

НЕЙРОМЕРЕЖЕВА ІДЕНТИФІКАЦІЯ ПОЗИ СПОРТСМЕНКИ, ЯКА
ВИКОНУЄ ВПРАВИ З ХУДОЖНЬОЇ ГІМНАСТИКИ

ШИФР: POSE ESTIMATION

ЗМІСТ

Вступ	3
Розділ 1. Постановка задачі та огляд робіт	5
1.1 Постановка задачі	5
1.2. Огляд існуючих рішень	6
Розділ 2. Розробка методу попередньої обробки зображень з відео та класифікації поз людини, яка виконує вправи з художньої гімнастики	8
2.1. Розробка власного набору даних	8
2.2. Вимоги до набору даних	9
2.3. Знаходження ключових точок скелету людини методом MediaPipe	9
2.4. Розробка багатошарового перцептрона для класифікації	15
2.5. Алгоритм роботи програми	18
2.6. Технічні вимоги	19
Висновки	21
Література	22

ВСТУП

Спорт завжди був важливою частиною людського життя та культури і служив засобом розваг, змагань та фізичної активності. В останні роки індустрія спортивної аналітики швидко розвивалася завдяки досягненням технологій та аналізу даних. Зокрема, відео аналітика стала важливим інструментом для спортивних тренерів, спортсменів і аналітиків, що дозволяє зрозуміти ефективність окремих гравців і команд.

Художня гімнастика-це вид спорту, що поєднує елементи балету, танців та гімнастики, в якому спортсмени тренуються з використанням ручних снарядів, таких як скакалка, м'ячі, обручі, палиці та стрічки. Це прекрасний та елегантний вид спорту, який вимагає сили, гнучкості, координації та музичності. Однак оцінка результатів художньої гімнастики може бути складним і суб'єктивним завданням.

Одним з основних факторів, що ускладнюють оцінку художньої гімнастики, є висока складність цього виду спорту. Заняття гімнастикою включають заплутані та складні рухи, які вимагають відмінної майстерності та точності. Спортсмену потрібно буде виконувати різні рухи тіла, стрибати, балансувати, кидати і маніпулювати снарядом. Складність програми може сильно відрізнятись від спортсмена до спортсмена, і судді повинні враховувати складність при присвоєнні балів.

Ще одна з основних причин, по якій складно оцінити ефективність художньої гімнастики, - це суб'єктивний характер цього виду спорту. Естетична привабливість художньої гімнастики відіграє важливу роль у системі підрахунку очок. Спортсменів цінують не тільки за технічні навички, а й за витонченість, художню інтерпретацію і виразність. Журі повинно враховувати такі фактори, як використання музики, хореографії, костюмів та обладнання під час оцінювання. Ці суб'єктивні фактори ускладнюють стандартизацію системи нарахування балів і можуть призвести до відмінностей між оцінками суддів.

Крім того, швидкість і плавність виконання можуть ускладнити суддям фіксацію кожного руху і деталей в режимі реального часу. Відео-повтори можуть допомогти суддям отримати більш точну оцінку, але можуть зайняти більше часу. Тому важливо розробити програми для оцінки результатів спортсменів.

Мета цього дослідження-створити метод нейронної мережі для автоматичного визначення пози людини, що займається художньою гімнастикою.

Робота доповідалась на конференції 2023 IEEE 13th International Conference on Electronics and Information Technologies (ELIT) та отримало авторське свідоцтво на програму[15-16].

Розділ 1. Постановка задачі та огляд робіт

1.1 Об'єкт дослідження та постановка задачі

За сучасними правилами під час виступу художня гімнастка рівня майстер спорту має право виконати максимум 9 елементів, якими може бути стрибок, обертання та рівновага. У правилах Міжнародної федерацією гімнастики (фран. Fédération Internationale de Gymnastique, скор. FIG) [1] описано як і які саме елементи дозволено виконувати та кількість балів, за її ідеальне виконання. Змагання під егідою FIG, наприклад чемпіонати Європи, Африки, північної Америки тощо; етапи кубка Світу, кубок Світу, Олімпіади проходять за правилами FIG [2].

Особливістю оцінювання в художній гімнастиці є те що раз на 4 роки, після кожної літньої олімпіади, FIG змінює правила оцінки виступів спортсменок. У правила додаються нові елементи, відбувається переоцінка вже існуючих елементів та вимог до того, що гімнастка повинна виконати за півтора хвилинний виступ. Під час останньої зміни була додана нова бригада судів “ВИКОНАННЯ”, яка призначає штрафні бали за технічні помилки незалежно та без консультації з іншими суддями.

Оцінювання виступів гімнасток відбувається в темпі змагання. Відразу після виступу оголошується рішення судів. Це рішення є колегіальним, але ґрунтується воно на суб'єктивному візуальному аналізі виступів. Виступи спортсменок є доволі динамічними, з стрімкою зміною поз. І за долі секунд кожен суддя має візуально розпізнати недоліки виконання тих чи інших елементів та оцінити їх.

З іншого боку, останнім часом стрімко розвиваються технології комп'ютерного аналізу відеосцен в реальному часі. Такі системи впроваджуються в різноманітних сферах, зокрема і в спорті. Але, в художній гімнастиці такий інструментарій підтримки прийняття суддівських рішень ще не розроблений. Тому, виникає зацікавленість в дослідженні методів автоматичного розпізнавання та аналізу поз гімнасток в реальному часі. Ці методи будуть корисними не лише суддям. Їх також можна використовуватися

і під час тренувань з метою видачі рекомендацій стосовно корекції вправ без залучення тренерів.

1.2. Аналіз конкурентних рішень

У роботі [3] наведено алгоритм присвоєння суддівських оцінок рухам художньої гімнастики. У статті [4] представлена гібридна динамічно-статична мережа фокусування уваги з урахуванням контексту для оцінки дій у довгих відео. Зазначені статті мають розвідувальний характер і не містять практично значимих результатів. У статті [5] досліджено проблеми оцінки пози людини та класифікації діяльності, і робиться висновок про перспективність використання глибокого навчання для вирішення вказаних задач.

Оцінювання пози людини є загальновідомою проблемою комп'ютерного зору для визначення положень суглобів. Зібрані на сьогодні набори даних для вивчення поз виявилися недостатньо складними з точки зору різноманітності поз, оклюзії об'єктів і точок зору. Це робить процес розмічення пози відносно простим і обмежує застосування моделей, які на них навчені. У статті «Yoga-82: новий набір даних для детальної класифікації поз людини» автори досліджують існуючі набори даних для класифікації поз людини та пропонують новий набір даних [6]. Для більшої різноманітності поз людини автори пропонують концепцію дрібнозернистої ієрархічної класифікації поз, у якій вони формулюємо оцінку пози як завдання класифікації та пропонують набір даних Yoga-82 для розпізнавання великомасштабних поз йоги за допомогою 82 класи. Yoga-82 складається зі складних поз, де точні анотації можуть бути неможливими. Щоб вирішити цю проблему, автори надають ієрархічні мітки для поз йоги на основі конфігурації тіла пози. Набір даних містить трирівневу ієрархію, включаючи положення тіла, варіації положень тіла та фактичні назви поз. У статті також представлена точність класифікації найсучасніших архітектур згорткових нейронних мереж на Yoga-82.

Хоча на сьогодні доступні кілька наборів даних для визначення поз людини, але вони є доволі примітивними і не підходять для випадків із складним положенням людини у просторі, якими є елементи у художній гімнастиці. Тому було вирішено розробити власний набір даних для визначення поз людини, яка виконує вправи з художньої гімнастики та дослідити на ньому підходи до розпізнавання поз.

Розділ 2. Попередня обробки зображень з відео та класифікації поз людини, яка виконує вправи з художньої гімнастики.

2.1. Розробка власного набору даних

Набір даних який буде складатись з тестового та тренувального набору даних для подальшого навчання моделі класифікації спортивних елементів. При розподілені на класи було обрано п'ять найбільш популярних за виконанням елементів. Під час тестування перших натренованих моделей на цьому наборі даних було виявлено багато хибно позитивних класифікації. А саме метод часто надавав зображенню клас того чи іншого елемента, хоча спортсмен не виконує жоден з елементів набору даних. Тому було вирішено спробувати додати два додаткових класи, які характеризують найчастіших положення спортсменок і не є оцінюваними елементами. Це призвело до значного зменшення хибно позитивних помилок. На рисунку 1 зображено як виглядають п'ять оцінюваних елементів, з 1 по 5 класи, та два не оцінюваних положення, 6 і 7 класи. Приклади виконання елементів взяті з офіційних правил FIG.

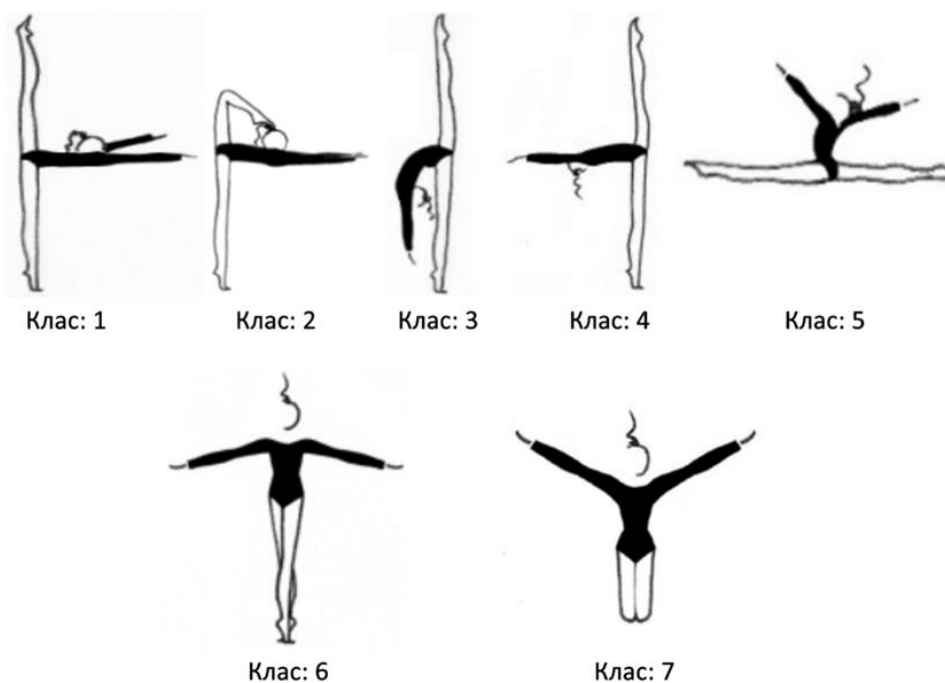


Рисунок 1. Візуалізація класів

2.2. Вимоги до набору даних

Зображення індивідуальних спортсменок, які мають займати у повний зріст не менше 50% довжини зображення. Усі частини тіла мають потрапляти на зображення. Мінімальна якість зображення може бути від 240 на 360 пікселів та повинна бути достатньою для точного визначення положення скелету спортсмена.

Ресурси використані для збору набору даних:

- Відео запис етапу кваліфікації та фіналу виступів спортсменок на змаганнях XXXII Олімпійські ігри у Токіо, викладені на платформі YouTube [7, 8]. Для навчання були відібрані стоп кадри виконання необхідних елементів.
- Фото звіти зі змагань офіційного фотографа Національної збірної України з художньої гімнастики Марії Музиченко [9].
- Фото звіти фотографа Ігоря Сахацького [10].
- Окремо для тестування роботи програми після навчання моделей було використано відео виступів Вікторії Онопрієнко з м'ячем у кваліфікації на чемпіонаті світу у місті Кітакюсю в 2021 [11].

Також у наборі даних для навчання відсутні зображення з Вікторією Онопрієнко для виключення упередженості моделі під час тестування програми. В результаті набір даних містить 5 оцінюваних елементів та 2 не оцінюваних положення. Кожен клас містить близько 90 фото у тренувальному наборі та по 6 фото для тестування під час навчання моделі.

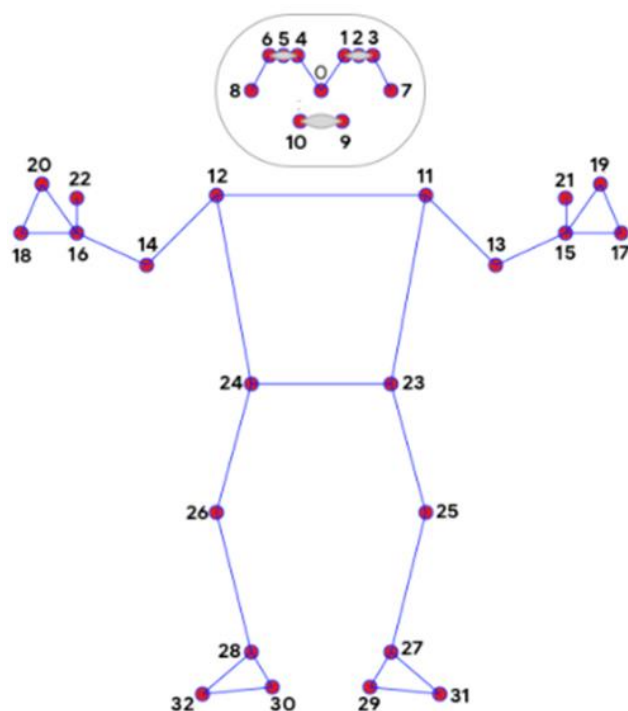
2.3. Знаходження ключових точок скелету людини методом *MediaPipe*

MediaPipe пропонує кросплатформні настроювані рішення машинного навчання для живих і потокових медіа [12]. Бібліотека містить усе необхідне для налаштування та розгортання на мобільних пристроях (Android, iOS), Інтернеті, настільних комп'ютерах, периферійних пристроях та IoT без особливих зусиль. Цей набір гнучких інструментів створений на основі бібліотеки TensorFlow Lite для найкращої впровадження машинного навчання на пристрої та продуктивності апаратного забезпечення.

Метод забезпечує продуктивність у реальному часі на більшості сучасних мобільних телефонах, настільних комп'ютерах, ноутбуках та не вимагає наявності відеокарти. Наявний програмний інтерфейс додатку на мовах програмування Python та JavaScript. Це дозволяє швидко спробувати метод для розв'язання власних задач.

MediaPipe Pose (Поза) — це рішення методом машинного навчання для відстеження пози тіла, яке визначає 33 3D-орієнтири, зображені на рисунку 2, та маску сегментації фону на всьому тілі з відеокадрів RGB.

Результатом обробки методом MediaPipe Pose одного зображення є вектор зі 132 значень, де в послідовності від першої ключової точки до 33-тої записані координати x , y , z у просторі та якість видимості. Під видимістю мається на увазі чи можливо побачити на кадрі частину тіла якій належить ключова точка. Якщо значення близьке до 1 тоді частину тіла видно якісно, чітко та однозначно. Якщо значення ближче до 0,5 тоді частину тіла, якій належить точка, може перекривати інший об'єкт (частина тіла), але сусідні по скелету точки дозволяють визначити ймовірне розташування цієї точки. Якщо значення наближається до 0 тоді частину тіла не видно зовсім або вона навіть не потрапляє на зображення.



- | | |
|--------------------|----------------------|
| 0. nose | 17. left_pinky |
| 1. left_eye_inner | 18. right_pinky |
| 2. left_eye | 19. left_index |
| 3. left_eye_outer | 20. right_index |
| 4. right_eye_inner | 21. left_thumb |
| 5. right_eye | 22. right_thumb |
| 6. right_eye_outer | 23. left_hip |
| 7. left_ear | 24. right_hip |
| 8. right_ear | 25. left_knee |
| 9. mouth_left | 26. right_knee |
| 10. mouth_right | 27. left_ankle |
| 11. left_shoulder | 28. right_ankle |
| 12. right_shoulder | 29. left_heel |
| 13. left_elbow | 30. right_heel |
| 14. right_elbow | 31. left_foot_index |
| 15. left_wrist | 32. right_foot_index |
| 16. right_wrist | |

Рисунок 2. Порядок позначень ключових точок у моделі mediapipe

Обробивши методом MediaPipe Pose власний набір даних маємо набір векторів які описують 7 класів, по 96 векторів в середньому для кожного класу. Далі це буде використане для навчання моделі класифікації елементів.

Для підключення цього методу у проект необхідно його спочатку встановити як бібліотеку, виконавши команду “pip install mediapipe” у терміналі проекту. Далі безпосередньо у файлі для виконання на мові python необхідно під'єднати бібліотеку та викликати необхідні для роботи методи для обробки зображень та візуалізації результатів обробки. Фрагмент коду на мові програмування python:

1. `import mediapipe as mp`
2. `mp_pose = mp.solutions.pose`
3. `mp_drawing = mp.solutions.drawing_utils`
4. `mp_drawing_styles = mp.solutions.drawing_styles`

Друге рішенням для тестування було обрано на базі трансформерів розроблене вченими з університету у Сідней. Результати дослідження були опубліковані у статті “ViTPose: Базові лінії простого трансформатора зору для

оцінки пози людини” [13]. У цій статті показані хороші можливості трансформаторів простого зору для оцінки пози з різних аспектів, а саме простоту структури моделі, масштабованість розміру моделі, гнучкість парадигми навчання та можливість передачі знань між моделями за допомогою простої базової моделі під назвою ViTPose. Зокрема, ViTPose використовує звичайні та неієрархічні трансформери зору як магістралі для вилучення характеристик для даного екземпляра людини та полегшений декодер для оцінки пози. Його можна збільшити від 100 мільйонів до 1 мільярду параметрів, використовуючи переваги масштабованої ємності моделі та високої паралелізмності трансформаторів, встановлюючи новий фронт Парето між пропускнуою здатністю та продуктивністю. Крім того, ViTPose є дуже гнучким щодо типу уваги, роздільної здатності введення, попереднього навчання та стратегії тонкого налаштування, а також роботи з кількома завданнями поз. Автори також емпірично демонструють, що знання великих моделей ViTPose можна легко передати малим за допомогою простого маркера знань. Експериментальні результати показують, що базова модель ViTPose перевершує репрезентативні методи на складному тесті MS COCO Keypoint Detection, тоді як найбільша модель встановлює новий рівень техніки.

Для забезпечення роботи цього методу у реальному часі необхідне використання відеокарти мінімальної потужності як у Nvidia RTX 2080. Дана модель використовує іншу анотацію для опису виявленого скелету людини, а саме, має лише 17 ключових точок. В результаті обробки зображення цим методом для кожного побудованого скелету видається вектор який описує координати x, y та точність виявлення для кожної з 17-ти ключових точок. Порядок позначень ключових точок зображено на рисунку 3.

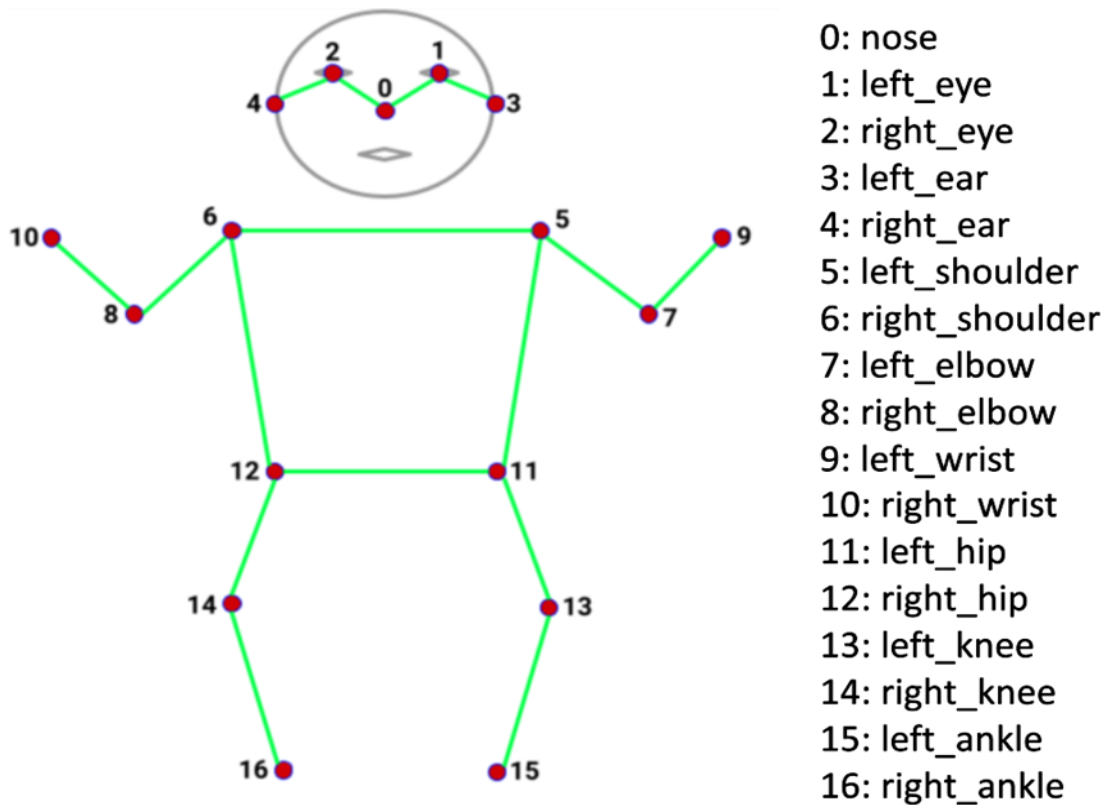


Рисунок 3. Порядок позначень ключових точок у моделі ViTPose

Для використання рішення ViTPose необхідно виконати в командному рядку перелік дії для клонування з репозиторію github та встановлення додаткових бібліотек потрібної версії, які обов'язково необхідні для роботи даного методу. Порядок виконання команд:

- `git clone https://github.com/open-mmlab/mmcv.git`
- `cd mmcv`
- `git checkout v1.3.9`
- `MMCV_WITH_OPS=1 pip install -e .`
- `cd ..`
- `git clone https://github.com/ViTAE-Transformer/ViTPose.git`
- `cd ViTPose`
- `pip install -v -e .`

Після виконання цих команд у теці проекту з'явиться дві нові теки mmcv та ViTPose. Використовуючи методи, що реалізовані у цих бібліотеках, було

розроблено два алгоритми. Перший алгоритм для обробки власного набору даних, другий для обробки відео потоку виступів спортсменів. Алгоритм обробки набору даних знаходить координати ключових точок скелету людини на зображенні та додає ці значення до набору даних, який подальшому буде використаний для навчання моделі класифікації.

Для порівняння було обрано чотири зображення, які були оброблені обома методами. Результати обробки зображені на рисунку 4.

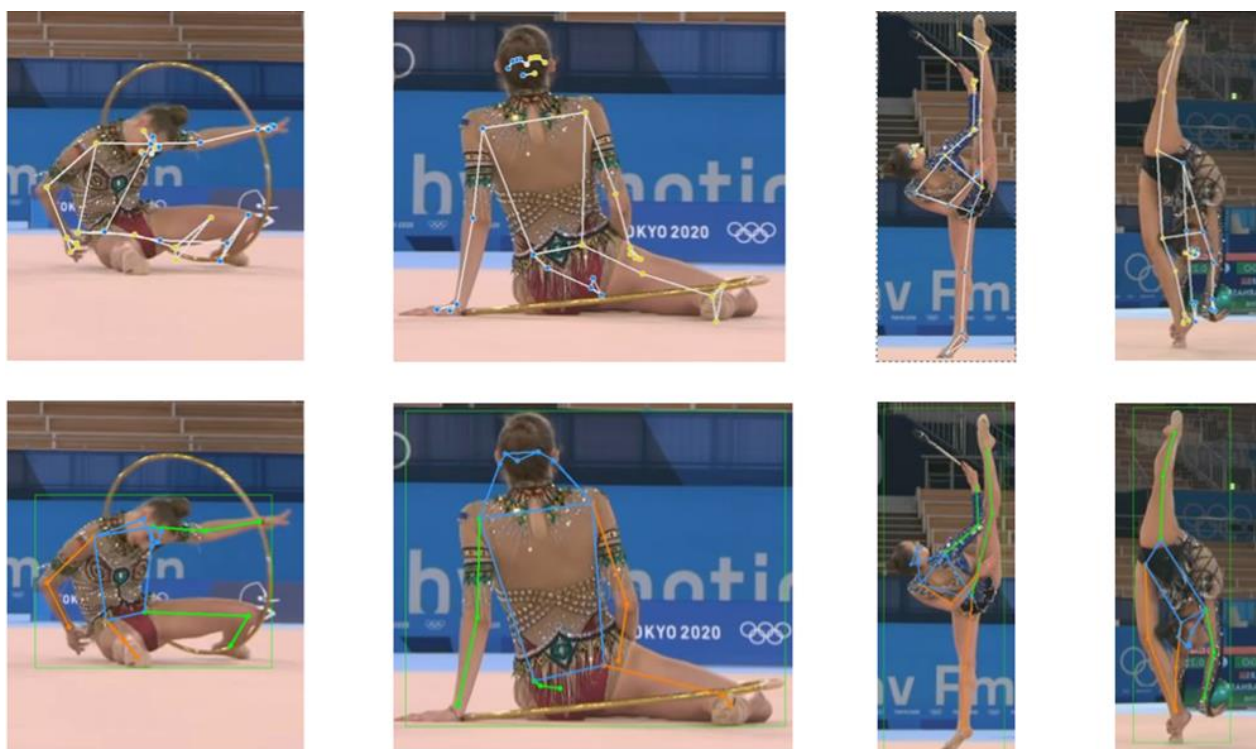


Рисунок 4. Перший ряд - mediapipe, другий - ViTPose

В першому ряду зображення оброблені методом mediapipe, в другому ряду методом ViTPose. На першому зображенні обробленому методом mediapipe зміщені ключові точки стегон, а також некоректно визначено точки на ногах. На другому зображенні, яке оброблено цим ж методом, точка правого коліна позначена ближче до стегна. На третьому зображенні помилково визначено лише пальці правої стопи. На четвертому зображенні, де спортсменка виконує елемент який належить до 3 класу, під час визначення методом mediapipe було переплутано ключові точки тулуба тіла. Через це було хибно визначено:

- праву ногу, як праву руку;
- праву руку, як ліву ногу;
- ліву ногу, як праву ногу;

Недоліком методу Mediapipe Pose є те, що якість відстеження скелету людини, під час виконання елементів пов'язаних з обертанням та групуванням в просторі, є низькою. Наприклад, під час виконання обертів модель починає плутати праві та ліві частини тіла. Ці недоліки ускладнюють подальший розвиток проекту на базі цього методу.

Результат обробки зображень, на рисунку 4, методом ViTPose є кращим за метод mediapipe та не має критичних помилок. Але цей метод вимагає більш потужного обладнання, а моделі займають більший об'єм пам'яті.

2.4. Розробка багатошарового перцептрона для класифікації.

Для класифікації спортивних елементів було вирішено побудувати багатошаровий мультикласовий перцептрон [14]. На рисунку 5 наведено 3 шаровий перцептрон, який є нерекурентною статичною багатошаровою штучною нейронною мережею, що містить два прихованих шара і вихідний шар. Для навчання багатошарового перцептрона використовується навчання на основі корекції помилок (навчання з вчителем), при цьому застосовується метод градієнтного спуску Adam.

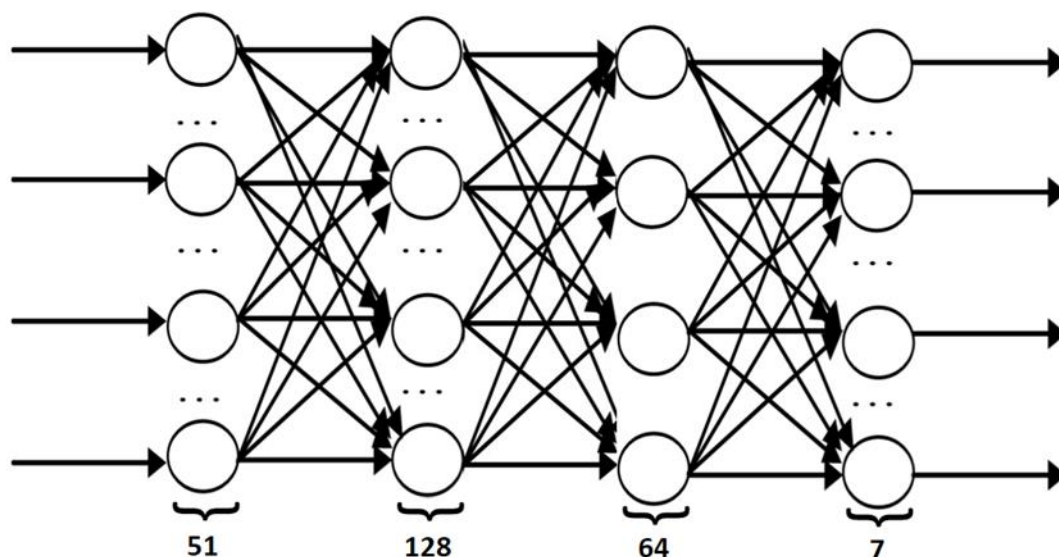


Рисунок 5. Структура 3-шарового перцептрона

Фрагмент коду визначення перцептрону для навчання моделі на мові програмування Python:

```
inputs = tf.keras.Input(shape=(51))
embedding = landmarks_to_embedding(inputs)
layer = keras.layers.Dense(128, activation=tf.nn.relu6)(embedding)
layer = keras.layers.Dropout(0.5)(layer)
layer = keras.layers.Dense(64, activation=tf.nn.relu6)(layer)
layer = keras.layers.Dropout(0.5)(layer)
outputs = keras.layers.Dense(len(class_names), activation="softmax")(layer)
model = keras.Model(inputs, outputs)
model.summary()
```

1. Побудова моделі починається з визначення вхідних даних, які мають розмір 51 нейрон.

2. Потім вхідні дані передаються в функцію `landmarks_to_embedding`, яка перетворює ключові точки скелету людини в вектор з фіксованою кількістю елементів.

3. Після цього створюються два Dense шари з 128 і 64 нейронами відповідно, з функцією активації `tf.nn.relu6`. Dense - це тип шару, який з'єднує всі нейрони попереднього шару з кожним нейроном поточного шару.

4. Між цими двома шарами вставляється Dropout шар з 0,5 ймовірністю випадкового вимкнення нейронів з метою запобігання перенавчання.

5. На виході з останнього Dropout шару створюється остаточний Dense шар з кількістю нейронів, що відповідає кількості класів. Функція активації останнього шару - "softmax", для отримання ймовірності кожного класу.

6. Під час першої ітерації навчання моделі, ваги моделі ініціалізуються випадковими значеннями.

7. Потім модель навчається на тренувальному наборі з випадковими вагами. Під час цього процесу, модель буде коригувати свої ваги за допомогою алгоритму Адам, з метою зменшення значення функції втрат на тренувальному наборі.

8. Під час навчання, модель використовує градієнтний спуск для оптимізації ваг. Мета полягає у тому, щоб знайти глобальний мінімум функції втрати, де ваги моделі будуть забезпечувати оптимальні передбачення на нових даних.

9. По закінченні кожної епохи, модель оцінюється на наборі для перевірки, щоб визначити її точність та зробити рішення про зупинку навчання за допомогою зворотного виклику ранньої зупинки (`early stopping callback`) та збереження контрольної точки (`checkpoint callback`), які були задані під час компіляції та початку навчання.

Точність навченої моделі на тестовій виборці - 0.9048. Точність свідчить про те, що модель працює на високому рівні, у більшості випадків правильно класифікуючи спортивні елементи. Цей рівень точності вказує на те, що модель було ефективно навчено класифікувати ці специфічні елементи.

Зображена на рисунку 6 матриця демонструє які класи було визначено правильно, а які хибно на тестовій виборці даних. Найбільше модель

помиляється коли бачить приклад 1 класу, плутаючи його з 4 класом. Причиною цього є те що ці два спортивні елементи є однаковими по формі положення тіла, а відрізняються лише напрямком тулуба і голови (клас 1: до підлоги, клас 4: до стелі).

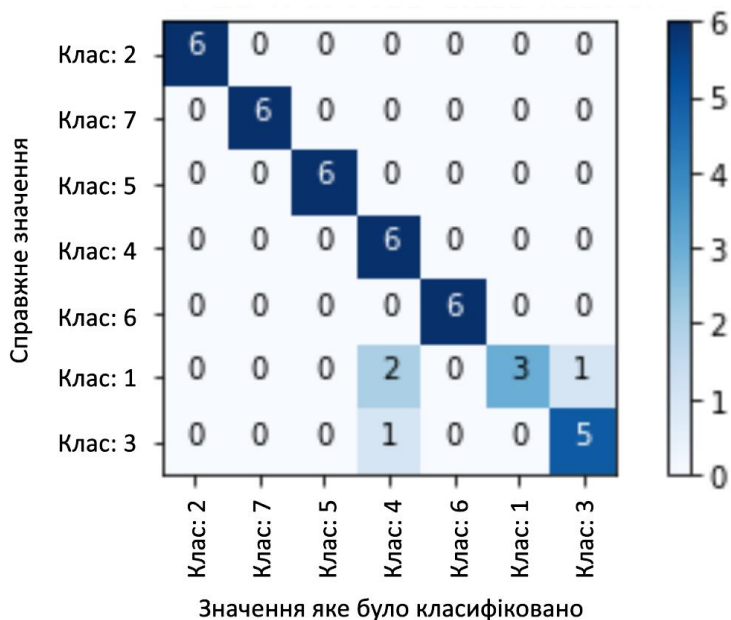


Рисунок 6. Матриця сплутувань.

2.5. Алгоритм роботи програми.

Програма працює наступним чином:

1. Відео яке необхідно обробити програма зчитує кадр за кадром.
2. Кожен кадр передається у першу модель для виявлення людини. В результаті обраний метод повертає координати обмежувальної рамки та точність виявлення кожної рамки де було виявлено людину.

3. У наступну модель для визначення положення скелету передається кадр з відео та відповідна йому анотація від першої моделі. Результати роботи першої моделі необхідні щоб звузити площу обробки зображення для другої моделі. Тобто модель буде визначати положення скелету людини лише в межах обмежувальних рамок від першої моделі. Це дозволяє пришвидшити роботу програми в даному місці. В результаті дана модель видає координати ключових точок скелету людини та якість видимості для кожної з ключових точок.

4. Модель класифікації елементів отримує для кожного кадру відповідний йому вектор значень, а результатом обробки є ймовірність для кожного з 7 класів. Серед класів обирається той клас значення ймовірність якого є найбільшим та задовольняє мінімально бажане значення.

5. Програма у верхньому лівому кутку кожного кадру візуалізує відповідне зображення класу який було класифіковано відповідним методом. Також правіше від зображення класу програма додає точність з якою метод визначив даний клас. У разі якщо точність класифікації недостатньо програма залишає початковий кадр без змін.

Приклад обробки кадру програмою зображено на рисунку 7.

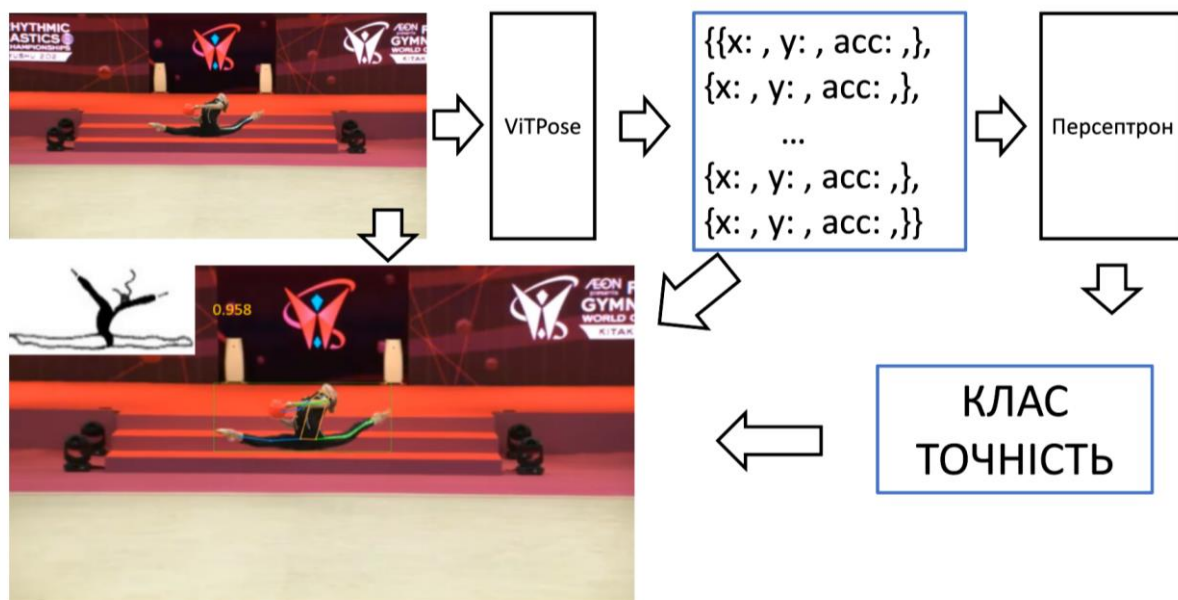


Рисунок 7. Порядок обробки кадру програмою.

2.6. Технічні вимоги.

Для написання комп'ютерної програми було обрано мову програмування Python версії 3.7.9. Технічні вимоги до програмно-апаратного забезпечення наступні:

- Операційна система: Windows 11
- Процесор: AMD Ryzen 5 3600

- Оперативна пам'ять: DDR4 16GB 3600hz
- Вбудована пам'ять: 512Gb
- Відео карта: Nvidia RTX 2080 Super - необхідна для запусків рішення на базі трансформерів у режимі реального часу.

Висновки

В результаті виконаних досліджень було зібрано та структуровано набір даних із зображенням виконання спортивних елементів. Було виявлено особливості у роботі моделей комп'ютерного зору MediaPipe Pose та ViTPose для визначення поз людини, яка виконує вправи з художньої гімнастики та обрано найкраще рішення для попередньої обробки підготовленого набору.

Головним результатом даної роботи є побудована та навчена модель для класифікації спортивних елементів, яка класифікує 7 елементів з точністю 0.9048. Програмний код написаний під час цього дослідження захищений авторським свідоцтвом [15].

Для подальшого розвитку цього методу необхідно розширити набір даних збільшивши кількість класів та прикладів для них. Також для покращення результатів класифікації спробувати використати LSTM та механізм уваги. Для підтримки актуальності оцінювання виконаних спортсменом елементів, необхідна розробка бази даних для зберігання та легкої зміни значень оцінок.

Література

1. Офіційний сайт Міжнародної федерації гімнастики. URL: <https://www.gymnastics.sport/site/> (дата звернення: 28.05.2023).
2. Правила FIG «2022 – 2024 CODE OF POINTS». URL: https://www.gymnastics.sport/publicdir/rules/files/en_2022-2024%20RG%20Code%20of%20Points.pdf (date of access: 27.05.2023).
3. Sierra-Palmeiro, E., Bobo-Arce, M., Pérez-Ferreirós, A., & Fernández-Villarino, M. A. (2019). Longitudinal Study of Individual Exercises in Elite Rhythmic Gymnastics. *Frontiers in psychology, 10*, 1496. <https://doi.org/10.3389/fpsyg.2019.01496>
4. Díaz-Pereira, M. P., Gómez-Conde, I., Escalona, M., & Olivieri, D. N. (2014). Automatic recognition and scoring of olympic rhythmic gymnastic movements. *Human movement science, 34*, 63–80. <https://doi.org/10.1016/j.humov.2014.01.001>
5. Bearman, A., & Dong, C. (2015). Human pose estimation and activity classification using convolutional neural networks. CS231n Course Project Reports.
6. Verma M., et al. Yoga-82: a new dataset for fine-grained classification of human poses. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. 2020. p. 1038-1039.
7. Olympics Gymnastics: Rhythmic Gymnastics - Individual All-Around- Qualification 1&2. Tokyo 2020. YouTube. URL: <https://www.youtube.com/watch?v=uRzmkLF8MVI> (date of access: 27.05.2023).
8. Olympics: FULL Rhythmic Gymnastics Individual All Around Final at Tokyo 2020. YouTube. URL: <https://www.youtube.com/watch?v=v6ZuroWdLTs> (date of access: 27.05.2023).
9. Альбоми зі зйомок на спортивних турнірах фотографа Марії Музиченко. URL: <https://muzychenko.photos/our-services/sports-photography> (дата звернення: 27.05.2023).
10. Портфоліо Ігоря Сахацького. URL: <https://sakhatskyi.com/portfolio/> (дата звернення: 27.05.2023).
11. Ukrainian RG Federation: Viktoriia Onopriienko Ball Qual 26,200 - World Championships Kitakyushu 2021. YouTube. URL: <https://www.youtube.com/watch?v=IKzuWUIe8Rc> (date of access: 27.05.2023).
12. GitHub - google/mediapipe: Cross-platform, customizable ML solutions for live and streaming media. GitHub. URL: <https://github.com/google/mediapipe> (date of access: 27.05.2023).
13. XU, Yufei, et al. Vitpose: Simple vision transformer baselines for human pose estimation. arXiv preprint arXiv:2204.12484, (2022).
14. Bielecki, A. (2019). Models of neurons and perceptrons: selected problems and challenges. Springer International Publishing.

15. А.с. 116622 Україна, УКРНОІВІ. Комп'ютерна програма «Pose estimation for sports (Rhythmic gymnastics)» / Х.Х.Х (Україна). №с202300058; заява 06.01.2023 ; опубл. 01.03.2023.
16. Х.Х.Х (2023). Real-time Classification, Localization and Tracking System (Based on Rhythmic Gymnastics). IEEE 13th International Conference on Electronics and Information Technologies (ELIT).