302
28. Mora S. V., Knottenbelt W. J. Deep Learning for Domain-Specific Action Recognition in Tennis. 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Honolulu. HI, USA. 2017. P. 170-178, Doi: 10.1109/CVPRW.2017.27
29. Schuldt C., Laptev I., Caputo B. Recognizing human actions: a local SVM approach," Proceedings of the 17-th International Conference on Pattern Recognition. Cambridge, UK. 2004. P. 32-36 Vol.3. Doi: 10.1109/ICPR.2004.1334462
30. Blank M., Gorelick L., Shechtman E., Irani M., R. Basri. Actions as space-time shapes. Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1. Beijing, China. 2005. Vol. 2. P. 1395-1402. Doi: 10.1109/ICCV.2005.28
31. Zhao H., Torralba A., Torresani L, Yan Z. "HACS: Human Action Clips and Segments Dataset for Recognition and Temporal Localization," 2019 IEEE/CVF International Conference on Computer Vision (ICCV). Seoul, Korea (South). 2019. P. 8667-8677. Doi: 10.1109/ICCV.2019.00876.
32. Smaira, L, Carreira, J., Noland, E., Clancy, E., Wu, A., Zisserman, A. (2020). A short note on the kinetics-700-2020 human action dataset. arXiv preprint arXiv:2010.10864.
33. Soomro K., Zamir A.R. Action Recognition in Realistic Sports Videos. In: Moeslund, T., Thomas, G., Hilton, A. (eds) Computer Vision in Sports. Advances in Computer Vision and Pattern Recognition. Springer, Cham. Doi: https://doi.org/10.1007/978-3-319-09396-3_9
34. Набір даних: «Olympic Sports Dataset». URL: http://vision.stanford.edu/Datasets/OlympicSports/ (дата звернення 27.11.2024)

35. Karpathy A., Toderici G., Shetty S., Leung T., Sukthankar R., Fei-Fei L "Large-Scale Video Classification with Convolutional Neural Networks". IEEE Conference on Computer Vision and Pattern Recognition, Columbus. OH, USA. 2014. P. 1725-1732. Doi: 10.1109/CVPR.2014.223.
36. Shao D., Zhao Y., Dai B., Lin D. "FineGym: A Hierarchical Video Dataset for Fine-Grained Action Understanding," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle. WA, USA. 2020. P. 2613-2622, Doi: 10.1109/CVPR42600.2020.00269.
37. Safdarnejad S. M., Liu X., Udpa L., Andrus B., Wood J., Craven D. Sports Videos in the Wild (SVW): A video dataset for sports analysis: in Proceedings of the 11-th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), Ljubljana, Slovenia, 2015, P. 1-7, Doi: 10.1109/FG.2015.7163105.
38. Li Y., Chen L., He R., Wang Z., Wu G., Wang, L. Multisports: A multi-person video dataset of spatio-temporally localized sports actions. In Proceedings of the IEEE/CVF International Conference on Computer Vision. P. 13536-13545. Doi: https://doi.org/10.48550/arXiv.2105.07404
39. Pobar M, Ivasic-Kos M. Active Player Detection in Handball Scenes Based on Activity Measures. Sensors. 2020. V. 20(5): 1475. URL: https://doi.org/10.3390/s20051475
40. TensorFlow: A system for large-scale machine learning. URL: https://research.google/pubs/tensorflow-a-system-for-large-scale-machine-learning/ (дата звернення 27.11.2024)
41. OpenCV - Open Computer Vision Library. URL: https://opencv.org/ (дата звернення 27.11.2024)
42. Sak H., Senior A. W., Beaufays F. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. URL: https://doi.org/10.21437/Interspeech.2014-80
43. Tran D., Bourdev L, Fergus R., Torresani L. Paluri M. Learning Spatiotemporal Features with 3D Convolutional Networks: in Proceedings of the IEEE International Conference on Computer Vision (ICCV). Santiago, Chile. 2015. P. 4489-4497. Doi: 10.1109/ICCV.2015.510
44. Jiaxin, Y., Fang, W., Jieru, Y. A review of action recognition based on convolutional neural network. In Journal of Physics: Conference Series. IOP Publishing. Vol. 1827, No. 1, p. 012138. Doi: 10.1088/1742-6596/1827/1/012138
45. Fan H., Ng H.-C., Liu S., Que Z., Niu X., W. Luk. Reconfigurable Acceleration of 3D-CNNs for Human Action Recognition with Block Floating-Point Representation: in Proceedings of the 28-th International Conference on Field Programmable Logic and Applications (FPL), Dublin, Ireland, 2018, P. 287-2877, Doi: 10.1109/FPL.2018.00056
46. Vennerød, C. B., Kjærran, A., Bugge, E. S. Long Short-term Memory RNN. URL: https://arxiv.org/abs/2105.06756 (дата звернення 27.11.2024)

47. Ivasic-Kos, M., Host, K., Pobar, M. (2021). Application of Deep Learning Methods for Detection and Tracking of Players. IntechOpen. doi: 10.5772/intechopen.96308

48. Olympics: FULL Rhythmic Gymnastics Individual All Around Final at Tokyo 2020. YouTube. URL: https://www.youtube.com/watch?v=v6ZuroWdLTs (дата звернення: 30.09.2024).

49. VEGAS Creative Software - Faster. More creative. Endless possibilities. URL: https://www.vegascreativesoftware.com/ (дата звернення: 30.09.2024).

50. Pose landmark detection guide. URL: https://ai.google.dev/edge/mediapipe/solutions/vision/pose_landmarker#configurations_options (дата звернення: 30.09.2024).

51. Xu H. et al. GHUM & GHUML: Generative 3D Human Shape and Articulated Pose Models: in Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle. WA, USA. 2020, P. 6183-6192, Doi: 10.1109/CVPR42600.2020.00622.