Міністерство освіти і науки України

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління
(повна назва)

Кафедра електронних обчислювальних машин
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА

Пояснювальна записка

Рівень вищої освіти другий (магістерський)

Методи генерації голосових повідомлень у системах коментування спортивних змагань
(тема)

Виконав:

студент II курсу, групи СПм-22-5

Біліченко О.О.
(прізвище, ініціали)

Спеціальність 123 «Комп’ютерна інженерія»
(код і повна назва спеціальності)

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

Освітня програма Системне програмування
(повна назва освітньої програми)

Керівник: доц. Барковська О.Ю.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри ЕОМ _____ Коваленко А.А.
(підпис) (прізвище, ініціали)

2024 р.

Харківський національний університет радіоелектроніки

Факультет комп’ютерної інженерії та управління

Кафедра електронних обчислювальних машин

Рівень вищої освіти другий (магістерський)

Спеціальність 123 «Комп’ютерна інженерія»
(код і повна назва)

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

Освітня програма Системне програмування
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
“____” _____ 20___ р.

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

студенту Біліченку Олександру Олександровичу
(прізвище, ім’я, по батькові)

1. Тема роботи Методи генерації голосових повідомлень у системах коментування спортивних змагань

затверджена наказом по університету від “ 01 ” квітня 2024 р. № 257Ст

2. Термін подання студентом роботи до екзаменаційної комісії 15 червня 2024 р.

3. Вхідні дані до роботи
датасет Deep Activity Recognition (4830 анотованих зображень),
обчислювач на базі GPU RTX A4000 (для тестування системи), Tesla T4(для навчання),
модель обробки природного мовлення GPT-4o,
архітектури нейронних мереж для виявлення об'єктів в реальному часі YOLOv8.

4. Перелік питань, що потрібно опрацювати у роботі
Огляд моделей обробки природного мовлення
Аналіз методів детектування об’єктів
Створення функціональної моделі системи
Розробка методології проведення досліджень
Проведення експериментів
Аналіз отриманих результатів

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (слайдів) 17слайдів

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

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

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Огляд моделей обробки природного	02.04.24-08.04.24	
	мовлення		
2	Розбір методів детектування об'єктів	09.04.24-16.04.24	
3	Створення функціональної моделі системи	17.04.24-22.04.24	
4	Розробка методології проведення досліджень	23.04.24-06.05.24	
5	Проведення експериментів	07.05.24-23.05.24	
6	Оформлення матеріалів кваліфікаційної роботи	24.05.24-03.06.24	
7	Подання кваліфікаційної роботи керівникові та	04.06.24-07.06.24	
	її попередній захист		
8	Подання кваліфікаційної роботи на рецензу-	08.06.24-12.06.24	
	вання		

Дата видачі завдання 01 квітня 2024 р.

Студент _____
(підпис)

Керівник роботи _____ доц.Барковська О.Ю.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка кваліфікаційної роботи: 107 с., 48 рис., 17 табл., 2 дод., 35 джерел.

МОДЕЛЬ, НЕЙРОННА МЕРЕЖА, ДЕТЕКТУВАННЯ, АРХІТЕКТУРА, СИСТЕМА, МЕТОД, МОДЕЛЬ, ПРОМПТ, КОМЕНТАР, ГЕНЕРАЦІЯ.

Метою роботи є створення системи для подійного генерування голосових повідомлень, яка може імітувати коментатора спортивних трансляцій на основі вхідних відеоданих. Вимогами до системи є можливість генерувати професійні та лаконічні голосові коментарі до спортивних подій.

В рамках роботи проаналізовано проблемну область (набуття експертності в області суддівства спортивних змагань з обраного виду спорту), існуючі технології аналізу відео, генерування тексту та аудіо; створена модель системи коментування спортивних змагань з обраного виду спорту; розроблено робочі модулі детектування та класифікації подій (в контексті гравців) на майданчику та подійного генерування голосових повідомлень. Для вдосконалення розроблених модулів системи було проведено дослідження, які показали переваги мультимодального запиту до генеративної моделі, а також визначили тип оптимізатора для найбільш точного детектування та класифікації динамічних обєктів.

ABSTRACT

Master’s thesis: 107 pages, 48 figures, 17 tables, 2 appendices, 35 sources.

MODEL, NEURAL NETWORK, DETECTION, ARCHITECTURE, SYSTEM, METHOD, MODEL, PROMPT, COMMENT, GENERATION.

The aim of the work is to create a system for the event-based generation of voice messages, which can simulate a commentator of sports broadcasts based on input video data. System requirements include the ability to generate professional and concise voice commentary for sports events.

As part of the work, the problem area (acquiring expertise in refereeing sports competitions from the chosen sport), existing technologies of video analysis, text and audio generation were analyzed; a model of the commenting system for sports competitions in the chosen sport was created; working modules for detection and classification of events (in the context of players) on the site and event generation of voice messages have been developed. To improve the developed modules of the system, studies were conducted that showed the advantages of multimodal query to the generative model, and also determined the type of optimizer for the most accurate detection and classification of dynamic objects.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

AI – штучний інтелект (англ. Artificial Intelligence)

ANN – штучна нейронна мережа (англ, Artificial Neural Network)

ASR – розпізнавання мови (англ, Automatic Speech Recognition)

CNN – згорткова нейронна мережа (англ, Convolutional Neural Network)

NLG – генерація природної мови (англ. Natural Language Generation)

NLP – обробка природної мови (англ. Natural Language Processing)

SGD – стохастичний градієнтний спуск (англ. Stochastic Gradient Descent)

SSD – твердотільний накопичувач (англ. Solid State Drive)

TTS – синтез мовлення (англ. Text-to-Speech)

ВСТУП

В сучасному світі спорту важко переоцінити роль технологій та інновацій у покращенні якості та доступності спортивних подій для глядачів. Одним із ключових аспектів цього є системи коментування спортивних змагань, які використовуються для надання інформації та аналізу глядачам у реальному часі.

За останні десятиліття спостерігається стрімкий розвиток технологій генерації голосових повідомлень (Text-to-Speech, TTS), що призвело до зростання якості синтезу голосу та його природності (Natural language processing, NLP техніки). Ці технології відкривають нові можливості для реалізації автоматизованих систем коментування спортивних подій, де голосові повідомлення генеруються за допомогою комп'ютерних алгоритмів.

Метою цієї роботи є дослідження різних методів генерації голосових повідомлень у системах коментування спортивних змагань. Вона спрямована на аналіз і порівняння ефективності та якості різних підходів до синтезу голосу в контексті спортивного коментування.

Основні завдання дослідження включають:

- класифікація подій у обраному виді спорту;
- розробка алгоритма розпізнавання емоцій;
- аналіз існуючих технологій для аналізу відео, генерування тексту та аудіо;
- аналіз методологій та їх ефективності.

Ця робота має важливе значення для подальшого розвитку систем коментування спортивних подій, оскільки вона спрямована на покращення якості та доступності голосового коментарію для широкого кола глядачів. Результати дослідження можуть бути корисними для розробників технологій спортивного коментування, телевізійних компаній, а також фанатів спорту, які шукають нові способи сприйняття спортивних подій.

У цьому контексті важливо звернути увагу на етапи розробки прототипу системи, оцінку якості генерованих голосових повідомлень та їх вплив на загальне сприйняття спортивної події глядачами. Дослідження включає в себе аналіз технічних аспектів синтезу голосу у контексті сприйняття голосового коментаря.

Робота скерована на вирішення актуальної проблеми в галузі спортивного коментування та має на меті сприяти подальшому розвитку цієї сфери за допомогою передових технологій генерації голосових повідомлень.

1 АНАЛІЗ СИСТЕМ КОМЕНТУВАННЯ ТА МЕТОДІВ ГЕНЕРАЦІЇ ГОЛОСОВИХ ПОВІДОМЛЕНЬ

1.1 Обґрунтування актуальності автоматизації коментування спортивних змагань

В сучасному світі, де штучний інтелект стрімко розвивається та вдосконалюється – питання автоматизації будь якого процесу є справою часу. За допомогою технологій AI людство кожного дня намагається зробити життя більш зручним, та точним. Не оминули зміни і сфери спорту, так наприклад, системи на базі штучного інтелекту широко використовуються в суддівстві, спортивному відеомонтуванні, прогнозуванні спортивних результатів та інших напрямках [1].

Рисунок 1.1 – Проблеми, які постають перед спортивними коментаторами

Більшість людей є прихильниками спорту та вболівальниками. Не секрет, що коментаторство спорту є невід’ємною частиною його самого з

точки зору глядачів. Але глядачі бачать лише “картинку переднього плану”, усі проблеми які постають перед коментаторами покриваються лише їх професіоналізмом. На рисунку 1.1 наведено основні складнощі та виклики такої професії, як спортивний коментатор.

Звичайно, окрім загальних проблем з якими зустрічаються коментатори спортивних подій, виникає ряд проблем, специфічних для окремого спорту. Наприклад, під час матчів по гольфу коментатору необхідно вивчити рельєф поля на якому проходить гра, а також розуміти глибоку специфіку та технічні властивості обладнання, яким користуються гравці. До того ж, через велику територію ігрового поля, не завжди є можливість спостерігати за грою, як за цільною картинкою. Як правило, у такому випадку камери для зйомки або трансляції матчу розташовані таким чином, щоб у кадр потрапило якнайбільше важливих для розуміння ходу гри моментів. Однак, через перемикання фокусу між камерами існує ризик щось пропустити.

Деякі види спорту також можуть вимагати від коментаторів витримки і навіть фізичної підготовки, таке часто зустрічається при тривалих івентах, марафонах або турнірах тривалістю кілька днів.

Що стосується основних проблем - це в першу чергу необхідність постійного фокусу в динамічних видах спорту, таких як футбол, баскетбол, волейбол, регбі, бокс та подібні. Відволікшись на секунду - можна втратити зміст події і розгубитись.

З цього випливає інший виклик - це вміння обіграти подібні моменти. Навіть якщо щось пішло не так у процесі коментування (неполадки з технікою, відсутність відео у коментатора або будь-який інший незручний момент) – глядач не повинен про це дізнатися.

Коментатор повинен бути в курсі всіх подій, що трапляються, а також знати про гру і команду все, аж до того, як правильно вимовляються прізвища гравців, щоб уникнути казусів. Підготовка до події займає вагому частину роботи коментатора.

До того ж коментатор повинен бути в міру об'єктивним і не мати упередженого ставлення до команд або гравців [2]. З досвідом формуються певні особистісні переваги, які коментатор повинен відкинути, щоб не транслювати свою прихильність до конкретної сторони. У той же час, необхідно розуміти та відчувати аудиторію. Наприклад, нейтральне ставлення до команд на національному турнірі може спричинити хвилю критики з боку вболівальників. У таких випадках коментатор повинен завжди бути на стороні своєї аудиторії.

Але найбільшою проблемою можна позначити те, що професійних коментаторів небагато і більшість спортивних подій залишається без їхньої уваги.

Більшість із цих проблем та викликів могли б бути вирішені за допомогою створення системи автоматичного коментування спортивних подій завдяки штучному інтелекту та технологіям генерації голосу. До того ж це могло б відкрити нові перспективи покращення глядацького досвіду та індивідуальної персоналізації автоматичного коментатора[3].

1.2 Огляд проблем в області подійного генерування голосових повідомлень

Для створення автоматичної системи коментування спортивних заходів необхідно розібратися з таким поняттям, як генерування подій голосових повідомлень. Для цього спочатку відповімо на запитання, що така подія?

У контексті спорту подію можна розглядати, як будь-який вимірюваний чи спостережуваний факт, який відбувається у певний час, у певному місці. Події у спорті можуть бути викликані динамічними об'єктами (спортсмен, час, погодні умови), або умовно-динамічними (об'єктами, які набувають динамічних властивостей при взаємодії з іншими об'єктами (м'яч, бейсбольна біта, шахова фігура)). Також існують статичні об'єкти (ворота, ігрове поле, доріжки для плавання). Взаємодія між об'єктами може

спричиняти певну подію. Статичні об'єкти, як правило, не можуть виступати в ролі ініціаторів подій.

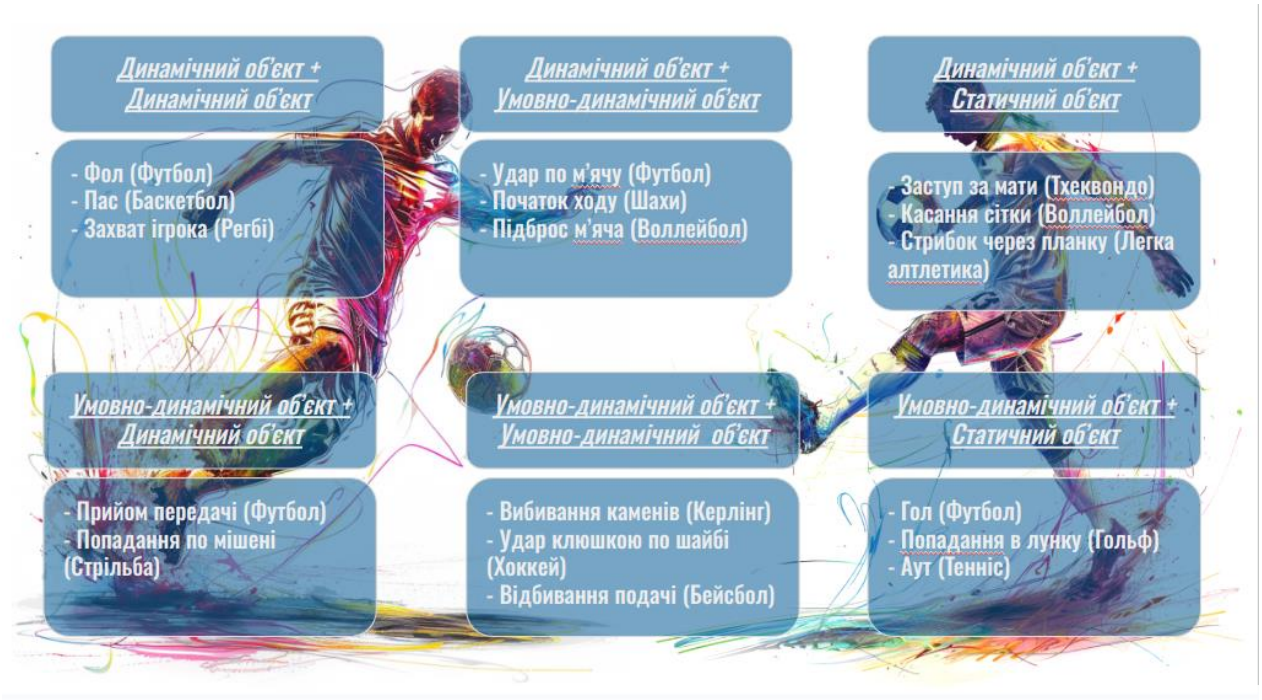


Рисунок 1.2 – Класифікація подій за типом взаємодії об'єктів

Кожна взаємодія об'єктів (рисунок 1.2) спричиняє створення певної події. У свою чергу події, що відбулися, також поділяються на позитивні, негативні та нейтральні, залежно від впливу самої події та можливих результатів, які він спричинить на настрій глядача.

Рисунок 1.3 – Сфери, в яких використовуються технології подійного генерування повідомлень

Реальні коментатори підсвідомо використовують цю класифікацію та несвідомо посилюють ефект отриманої емоції за допомогою інтонаційного забарвлення у своїй промові.

Стосовно ж автоматизованої системи – вираження емоцій одна із основних її проблем [4]. Таке зауваження характерне не для всіх сфер де використовуються технології подійної генерації голосових повідомлень (рисунок 1.3). Для деяких із них втрата природності аудіо не є критичною.

Для сукупного визначення проблеми необхідно зрозуміти загальний алгоритм роботи систем подій генерації повідомлень, представлений на рисунку 1.4.

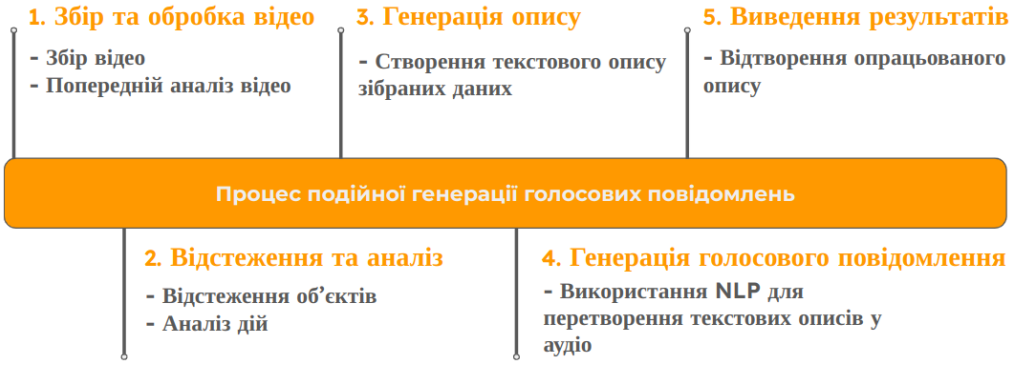


Рисунок 1.4 – Алгоритм процесу генерації подій голосових повідомлень

Результатом аналізу кожного з етапів було виявлено певні проблеми з якими, як правило, зустрічаються в процесі подійної генерації голосових повідомлень (таблиця 1.1).

Таким чином певні проблеми можуть виникати на будь якому з етапів. Якщо спробувати знайти спільні риси, то можна сказати, що вирішення цих проблем спрямоване на створення голосових коментарів такої якості, щоб глядач не міг відрізнити синтезоване аудіо від живого голоса. Кінцевим результатом має бути згенерований аудіоряд, який буде імітувати коментатора у реальному часі, та корректно описувати дії, що відбуваються на відео.

Таблиця 1.1 - Проблеми у сфері подійної генерації голосових повідомлень

Етап	Проблеми
Збір та обробка відео	- обробка затримок; - передача та обробка в реальному часі.
Відстеження та аналіз	- конкретність; - розпізнавання ключових подій.
Генерація опису	- натуральність та повнота тексту; - мовні варіації та акцент.
Генерація голосового повідомлення	- акустична подібність; - звукова ідентичність; - якість звуку.
Виведення результатів	- гармонійність картинки та подій.

Окремо треба відмітити пік невдоволення серед людей, пов’язаних з етичними проблемами, накшталт використання зловмисниками зовнішності, образів та голосів відомих людей. Тому має бути досягнута, або звукова ідентичність, або домовленість з коментатором по використанню його голоса або стилю коментування.

1.3 Аналіз існуючих систем та рішень

Після визначення проблем, які мають бути вирішені у процесі створення системи автоматичного коментування спортивних подій, імітуючи голос коментатора, необхідно проаналізувати існуючі рішення і те, як вони вирішують подібні проблеми.

Дослідження в цьому напрямку має на меті виявлення сучасних тенденцій та кращих практик у галузі генерації голосових коментарів для

спортивних подій, з метою покращення існуючих або створення цілком нової системи з урахуванням результатів дослідження.

1.3.1 Дослідження систем коментування у сфері спорту

На сьогоднішній день існує досить багато систем, так чи інакше пов'язаних з деякими етапами процесу синтезування голосових коментарів (рисунок 1.4). Деякі з опублікованих робот навіть реалізують систему, яка є повноцінною імітацією автоматичного коментатора.

Однією з таких робіт є робота, присвячена автоматичній генерації коментарів до бейсбольного матчу [4].

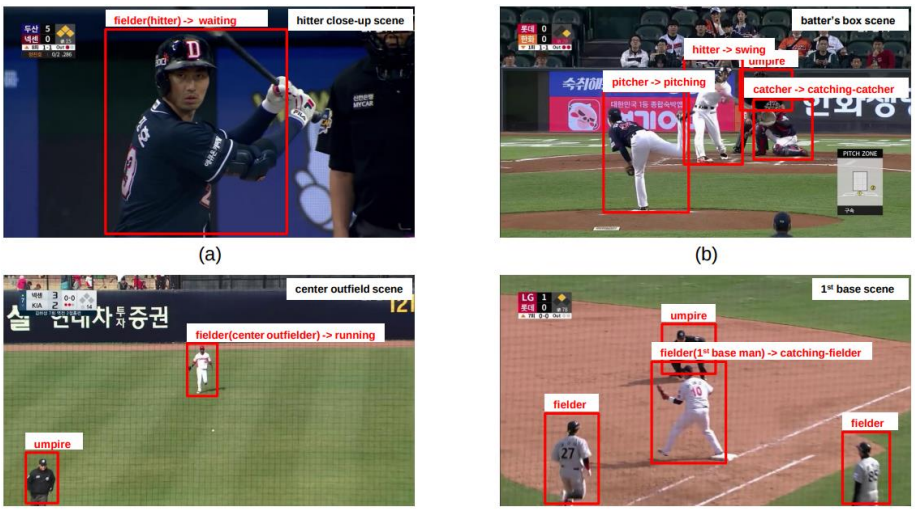


Рисунок 1.5 - Система розпізнавання ключових моментів гри

За основу даної роботи взято створення системи, яка дозволяє в реальному часі аналізувати картинку відео (рисунку 1.5) та зіставляти отримані результати з підготовленою онтологією (рисунок 1.6).

Для системи розпізнавання використовується комбінація з чотирьох моделей deep-learning (рисунок 1.7) вона дозволяє визначати відео, де знаходиться певний гравець, яка його роль у даному кадрі і що він робить в даний момент.

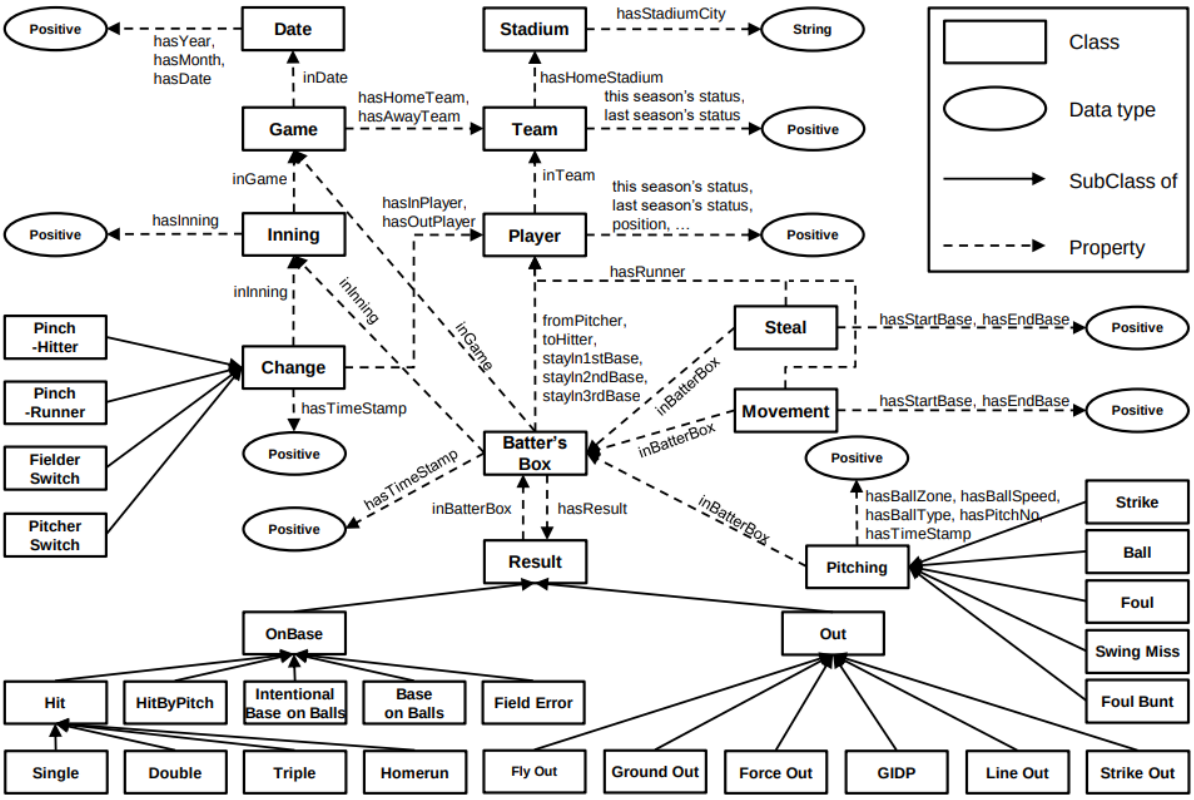


Рисунок 1.6 – Схема бейсбольної онтології, що використовується для розпізнавання подій

На рисунку 1.7 представлені такі моделі:

- класифікатор сцен (Where);
- детектор гравців (Who);
- розпізнавальний рух (Doing What);
- розпізнавач результатів подачі (Expected Result).

Класифікатор сцен визначає, у якій із локацій в даний момент часу знаходиться фокус камери. Модель вміє маркувати кадри одним із тринадцяти класів сцен. Також, слід зазначити, що до цих класів входять також зони з трибунами вболівальників та тренерські крісла, що дає можливість коментувати та аналізувати реакцію за межами ігрового поля.

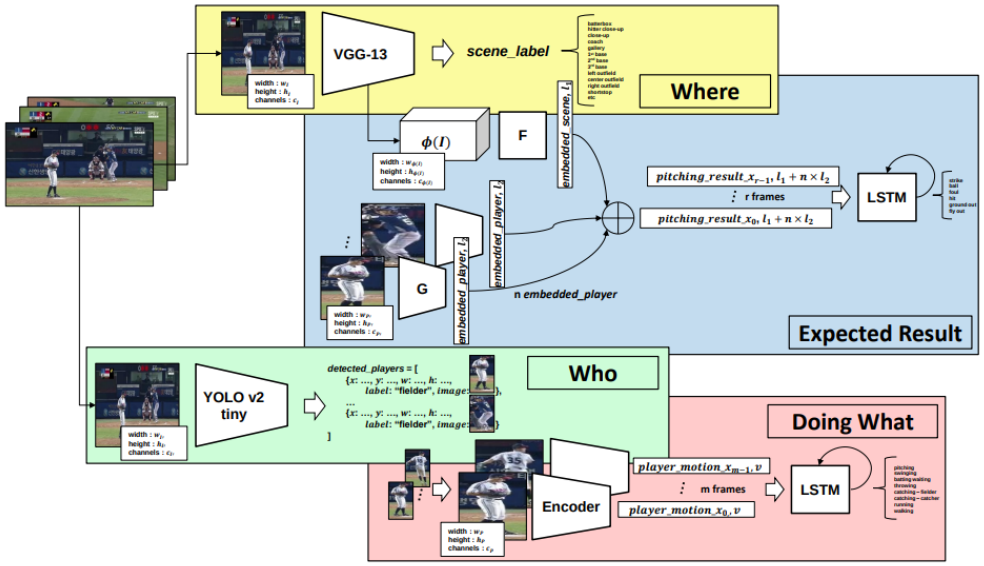


Рисунок 1.7 – Узагальнена архітектура моделей deep-learning, що використовуються для аналізу відео

Детектор гравців, у свою чергу, займається асоціацією гравців зі своїми ролями за зовнішніми ознаками. Таким чином, система розуміє, яких дій слід очікувати від конкретного гравця, знаючи його роль.

Розпізнавач рухів аналізує послідовність картинок, намагаючись виявити одну з восьми можливих послідовностей, яка буде характеризувати ключовий рух.

Що стосується розпізнавача результатів – його завдання дочекатися завершення подачі та зробити висновки, якою ключовою подією закінчилася подача на основі відео.

Підсумовуючи аналіз цього рішення, можна відзначити, що авторам роботи вдалося досягти хорошої продуктивності за рахунок комбінації чотирьох моделей в аналізі відео, з подальшим зіставленням з даними з онтології. Такий підхід чудово підходить для подальшого масштабування проекту. Опціями для розвитку можуть бути:

Розширення варіативності коментарів. Можна розширити детектор гравців, щоб визначати не тільки ролі гравців за їх позиціями, а й прізвища, на вирішення проблеми конкретності;

Додавання мультимовності. Ця проблема не може бути вирішена звичайним перекладом іншою мовою з наступним озвученням, оскільки спортивна термінологія може бути специфічною в термінах, що спричинить некоректний переклад

Додавання TTS технології для озвучення згенерованого коментаря.

Ще одну схожу систему презентували на турнірі Wimbledon у 2023 році, її розробкою займалась IBM[6]. Вони позиціонують своє рішення, не як заміну реальних коментаторів, але як можливість відкрити світ спорту для вболівальників з іншого боку, покриваючі менш популярні матчі (юніорів, представників старших ліг, спортсменів з обмеженими здатностями, тощо…). Відомо що IBM не обмежуються сферою тенісу, і продовжують покращувати досвід вболівальників і навіть працівників сфери спорту з допомогою технологій AI.

Також технологія IBM Watson знайшла застосування у сфері гольфу, дозволяючи глядачам поринути у технічні деталі та глибокий аналіз дій гравця на полі (рисунок 1.9).

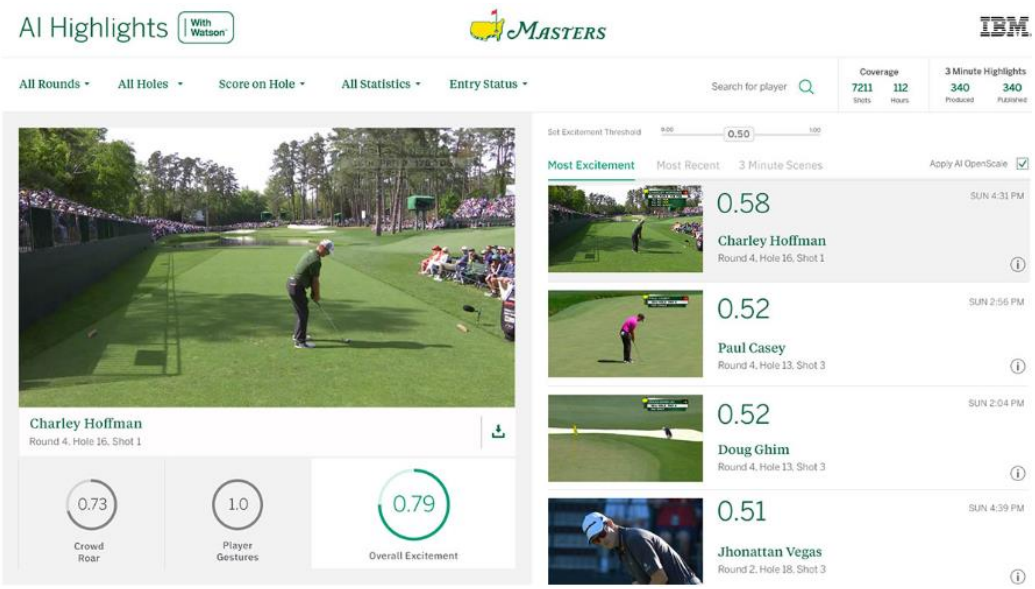


Рисунок 1.9 - Система IBM Watson Media інтегрована у веб-інтерфейс

Дана система дозволяє аналізувати відео та давати оцінку ударам гравця на основі характеристик, що вказують на якість удару. Також, крім

відео, технологія аналізує звук, у пошуках оплесків або звуку емоційного підйому публіки.

Однак IBM Watson позиціонує себе більше як платформа, ніж готове рішення для імітації спортивного коментатора. Будучи одним із першопрохідників у сфері AI – вони мають широкий набір інструментів для роботи з генеративним штучним інтелектом.

Ще однією схожою системою є Opta Vision [7]. Вона спеціалізується на аналізі даних таких видів спорту, як футбол, баскетбол, регбі та крикет. Особливу популярність цей сервіс має завдяки вмінню аналізувати результат тієї чи іншої події, залежно від умов певного моменту часу.

Процес обробки даних складається з чотирьох етапів, представлених на рисунку 1.10:

- збір даних відстеження (з камер, каналів);
- збір даних про події (оцінка аналітиків);
- синхронізація даних;
- моделювання.

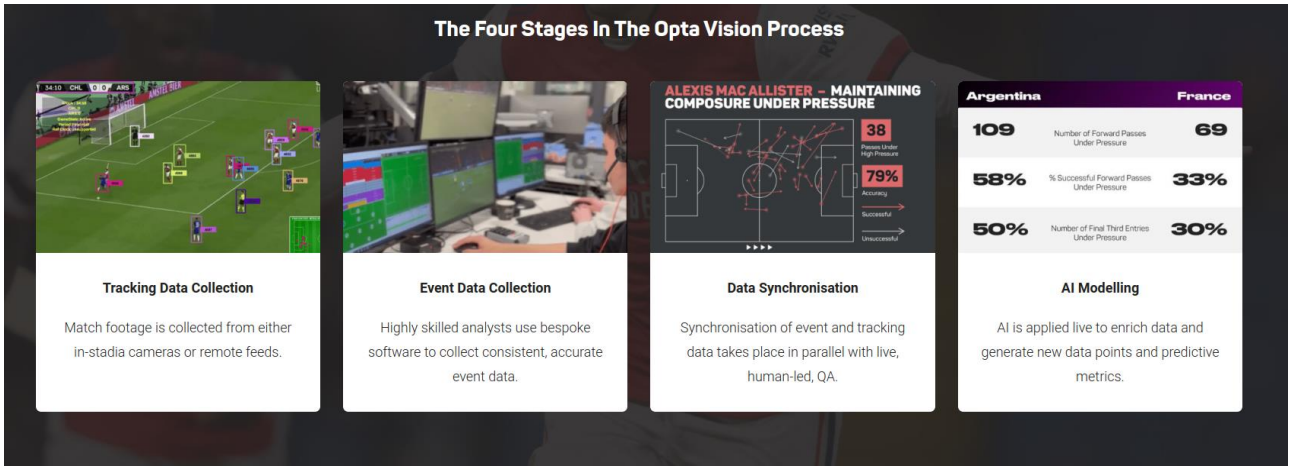


Рисунок 1.10 – Процес роботи технології Opta Vision

Присутність людини в ланцюжку системи з одного боку відкидає можливість автоматизації системи загалом, проте робить її більш точною та ефективною.

Ще однією описовою системою трекінгу рухів є технологія Tracab [8]. Ця система включає у себе дві стереокамери, які охоплюють зйомку поля цілком (рисунок 1.11). А також систему розпізнавання рухів, яка аналізує всі події на полі, які можна використовувати для автоматичного генерування коментарів.

Рисунок 1.11 – Система TRACAB (одна з пари стереокамер та зображення з візуалізацією метрик)

Також існує рішення від Sport Logiq [9], здатне аналізувати відео та додавати різні візуалізації. Дане рішення пропонує спортивну аналітику з відео в галузі хокею, футболу та регбі (рисунок 1.12).

Рисунок 1.12 – Інтерфейс системи Sport Logiq

Ця система призначена переважно для команд, ніж для вболівальників. Дозволяючи тренерам слідкувати за характеристики спортсменів і займатися більш ефективним плануванням. Проте аналітика системи могла бути використана також для подійної генерації коментарів.

До систем трекінгу можна віднести також рішення від The MoCA Project [10]. Даний проект займається різною адаптацією відео, а також вилученням та ідентифікацією об'єктів, що може бути використане для аналізу матчу та визначення ключових подій з перспективою подальшого додавання генерації повідомлень на основі отриманих даних та синтезу цих повідомлень в аудіо.

Для спортивної адаптації використовується чотири модулі розпізнавання та один для ретаргетингу відео (рисунок 1.13).

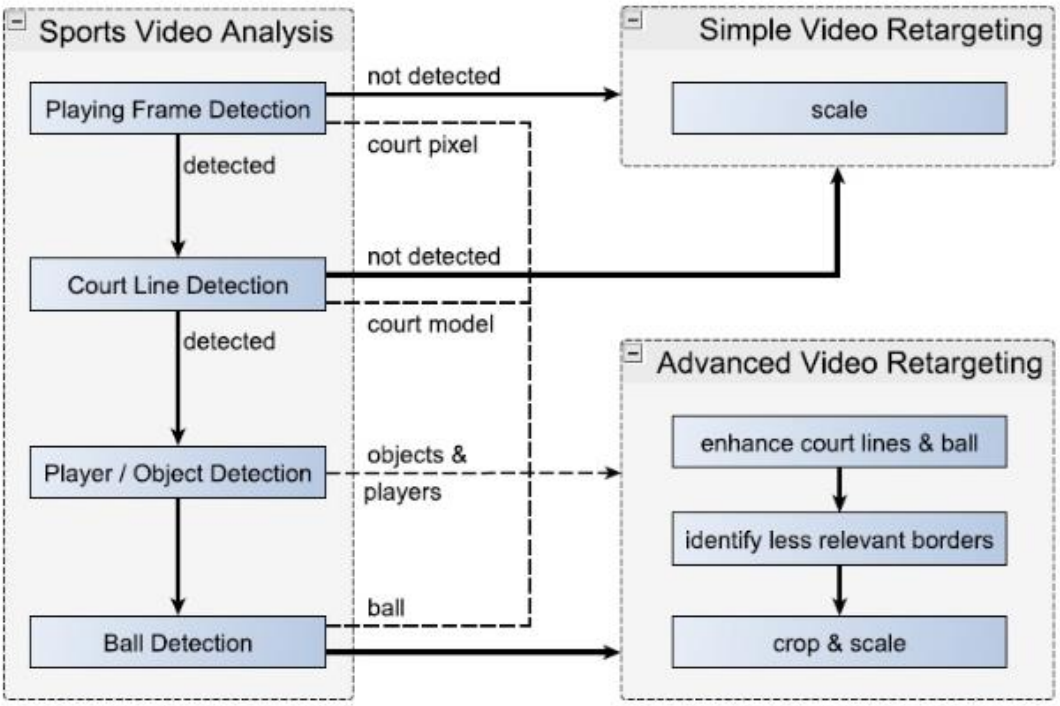


Рисунок 1.13 - Модель спортивної адаптації відео The MoCA Project

Відповідно до цієї моделі кожен етап обробляється послідовно, для більш високої продуктивності роботи моделі. Наприклад, якщо система не може ідентифікувати розмітку поля – наступний етап розпізнавання гравців на полі не буде задіяний. Це дозволяє системі бути швидкою та оперативною.

Якщо всі етапи були ідентифіковані – модуль ретаргетингу відео покращує та збільшує зображення ліній корту та м'яча.

Можливо, подібна система не є аналогом генерування коментарів на основі отриманих даних, проте ідея попереднього покращення відео могла б допомогти впоратися з проблемою фокусу на ключових предметах, посиливши точність розпізнавання об'єкта та його місцезнаходження.

1.3.2 Аналіз альтернативних галузей, які можуть використовувати системи генерування повідомлень

Як показано на рисунку 1.3, існує низка альтернативних галузей, які потенційно можуть використовувати технології генерування подій повідомлень.

У таблиці 1.2 детально розглядаються дані галузі, для знаходження подібностей та подальшого пошуку альтернативних рішень поставленого завдання.

Таблиця 1.2 – Аналіз сфер подієвого генерування голосових коментарів

Досліджувана галузь	Предмет аналізу	Формат даних, які аналізуються	Очікуємий результат
1	2	3	4
Спорт	Спортивна подія	Відео	- генерування коментарів щодо подій на відео; - синтез аудіо на основі отриманої аналітики.
Підкасти	Актуальні спеціалізовані теми.	Текст, аудіо, відео.	- генерування тексту підкасту на основі проаналізованих даних. - синтез аудіо.
Новини	Новини	Текст, аудіо, відео.	- генерування новинного контенту на основі проаналізованих даних. - синтез аудіо. - генерація, та монтаж відео.

Продовження таблиці 1.2

1	2	3	4
IVR	Дзвінки	Аудіо	- подійне генерування відповідей на запитання; - формування діалогу.
Реклама	Товар	Текст, аудіо, відео.	- генерування рекламного контенту на основі проаналізованих даних. - синтез аудіо. - генерація, та монтаж відео..
Аудіоуроки	Навички, які потребують вивчення	Текст	- озвучення підготовлених уроків; - генерація повноцінного уроку по тезам
Екскурсії	Місцеположення на карті	Відео	- генерація аудіо на основі зібраних даних про місцеположення, для створення віртуальної моделі місцевості навкруг, та надання їй вербального опису.
Аудіообробка	Аудіофайли	Аудіо	- покращення та адаптація аудіо.
Голосові Асистенти	Завдання від користувача	Текст, аудіо	- імітація поведінки асистента та виконання завдань від імені користувача.
Системи оповіщення	Непередбачувані ситуації	Відео, аудіо	- генерування аудіооповіщення з вказанням причини повідомлення, та послідовності дій, яку треба виконати.
Інструкції	Документація	Текст	- озвучення текстових інструкцій в голосові; - надання додаткових порад
Аудіокниги	Книги	Текст	- озвучення книг у текстовому форматі; - подійна стилізація голосу читача.

Кожна з перерахованих галузей представлена певним набором систем та технологій, за рахунок яких поставлені завдання реалізуються тією чи

іншою мірою. Далі будуть розглянуті сервіси і технології, які мають генеративні інструменти і заслуговують на окрему увагу.

Descript (рисунок 1.14). Даний сервіс пропонує інструменти для створення аудіо та відеоконтенту. Сервіс використовує технологію глибокого навчання для створення транскрипцій, редагування голосових копій для подкастів. З генеративного функціоналу представлені інструменти створення вижимок, описів, варіацій текстів у різних стилістиках.

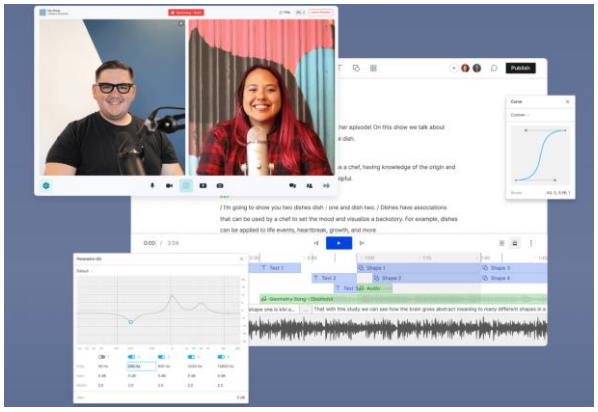


Рисунок 1.14 – Інтерфейс сервісу Descript

Resemble. AI (рисунок 1.15). Даний сервіс пропонує легку інтеграцію для зовнішніх сервісів та вміє генерувати голос з урахуванням тональностей, акцентів та настроїв. Синтезовані голоси виходять живими та правдоподібними. Додатково сервіс пропонує низку інструментів для ідентифікації діпфейків та захисту інтелектуальної власності. Це популярний інструмент для таких напрямків, як подкастинг, голосові помічники, аудіокниги і т.п.

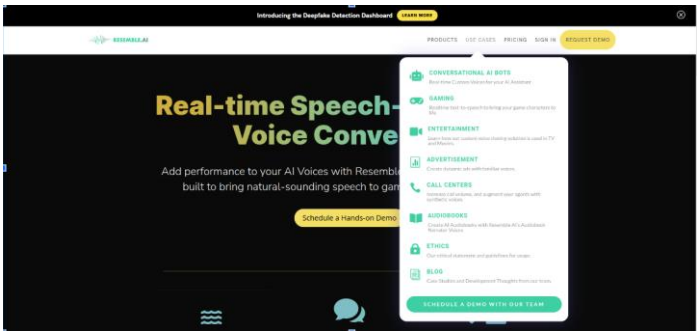


Рисунок 1.15 – Області застосування сервісу Resemble.AI

Nuance (рисунок 1.16). Даний сервіс пропонує генеративні системи IVR, які можуть замінити локальний колл-центр для бізнесу. Дані системи можуть підлаштовувати стиль мови залежно від розмови, і навіть можуть прогнозувати потреби клієнтів.

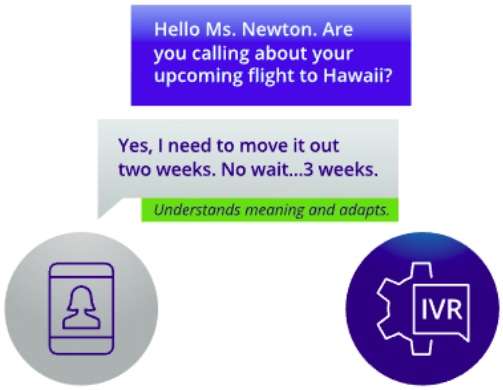


Рисунок 1.16 – Процес роботи IVR системи Nuance

Phrasee (рисунок 1.17). Сервіс для генерації та керування контентом має велику кількість темплейтів для створення контенту. У них входять: генерація email-ів, повідомлень, реклам для соціальних мереж, Інтернету, електронної комерції та реклами в цілому, а також створення статей.

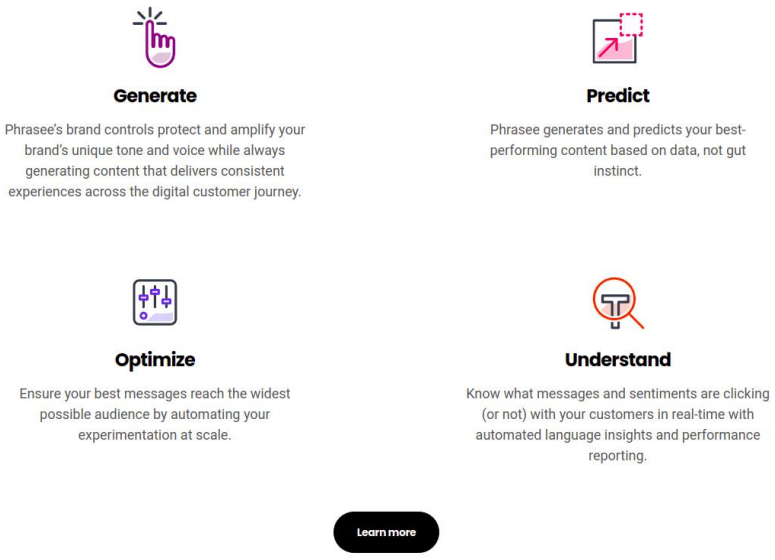


Рисунок 1.17 – Функціонал сервісу Phrasee.co

Synthesia (рисунок 1.17). Сервіс із генеративним AI, який має інструменти для генерації відео та аудіо. Сервіс використовується для створення навчальних відео, для продажу та реклами, а також для обслуговування клієнтів.

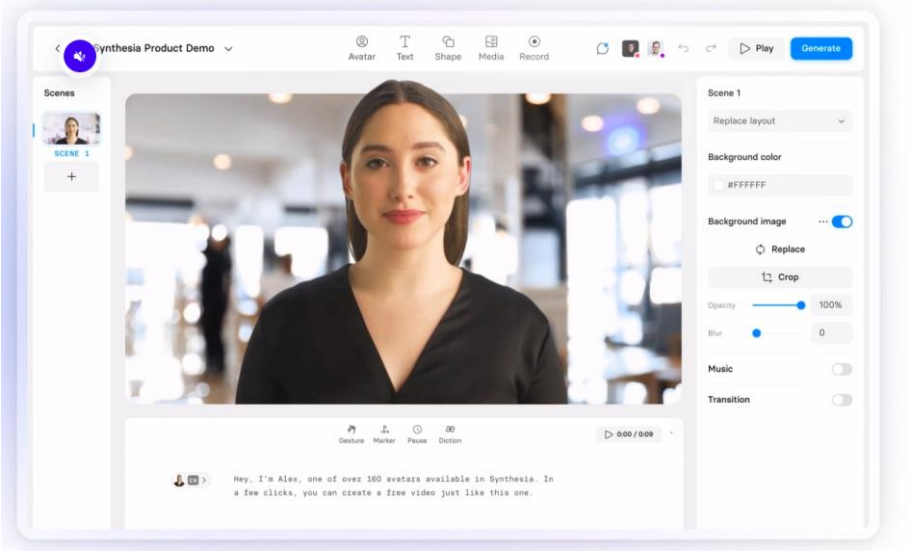


Рисунок 1.17 – Інтерфейс сервісу Synthesia

У процесі аналізу існуючих рішень було розглянуто багато сервісів, таких як: Second Spectrum SmartTracker, SAP Sports One Commentary, Channel1, Replica Studios, Synthetic Voices, WordTyne, Trint, Narrativa, Quill, Heliograf, Arria NLG, Amazon Polly, Google Cloud Text- to-Speech, Google Duplex, Pilvo, Aculab, Cepstral, Wibbitz, VidMob, Albert (Adgorithmics), Kenshoo, Bannersnack, Adext AI, Lumen5, Speechelo, NaturalReader, ISpeech, SpeechKit, ReadSpeaker, Microsoft Azure Speech Services, Response , VocaliD, Speechmatics.

Найчастіше ці послуги є технологіями TTS, у деяких випадках NLP з можливістю кастомізації. Деякі з них дозволяють у режимі реального часу створювати транскрипцію або перекладати тексти. В результаті дослідження були виявлені проблеми з таблиці 1. Як правило, це недостатньо висока якість згенерованих даних, за якими можна визначити роботу штучного інтелекту без використання зовнішніх інструментів.

1.3.3 Виявлення переваг та недоліків існуючих рішень

У процесі аналізу існуючих рішень було встановлено, що лише невелика частка існуючих рішень представляє повноцінну генеративну систему, здатну аналізувати вхідні дані, давати розгорнутий опис цих даних та синтезувати цей опис голосом.

Далі представлена порівняльна таблиця існуючих на ринку рішень, яка підкреслює їх сильні і слабкі сторони (таблиця 1.3).

Таблиця 1.3 – Порівняльна таблиця існуючих генеративних систем

Назва	Напрямок	Аналіз вхідних даних	Генерація тексту	TTS	NLP	Локалізація	Генерація відео	Real Time
1	2	3	4	5	6	7	8	9
Наукова робота	Спорт (бейсбол)	Так	Так	Ні	Ні	Ні	Ні	Не тестувалось
IBM Watson	Спорт (футбол, теніс, гольф)	Так	Так	Так	Так	Невідомо	Візуалізація	Так
Opta Vision	Спорт (футбол, баскетбол, крикет, регбі)	Так	Так	Ні	Ні	Невідомо	Ні	Так
Tracab	Спорт (футбол)	Так	Ні	Ні	Ні	Ні	Візуалізація	Так
Sport Logiq	Спорт (хоккей, футбол, регбі)	Так	Ні	Ні	Ні	Ні	Візуалізація	Так
The MoCA Project	Спорт (теніс)	Так	Ні	Ні	Ні	Ні	Візуалізація	Так
Descript	Контент менеджмент	Так	Так	Так	Так	Ні	Так	Так

Продовження таблиця 1.3

1	2	3	4	5	6	7	8	9
Resemble. AI	Сінтез мови	Так	Ні	Так	Так	Так	Ні	Так
Nuance	IVR	Так	Так	Так	Так	Невідомо	Ні	Так
Phrasee	Реклама	Так	Так	Ні	Ні	Невідомо	Ні	Ні
Synthesia	Реклама, Навчання	Так	Так	Так	Так	Так	Так	Ні

Як можна побачити – у представлених рішень є свої плюси та мінуси. Найбільш сильним рішенням серед спортивних систем є IBM Watson, ця технологія має найширший функціонал, будучи на ринку з 50-х років і зміцнивши свої позиції у сфері штучного інтелекту.

Серед альтернативних рішень лідером є сервіс Synthesia, який має великий функціонал, проте вимагає часу на створення генеративного контенту, а також не має навичок аналізу вхідного відео.

Таким чином, кожна з розглянутих систем має свої переваги і недоліки, проте можна виділити, що найбільш успішні системи аналізу спортивних подій навчають свої системи на великій кількості даних, співпрацюючи зі спортсменами та командами. Також частково функції аналізу та маркування виконують люди. Наявність людини в механізмі системи, звичайно ж, не дозволяє зробити її повністю автоматизованою. Однак це надає більшої точності отриманим результатам, що може бути чимало важливо з комерційної точки зору.

1.4 Оцінка можливостей та доцільності інтеграції з метою вдосконалення

У процесі аналізу існуючих технологій для генерації голосових повідомлень, з метою коментування спортивних подій, було виявлено, що наявні

рішення не повністю вирішують задачі по автоматизованому генеруванню аудіоряду коментатора. Кожна з розглянутих систем не покриває всі етапи, зазначені на рисунку 4. Це означає, що для реалізації автоматичного коментування спортивних подій, може бути задіяна інтеграція цих систем, або створення власного рішення, з метою вдосконалення визначених процесів.

Подібна система має надавати користувачам наступні можливості:

- можливість використання як компаніями, так і окремими користувачами;
- аналітика подій;
- генерування тексту на мові користувача;
- синтез аудіо з отриманого тексту;
- точний мастерінг звука та зображення.

Треба відзначити, що жодна зі спортивних систем не є універсальною. Не дивлячись на будь які об’єктивні причини подібної зручності, всі сервіси працюють з певною вибіркою спорту. Саме це правило дає системам бути більш точними та ефективними.

Спираючись на проведений аналіз, можна сказати що буде доцільно використовувати також у нашій системі:

- Computer Vision (для ідентифікації гравців, рухів, маркерів, тощо…);
- NLG та NLP (для генерації тексту на основі отриманої аналітики);
- TTS (для генерації голосу);
- Real-Time Data Integration (система повинна мати змогу працювати у реальному часі).

Також у нову систему потенцыйно може бути впроваджено:

- розпізнавання емоцій. Має бути розроблений алгоритм, за допомогою якого можна буде розпізнавати емоційну інтенсивність гри, для відповідного відображення цього у тоні згенерованого аудіо;

- інтеграція історичних даних. Ще однією опцією покращення глядацького досвіду є спроможність коментатора надавати історичну довідку, тим самим збагачуючі контекст висновками з минулих ігр;

- персоналізовані коментарі на основі уподобань користувача. Коментування спортивних подій може бути персоналізовано таким чином, щоб глядач відчував, ніби він вболіває за ту саму команду, що і коментатор;

- система має дотримуватись етичних принципів [11]. Коментарі не мають бути упередженими, мають поважати конфіденційність гравців, сприяти чесній грі.

1.5 Визначення мети та задач дослідження. Висування вимог до результуючої системи

Метою дослідження є створення системи для подійного генерування голосових повідомлень, яка може імітувати коментатора спортивних трансляцій на основі вхідних відеоданих. Вимогами до системи є можливість генерувати професійні та лаконічні голосові коментарі до спортивних подій.

В рамках дослідження визначаються наступні задачі:

- аналіз проблемної області (набуття експертності в області суддівства спортивних змагань з обраного виду спорту);

- огляд існуючих технологій аналізу відео, генерування тексту та аудіо;

- створення моделі системи коментування спортивних змагань з обраного виду спорту;

- розробка модулю детектування та класифікації подій (в контексті гравців) на майданчику;

- розробка модулю подійного генерування голосових повідомлень;

- розробка модулю синхронізації згенерованого аудіо та відео;

- проведення дослідження впливу оптимізатора на точність детектування динамічних об’єктів в кадрі;

- проведення дослідження впливу мультимодальності на повноту і точність згенерованого коментаря;

- проведення дослідження впливу вибору кількості ключових кадрів на синхронізацію та затримки генерації коментаря на подію у залежності від розміру батчу;
- аналіз отриманих результатів.

Додатково можна покращити результуючу систему додаванням інструментів по аналізу аудіо, для відстеження підйомів та спадів настрою натовпу. Можна додати інструменти з багатомовної підтримки, інтерактивні елементи, за допомогою котрих можна буде визначити настрій та стиль коментатора. Також корисним може бути надання можливості зворотнього зв’язку, щоб користувач міг надавати інформацію про якість коментарів і безперервно покращувати систему.

2. АНАЛІЗ ТЕХНОЛОГІЧНОГО ТА МЕТОДОЛОГІЧНОГО ПІДҐРУНТЯ ДЛЯ ПОБУДОВИ РЕСПОНСИВНОЇ СИСТЕМИ ПОДІЙНОГО РЕАГУВАННЯ ТА ГЕНЕРАЦІЇ ГОЛОСОВИХ ПОВІДОМЛЕНЬ

2.1 Аналіз технологій для створення комплексної системи розпізнавання маркерів

Як було зазначено у попередньому розділі, системи аналізу відео не є універсальними. Як правило, система готується під якийсь конкретний вид спорту, навчаючи свої моделі за певними датасетами.

Усі види спорту можна класифікувати за видом діяльності [12] (рисунок 2.1).

Рисунок 2.1 – Класифікація типів спорту за видом діяльності

Як можна спостерігати з результатів аналізу, більшість систем навчені аналізувати відео для командних видів спорту. Ймовірно, класифікація видів спорту у питанні вибору відіграє другорядну роль. На першому плані стоїть попит ринку.

У командних видах спорту тактика та стратегія є ключовими. Зважаючи на високу динамічність даних видів спорту, складання детального аналізу того, що відбувається на полі - не просте завдання. До того ж, при високому темпі гри – дуже просто пропустити важливі моменти, які на перший погляд можуть бути непомітними.

Щоб уникнути ненавмисного копіювання існуючих підходів, а також через особисті переваги, далі робота вестиметься з таким видом командного спорту, як волейбол. Наступні спостереження та аналіз будуть проводитись у контексті даного спорту.

Для цього спочатку нам необхідно отримати уявлення про загальний вид системи, якої ми б хотіли досягти. На рисунку 2.2 зображено функціональну схему цільової системи, представлену у вигляді IDE.F0 нотації [13].

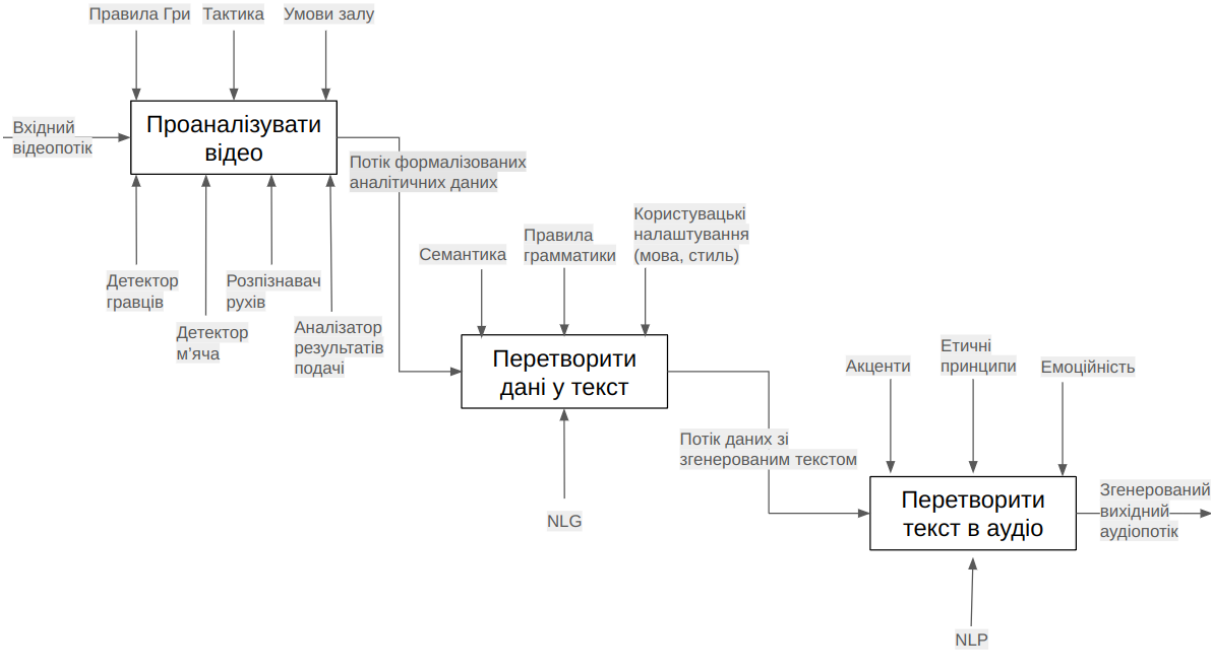


Рисунок 2.2 – IDEF0 нотація процесу перетворення вхідного відео у згенерований вихідний аудіопотік

Далі розглянемо кожен із етапів окремо, визначимо технічні вимоги до апаратури та технічного стеку.

2.1.1 Опис та вимоги до системи реєстрації динамічних об’єктів.

У реалізації системи генерації голосових повідомлень для коментування спортивних подій, ключовою складовою є система реєстрації динамічних об'єктів. Ця система відповідає за збір та аналіз вхідних даних, таких як відеопотік з камер. Вона має точно визначати та відстежувати рух спортивних об'єктів на полі гри.

Система повинна бути здатною визначати різні метрики та характеристики, які необхідні для подальшого аналізу та коментування. Тому визначимо певні вимоги до неї [14]:

- точність і швидкість. Система повинна працювати з високою точністю та швидкістю, оскільки події на спортивних майданчиках чи трасах можуть відбуватися дуже динамічно. Навіть найменші затримки чи помилки можуть вплинути на якість коментарів та загальне враження від трансляції;

- робота в реальному часі. Оскільки коментатори, як правило, працюють у форматі прямого включення - система повинна бути здатна обробляти та аналізувати дані в режимі реального часу. Це включає в себе миттєву реакцію на зміни в руху об'єктів та швидкість оновлення інформації;

- гнучкість та масштабованість. Система має працювати однаково гарно при мінімально допустимій кількості камер, та при оптимальній кількості. Варто враховувати, що різні зали можуть бути інженерно сконструйовані по різному, тому система зйомки має бути максимально гнучкою;

- надійність, та відновлюваність. Забезпечення надійності роботи системи є критично важливою, оскільки від неї залежить безперебійність коментарів під час трансляцій. Також, система повинна мати механізми для відновлення роботи після можливих помилок або випадків втрати зв'язку;

- інтегрованість з програмним забезпеченням. Система повинна бути легкою для інтеграції з іншими компонентами. Відео має безперервним потоком потрапляти до модуля аналізу для подальшого опрацювання.

Що стосується конфігурації розміщення камер та їх кількості, як згадувалося раніше все залежить від інженерної конструкції самого залу.

Якщо зал обладнаний трибунами (рисунок 2.3), або є можливість встановлення обладнання на підвісні монтажні пристрої - достатньо однієї камери для реєстрації подій матчу.

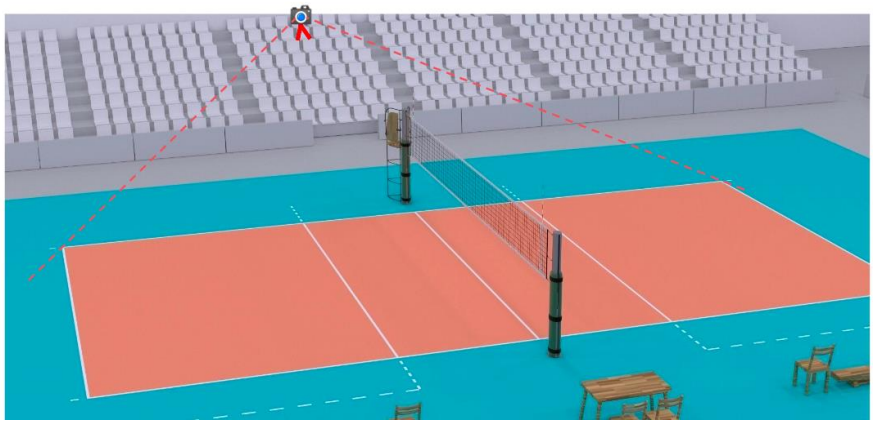


Рисунок 2.3 – Конфігурація з розташуванням обладнання вище за рівень сітки

У такому разі формулювання: “Чим вище – тим краще” не є доречним. Вибір висоти має бути обумовлений попаданням у кадр всіх кордонів поля, наскільки це можливо. Необхідно також пам'ятати, що коментатор не виконує роль судді, навіть якщо система зареєструє суворе порушення правил, але суддя не “дасть свисток” – гра продовжуватиметься. Тому замість пошуку місця для зйомки зі стовідсотковим захопленням меж поля – краще вибрати позицію з якої всі жести судді будуть видні та читані, навіть якщо заради цього доведеться пожертвувати невеликим шматком видимості поля.

Однак існують певні ризики використання єдиної камери. Камеру можуть зачепити глядачі або в неї може потрапити м'яч. У камери може раптово сісти батарейка, або перед об'єктивом може бути людина, тим самим обірвавши ресурс для подачі інформації в систему. Тому краще, щоб система включала кілька приладів для реєстрації матчу.

У випадку, якщо зал не має трибун, або спеціальних пристроїв, які могли б допомогти встановити обладнання вище за рівень сітки – камери можуть бути розташовані за крайніми задніми межами поля на відстані (рисунок 2.4), щоб у кадр потрапляли всі спортсмени та суддя.

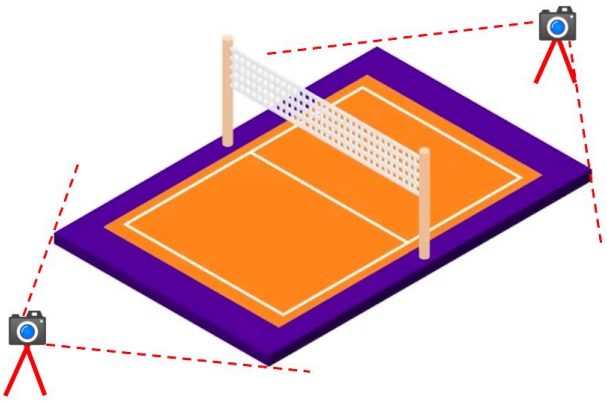


Рисунок 2.4 – Конфігурація розташування обладнання поза задніми лініями ігрового поля

Така конфігурація буде складно працювати з однією камерою, оскільки друга половина поля буде практично не видно з такої позиції (рисунок 2.5).

Рисунок 2.5 – Приклад зйомки з позиції задньої лінії без височини

З такої позиції виразно видно дії гравців однієї команди. Також можна роздивитись номери гравців для більш точної їхньої ідентифікації штучним інтелектом. Однак у деяких ситуаціях задня лінія гравців може перекривати

огляд камери на гравців передньої лінії. Тому, за будь-якої можливості, необхідно використовувати височину, щоб змінити кут огляду, тим самим полегшивши роботу аналізатору.

Вимог щодо центрування камери позаду задньої лінії немає. Однак слід пам'ятати, що в даній позиції камера буде на лінії потенційних траєкторій польоту м'яча під час атаки (рисунок 2.6).

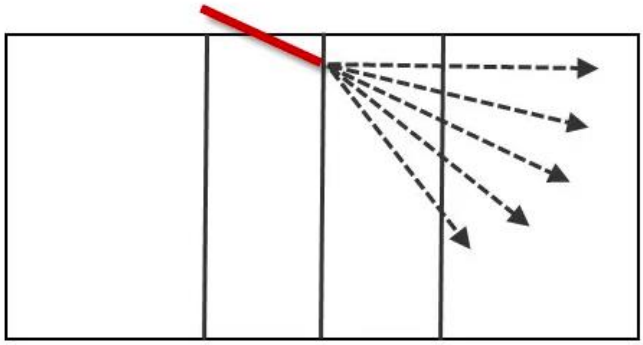


Рисунок 2.6 – Потенційні траєкторії польоту м'яча під час атаки

Виходячи з цього, альтернативним варіантом розташування камер буде розташування їх на одній лінії з сіткою (рисунок 2.7).

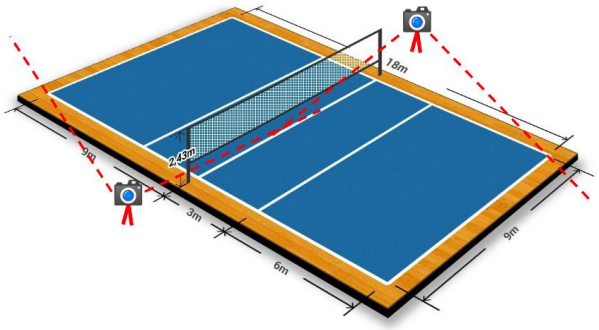


Рисунок 2.7 – Конфігурація розташування обладнання на лінії з сіткою

Ця позиція камер є найбезпечнішою, з погляду влучення м'яча. Однак, якщо зал невеликий – є ймовірність, що в кадр камери часто потраплятимуть об'єкти, що не мають відношення до матчу, наприклад, глядачі, тренер, запасні гравці.

Чим більше камер ми зможемо підключити до нашої системи, тим вища гарантія точності передачі подій у очікуваному вигляді. При цьому кожен канал відеопотоку може аналізуватись в окремому потоці (рисунок 2.8).

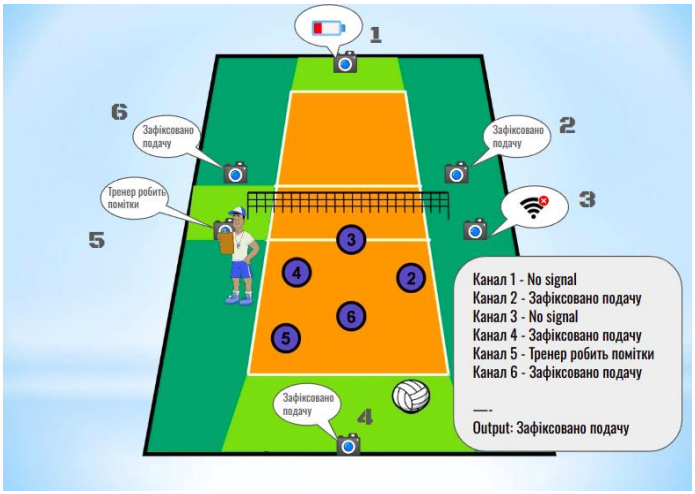


Рисунок 2.8 – Схематична імітація поканальної фіксації події (подачі м'яча)

Що стосується вимог безпосередньо до камер, для спортивної зйомки найкраще використовувати професійні відеокамери або гібридні. Професіонали спортивної зйомки рекомендують камери Cannon (професійні: Canon Cinema EOS; гібридні: EOS R5, EOS R6, EOS C70). Проте головними критеріями вибору техніки будуть:

- висока роздільна здатність (2к, 4к, 8к);
- висока частота кадрів (50 fps – 120 fps);
- наявність вбудованих фільтрів та налаштувань для адаптації під освітлення
- хороше автофокусування.

Додатково, крім зйомки відео, необхідно аналізувати і аудіодоріжку. У волейболі звук свистка може означати дозвіл подачі або зупинку гри.

Щодо торкання м'ячем стелі, що є аутом - камерам знизу буде складно визначити, чи точно було торкання. Для цього необхідно або ставити датчики вібрації на стелю і збирати дані, або встановлювати додаткову камеру на рівні стелі.

Це є труднодосяжним щодо організації. Не в кожному залі можна встановити датчик вібрації або камеру на стелю і підключити їх до загальної системи. Датчик вібрації можна повісити на сітку, щоб відстежувати торкання сітки руками. Однак визначення торкання сітки датчиком вібрації може бути неточним, наприклад якщо при боротьбі над сіткою м'яч торкнувся сітки. Камера також може підвести – будучи розташованою під певним кутом, є можливість фіксації події без фактичного її наступу.

Найбільш простим виходом із цієї ситуації буде навчання моделі розуміти жести судді. Оскільки створювана система не претендує на суддівство – її основна мета – коментувати те, що відбувається на полі. Тому навіть якщо суддя помилився – це знаходиться поза нашою зоною відповідальності.

2.1.2 Визначення технологічного стеку для обробки подій

Для обробки подій нам знадобиться розуміння, які саме події є ключовими [15]. Для цього звернемося до правил гри у волейбол та підготуємо таблицю ключових елементів, що мають роль у процесі матчу. (таблиця 2.1).

Завданням, яке стоїть перед нами є створення системи для аналізу вхідного відео, спираючись на перелічені елементи.

Оскільки ми не можемо використовувати маркерний відеоаналіз, через те, що для цього необхідно було б оснастити деякі елементи з категорії “Об'єкти” маркерами, що є не просто здійснити без попередньої підготовки. Натомість ми ідентифікуватимемо об'єкти за їх відмінними характеристиками.

Слід також окремо наголосити на необхідності запровадження моделі розпізнавання жестів суддів, оскільки вони описують вагому частину подій, що відносяться до результату розіграшу (рисунок 2.9).

Таблиця 2.1 – Ключові елементи у класичному волейболі.

Категорія	Елементи
Об'єкти	М'яч, Сітка, Волейбольний майданчик, Людина, Антена, Стійка.
Події	Початок матчу, Чотири торкання м'яча, Подача, Атака, Захист, Прийом, Аут, Очко, Перехід, Травма, Зовнішня перешкода, Передача, Перехід подачі, Партія, Вирішальна партія, Удар за підтримки, Захоплення, Заслон, Подвійне торкання, Тайм-аут , Заміна, Жеребкування, Розминка, Позиційна помилка, Блок, М'яч зачіпає сітку, М'яч у сітці, Заступ на полі суперника, Пас, Зміна сторін, Попередження, Видалення, Дискваліфікація, Затримка при подачі, М'яч не підкинутий при подачі, Помилка переходу, розстановки, Торкання сітки, Гра над сіткою на боці противника, Помилка при атаці, Заступ під час подачі, Торкання м'яча
Ролі	Гравець, Суперник, Суддя, Лінійний арбітр, Тренер, Помічник тренера, Масажист, Лікар, Капітан команди, Ліберо, Запасний гравець, Глядач.

Керуючись досвідом отриманим у результаті, аналізу існуючих рішень у цій галузі – можна зробити висновок, що найбільш оптимальним варіантом є створення кількох модулів для ідентифікації:

- детектор гравців. Його завдання визначати гравців, їхній номер (за наявності);

- розпізнавач м'яча. Також виконує роль визначення позиції м'яча у просторі матчу;

- розпізнавач жестів судді. Порівнює жести судді з подіями, що стосуються результатів розіграшу;

- ідентифікатор свистка. До його обов'язків входить аналіз аудіо для розпізнавання звуку свистка;

- детектор рухів, який відповідає за реєстрацію більшості ключових подій;

- аналізатор результатів розіграшу. Необхідний для визначення результатів після того як пролунає свисток.

Рисунок 2.9 – Значення жестів головного судді та лінійного арбітра

Послідовність взаємодії перерахованих модулів зображено на рисунку 2.10.

Для реалізації подібної системи відеоаналізу можна використовувати сучасні технології та інструменти, що дозволяють ефективно обробляти дані та обмінюватися результатами між модулями системи. У таблиці 2.2 представлений технологічний стек, який буде використовуватися для створення системи відеоаналізу.

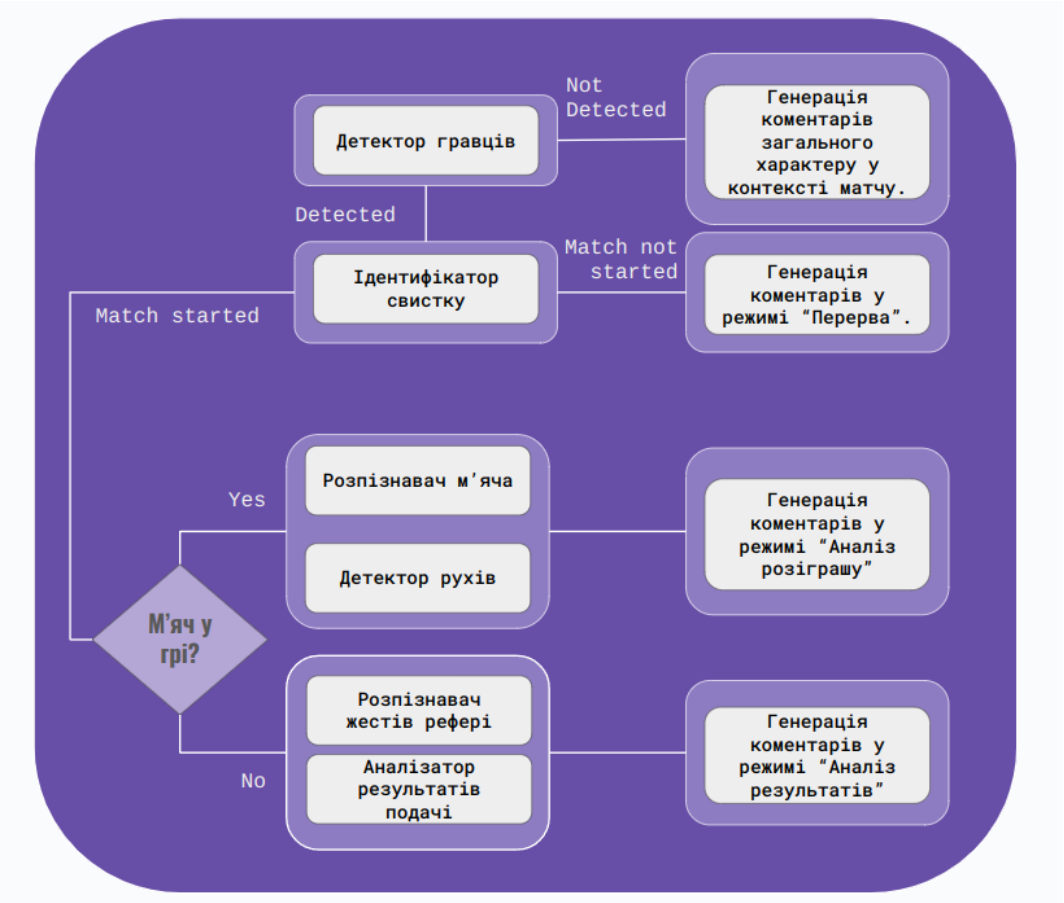


Рисунок 2.10 – Алгоритм взаємодії модулів відеоаналізу

Таблиця 2.2 – Технологічний стек для створення модулю відеоаналізу системи подійного генерування коментарів

Категорія	Технологія	Примітки
1	2	3
Обробка та Аналіз Відеоданих	OpenCV	Бібліотека для обробки відеоданих, включаючи зчитування, запис та обробку кадрів, а також виявлення об'єктів на відео.
	NumPy	Бібліотека для роботи з масивами даних, дозволяє ефективно працювати з двомірними масивами даних в контексті відеоаналізу.
Моделі машинного навчання	TensorFlow Keras	Навчання та розгортання моделей глибокого навчання. Дозволяє створювати та оптимізувати нейронні мережі для різних завдань відеоаналізу.
	TensorFlow Object Detection API	Забезпечує набір готових моделей для виявлення об'єктів на відео, включаючи детектори гравців та рухів.

Продовження таблиця 2.2

1	2	3
Алгоритми	YOLO	Швидка та ефективна модель для об'єктного виявлення, може бути використана для Детектора гравців та Детектора рухів.
	LSTM	Аналіз послідовностей дій у відео та виявлення розіграшів та ігрових ситуацій.
	CNN	Ідентифікація звуку свистку та розпізнавання жестів суддів на відео.
Середовище	Google Collab	Віртуальна машина з можливістю використання обчислювальних ресурсів графічних процесорів.
Структура даних	Protobuf	Бінарна структура даних для комунікації між модулями. Рекомендована для систем реального часу.
Кешування	Apache Kafka	Розподілена платформа потокової передачі подій. Може також використовуватись як сховище тимчасових даних, наприклад при мультиканальному процесінгу.

2.1.3 Визначення технологічного стеку для генерації голосових повідомлень

Генерація голосових повідомлень на базі отриманої інформації складається із двох етапів:

- генерація тексту, який описуватиме отримані дані у формі повідомлення коментатора;
- перетворення тексту на голос.

Також додатково необхідно створити алгоритм розпізнавання емоцій[16]. Він може бути використаний у персоналізації коментатора та знаходиться у трьох режимах:

- глядач вболіває за команду, яка починає гру зліва;
- глядач нейтрально ставиться до команд;
- глядач вболіває за команду, яка починає гру праворуч.

Для початку нам потрібно класифікувати події з таблиці 2.3 на вигляд емоцій.

Таблиця 2.3 – Класифікація подій у волейболі на вигляд емоцій з позиції команди X

Тип емоції	Подія
Радість	Аут при прийомі; Блок при захисту; Отримання очків.
Напруження	Атака; Вирішальна партія.
Здивування	Подача + гол без торкання супротивником м'яча; Атака + удар + блок; Фол + немає свистка та жестів від суддів. Відсутність фолу + свисток + жест від суддів; Понад три розпасовування від кожної з команд за розіграш.
Розчарування	Аут під час подачі; Аут при розігруванні; Аут під час атаки; Травма; Фол.
Нейтральність	Початок матчу; Передача; Перехід подачі; Партія; Тайм-аут; Заміна; Жеребкування; Розминка; Пас; Зміна сторін.
Хвилювання	Прийом; Захист; Травма.
Захоплення	Подія з категорії Напруження/Здивування з наступною миттєвою подією з категорії Радість; Понад п'ять розпасовок від кожної з команд за розіграш.

Дана класифікація є поверховою, можна гранулярно додавати комбінації подій, надаючи їм нові типи емоцій, або розширюючи існуючі.

Якщо обраний нейтральний режим - події з категорії “Розчарування” та “Радість” мають бути прокоментовані з емоцією “Напруження”.

Важливо, що емоційне забарвлення слід надавати і під час перетворення тексту в аудіо, і у процесі генерації тексту, для коректної передачі настрою коментатора.

Сопоставлення з цією класифікацією може бути використано у функції-мідлварі, яка на виході створювала б промпт для GPT моделі (рисунок 2.11).

В комбинации с GPT моделью для генерации текста ми будемо використовувати Mozilla TTS технологію для перетворення згенерованого тексту в аудіо. Вибір цієї TTS був скерований наступними перевагами:

- Open Source;
- підтримка багатьох мов та акцентів;
- проста інтеграція з Google Colab;
- широкі можливості кастомізації голосу;
- підтримка емоційних міток.

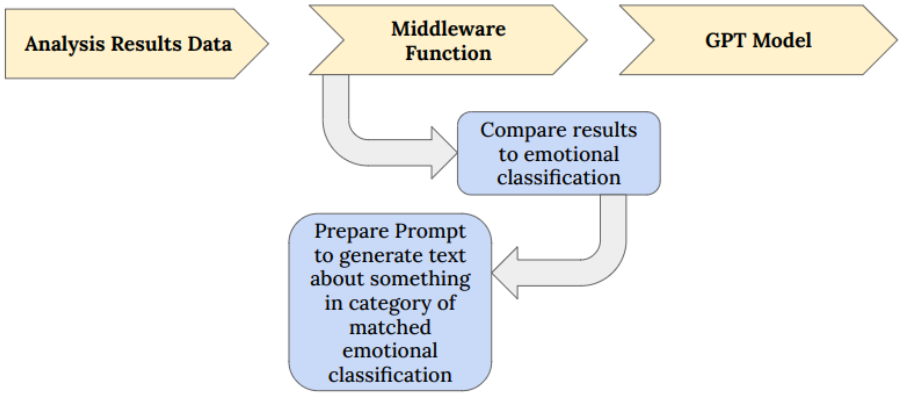


Рисунок 2.11 – Представлення логічної функції мідлвари, для генерації промпту GPT моделі

2.2 Аналіз методологій та їх ефективності в контексті поставленої задачі

Для вибору оптимального підходу до реалізації системи необхідно проаналізувати ефективність основних методологій. Існує велика кількість метрик для оцінки якості розпізнавання динамічних об'єктів (таблиця 2.4).

Таблиця 2.4 - Основні метрики оцінки ефективності методології

Метрика	Опис
1	2
Точність роботи (Accuracy):	Точність визначає відсоток правильно класифікованих об'єктів відносно загальної кількості. Ця метрика дозволяє оцінити загальну відповідність результатів класифікації.

Продовження таблиця 2.4

1	2
Повнота (Recall)	Визначає відсоток правильно виявлених позитивних об'єктів від усіх позитивних об'єктів у даних..
Точність (Precision)	Визначає відсоток правильно виявлених позитивних об'єктів від усіх виявлених об'єктів. Ці метрики особливо важливі у випадках дисбалансу класів даних та визначення коректної класифікації.
F1-міра (F1-Score):	F1-міра є гармонійним середнім між точністю і повнотою. Вона підходить для ситуацій, коли необхідно збалансувати класифікації між ними.
AUC-ROC (Area Under the Receiver Operating Characteristic Curve)	Оцінює здатність моделі правильно розподіляти позитивні та негативні класи.
Середній час обробки кадру	Ця метрика вимірює середній час, який потрібно моделі для обробки одного кадру відео. Важлива для систем реального часу, де швидкість обробки є критичною.
False Alarm Rate (FAR)	Визначає відсоток невірно класифікованих негативних об'єктів серед усіх негативних об'єктів.
False Positive Rate (FPR)	Визначає відсоток невірно класифікованих позитивних об'єктів серед усіх негативних об'єктів.
Швидкість навчання та виявлення	Ця метрика вимірює час, який потрібно моделі для навчання на наборі даних та для виявлення (інференсу) нових об'єктів. Швидкість роботи моделі є критичною для реального використання в системах реального часу.
Масштабованість	Оцінюється здатність моделі працювати ефективно при збільшенні об'єму даних або складності завдання. Масштабованість є важливим аспектом для систем, які можуть стикатися з великими потоками відеоданих або різними умовами зйомки.

Оскільки наша система має працювати в реальному часі і бути точною – найбільш підходящими метриками послужать:

- швидкість навчання та виявлення обєктів та подій;
- точність. В умовах динамічної відеозйомки є важливим точно визначати об'єкти, які впливають на ідентифікацію подій. Ошибки у визначенні можуть вилитися у негативний глядацький досвід;
- масштабування. Оскільки система має працювати за принципом: “Чим більше каналів відео-інпуту, тим точніше і краще” здатність до масштабування є вкрай важливою.

2.2.1 Аналіз ефективності використання нейронних мереж конвуляційного типу

Перераховані вище метрики використовувалися для аналізу ефективності нейронних мереж конвуляційного типу. Для дослідження використовувалися CNN архітектури (ResNet, VGG, MobileNet) [17].

Тестування проводилося на тестових наборах даних, результати якого представлені у таблиці 8. Слід зазначити, що показник швидкості представлений у таблиці – відповідає часу потрібного CNN для обробки одного кадру відео. Експерименти з масштабованістю проводилися на великому обсязі даних з збільшенням градієнтного розміру моделі.

Таблиця 2.5 – Результати аналізу ефективності використання нейронних мереж конволюційного типу

	Точність	Швидкість	Масштабованість
ResNet	92%	120 мс/кадр	Добре
VGG	88%	140 мс/кадр	Добре
MobieNet	85%	90 мс/кадр	Відмінно
Середнє значення	88%	117 мс/кадр	Добре

Масштабованість архітектур ResNet і VGG є досить продуктивною при збільшенні обчислювальних потужностей. MobileNet – є менш ресурсомістким, маючи невелику кількість вхідних параметрів, що відбивається на нижчій точності, маючи при цьому високу здатність до масштабування.

2.2.2 Аналіз ефективності використання дерев рішень

Для аналізу дерев рішень як об'єктів експерименту використовувалися дерева рішень з глибиною 3, 5 та 10 [18].

Тестування проводилося на тестових наборах даних, результати якого наведено в таблиці 2.6.

Таблиця 2.6 – Результати аналізу ефективності використання дерев рішень

	Точність	Швидкість	Масштабованість
ДР с глубиной 3	75%	5 мс/кадр	Погано
ДР с глубиной 5	80%	10 мс/кадр	Погано
ДР с глубиной 10	85%	20 мс/кадр	Погано
Середнє значення	80%	12 мс/кадр	Погано

Таким чином дерева рішень виявляються ефективними в роботі з кадрами в реальному часі, забезпечуючи швидкий час обробки, але менш масштабованими в порівнянні з деякими складнішими моделями. Вони можуть виявитися неефективними при обробці великих обсягів даних.

2.2.3 Аналіз ефективності використання SVM метод

Для аналізу методу SVM як об'єкти експерименту були використані SVM моделі з різними конфігураціями[19], такими як:

- SVM із лінійним ядром;
- SVM з поліноміальним ядром;
- SVM з RBF (Rasial Basis Function) ядром.

Тестування проводилося на тестових наборах даних, результати якого наведено в таблиці 2.7.

Таблиця 2.7 – Результати аналізу ефективності використання методу SVM

	Точність	Швидкість	Масштабованість
Лінійне ядро	80%	5 мс/кадр	Відмінно
Поліноміальне ядро	85%	15 мс/кадр	Добре
RBF ядро	90%	10 мс/кадр	Добре
Середнє значення	85%	10 мс/кадр	Добре

Таким чином, підбиваючи загальний підсумок, порівняємо середні значення, отримані в результаті експериментів ефективності різних методологій (таблиця 2.8).

Таблиця 2.8 – Порівняння середніх результатів аналізу ефективності методологій

	Точність	Швидкість	Масштабованість
CNN	88%	117 мс/кадр	Добре
Дерева рішень	80%	12 мс/кадр	Погано
SVM	85%	10 мс/кадр	Добре

З цього випливає, що найбільш ефективним методом для реалізації системи відеоаналізу буде метод SVM, оскільки його результати мають високі показники в метриках роботи в реальному часі. Альтернативно може бути застосоване рішення з CNN, у модулях, що вимагають більш високої точності.

3 ОБҐРУНТУВАННЯ ВИБОРУ ТЕХНОЛОГІЇ

3.1 Огляд існуючих інтелектуальних моделей для розпізнавання статичних та динамічних об’єктів

У ході проведення досліджень інтелектуальних моделей розпізнавання статистичних та динамічних об'єктів було виявлено певні труднощі, вирішення яких слугувало вектором ходу роботи. У таблиці далі представлені найбільш проблемні моменти, і потенційні рішення, які можуть їх нівелювати.

Таблиця 3.1 – Класифікація проблематики, стосовно компонентів системи

Проблема	Опис	Рішення
Вхідне відео	Якість відео; Ракурс зйомки; Висока динаміка.	Контроль за розміщенням камер та їх технічними характеристиками. Зміна фокусування детектору на менш динамічні об'єкти
Суддівство	Залежність зміни рахунку від рішень рефері.	Навчання окремої моделі розпізнавання жестів рефері. Відстеження безпосередньо зміни рахунку.
Датасети	Відсутність професійних датасетів з тематики волейболу. Недостатня кількість анотованих зображень у датасетах.	Створення власного датасета під завдання.

Вимоги до вхідного відео відіграють ключову роль успішності роботи системи. Було проаналізовано близько 10 різних відео з волейбольними матчами. Ракурс зйомки, у кожному з них є приблизно однаковим (рис 3.1).

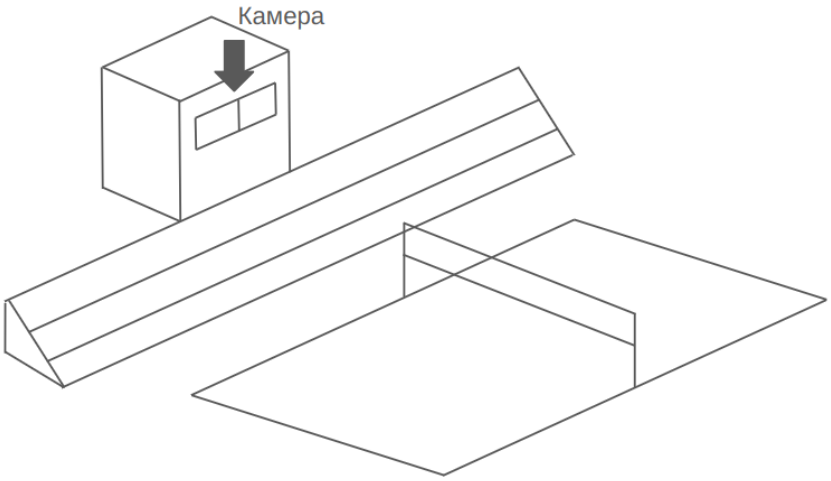


Рисунок 3.1. – Розташування камер відносно волейбольного поля у більшості матчів

Цей ракурс не дозволяє повноцінно оцінити положення м'яча щодо волейбольного поля (рисунок 3.2). Також, враховуючи високу динаміку волейболу, як спорту – іноді швидкість м'яча досягає пікових для захоплення зображення значень, що робить м'яч невидимим для камери (рисунок 3.3).

Рисунок 3.2. – Положення м'яча не дозволяє визначити, чи знаходиться м'яч у грі, або вийшов за межі поля

Рисунок 3.3 – М'яч після подачі стає невидимим для камери

Існує два підходи до вирішення цієї проблеми. Перший полягає у повному контролі над розташуванням камер. Таким чином можна було б проводити зйомку за виділеними зонами, на предмет будь-якої події. Однак цей підхід неможливий без контексту, оскільки одна і та сама подія, що потрапила на камеру однієї з зон, матиме різний результат. Так, наприклад, відскок від блоку в аут буде вважатися очком для команди, яка проводила атаку. Визначити в динаміці, хто торкнувся м'яча останнім, після зустрічі гравців, що блокують, є досить непростим завданням. Тому рішенням може бути аналіз дій гравців, які є менш динамічними суб'єктами у розкадруванні. Події на полі можуть бути більш точно трактовані за позицією гравця в той чи інший момент часу

Особливістю обраного виду спорту є те, що єдиним джерелом правди на полі виступає рефері. З цієї причини ідентифікація порушень системою може бути надмірною, оскільки несе за собою потенційну розсинхронізацію фактичних подій на полі із зареєстрованими суддею. На рисунку 3.4 заступ здійснено після свистка судді. Але система коментування буде детектувати цей кадр як заступ, не маючи відповідного контексту.

Рисунок 3.4 – Момент заступу на поле противника після свистка судді, яке потенційно може бути розглянуте системою розпізнавання, як порушення

На наступному рисунку представлений момент заступу на полі противника, без штрафного очка, що свідчить про те, що фіксація порушень не має особливого сенсу без синхронізації з діями рефері.

Рисунок 3.5 – Момент заступу на поле противника, який не був зарахований рефері у якості очка

Вирішенням цієї проблеми може бути аналіз команд і жестів суддів (рис 2.9), з допомогою створення окремої CNN, навченої під це завдання.

На момент написання цієї роботи у відкритому доступі є невелика кількість професійних датасетів. Більшість із наявних датасетів, ймовірно, були створені для навчальних цілей.

Так використання готового датасету [20], після навчання дає незадовільні результати (рисунок 3.6).

Рисунок 3.6 – Результат розпізнавання на базі готового датасету

Так, наприклад, використання датасету OpenImages [21] дає кращу детекцію об'єктів на полі (рисунок 3.7), проте виведення результатів може виявитися складним для трактування, до того ж у ході експериментів із OpenImages – розпізнавання м'яча в динаміці на полі виявилося непосильним завданням.

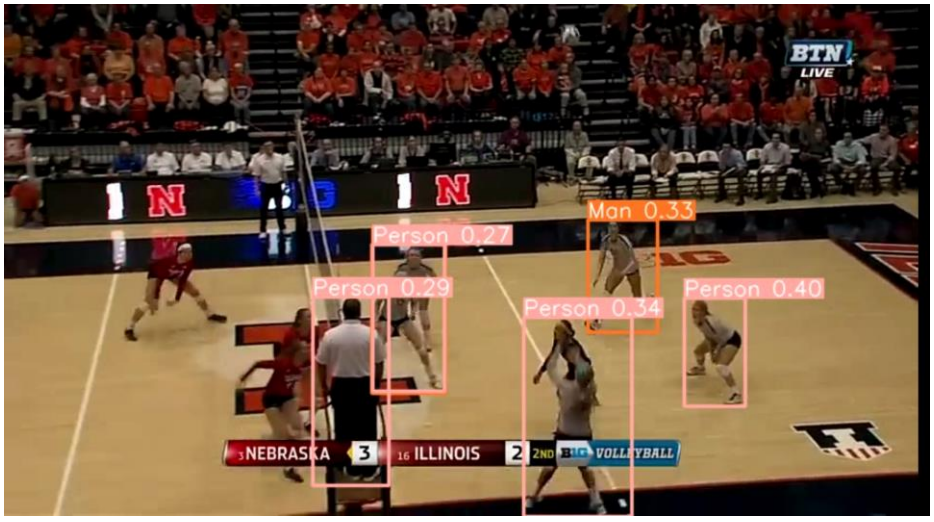


Рисунок 3.7 – Результат розпізнавання на базі датасету OpenImages

Вирішенням цієї проблеми є самостійне створення датасету під необхідні завдання. У ході роботи було створено датасет із ручною анотацією об'єктів у розмірі 270 зображень [22]. Результат його використання виявився більш ефективним, порівняно з попередніми експериментами (рисунок 3.8).

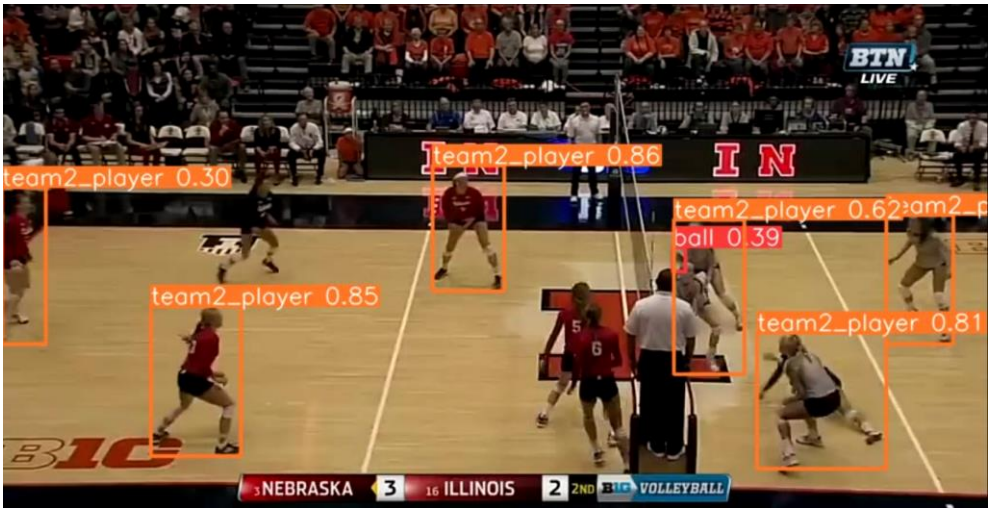


Рисунок 3.8 – Результат розпізнавання на базі кастомного датасету

Більш ефективними результати виявилися при обробці відео з того ж ракурсу і того ж масштабу, що зображення в датасеті (рисунок 3.9).

Рисунок 3.9 – Результат розпізнавання на базі кастомного датасету на тому ж полі, що і на зображеннях, анотованих у датасеті

З чого випливає, що підготовка кастомного датасету для нетривіальних завдань є, хоч і більш витратною за часом, але і більш ефективною, порівняно з використанням готових датасетів.

3.2 Обґрунтування вибору технології детектування та розпізнавання дій людини

Для вибору відповідних технологій, які б ефективно розпізнавати дії людини на полі в динаміці було проведено аналіз існуючих рішень, результати якого представлені в таблиці 3.2.

Порівнюючи платні сервіси Microsoft Azure Cognitive Services [27] та OpenAI GPT [31], можна сказати, що сервіс від Azure є більш відповідним рішенням для екосистеми Microsoft і є менш гнучким, оскільки має обмежені можливості настроювання. OpenAI у свою чергу є більш гнучким, проте він може бути дуже витратним для покадрової обробки зображень, а також має певні обмеження у політиці використання та адаптації під вузькоспеціалізовані напрямки.

Порівнюючи OpenCV [29] з CNN [30], можна зробити висновок, що OpenCV більше підходить для завдань загального призначення обробки зображень і відео. Щодо глибокого навчання OpenCV є більш обмеженим. CNN ж є технологією спрямованої більш високу точність результатів, вартість якої – необхідність у достатніх обчислювальних ресурсах.

Для виконання поставленої задачі найбільш підходящим інструментом буде CNN, оскільки в динамічних видах спорту фіксування об'єктів, що швидко рухаються, відіграє ключову роль.

Зокрема, для реалізації завдань розпізнавання об'єктів у реальному часі відмінно підходить CNN архітектура YOLO v8.

Таблиця 3.2 – Системи на основі штучного інтелекту, які базуються на поєднанні апаратних та програмних засобів для розпізнавання об’єктів.

	Amazon Recognition [25]	Google Cloud Vision [26]	Microsoft Azure Cognitive Services [27]	IBM Watson Visual Recognition [28]	OpenCV з TensorFlow [29]	CNN [30]	GPT (OpenAI API) [31]
Базові можливості	Ідентифікація обличчя. Порівняння облич. Атрібутика облич. Модерація контенту. Виявлення брендів. Розпізнавання тексту	Ідентифікація тексту (картинки, документи, пропис). Ідентифікація логотипів/етикеток. Ідентифікація об’єктів архітектури і їхнього геоположення. Властивості зображення. Розпізнавання об’єктів / веб об’єктів на рисунку.	Опрацювання NLP. Розпізнавання мови. Вилучення інсайтів з відео. Аналіз зображення. Ідентифікація облич. Класифікація зображень. Роспізнавання тексту.	Розпізнавання об’єктів та сцен. Класифікація зображень. Виявлення обличчя. Роспізнавання тексту. Аналіз колірної палітри. Модерація контенту. Сегментація зображення.	Розпізнавання облич. Виявлення об’єктів. Визначення руху. Розпізнавання та розуміння тексту. Вимірювання відстаней і розмірів. Фільтрація та підсилення зображень.	Згорткові шари. Пулінгові шари. Повні зв’язні шари. Функції активації. Регуляризація та уникнення перенавчання.	Обробка природньої мови. Генеративні моделі. Розмовні агенти. Кодування та програмування. Навчання моделей. Аналіз даних.
Необхідність додаткового апаратного забезпечення	-	-	-	-	+	+	-
Підтримка розпізнавання об’єктів в режимі реального часу	+	(Google Cloud Video Intelligence)	+	+	+	+	-
Легкість масштабування системи	+	+	+	+	Варьюється від складності поставленої задачі.	Варьюється від архітектури	Доволі легко
Підтримка мультимодальних даних	-	+	+	+	Частково	+	+
Можливість донавчання	-	-	+	+	Не передбачено навчання	+	-
Легкість інтеграції	Складно для ріл тайму	Доволі легко	Легко	Доволі легко	Доволі легко	Легше за середнє	Легко
Швидкість обробки запиту	> декількох секунд для відео	Варьюється. Від 100 мс до декількох секунд	Варьюється. В середньому 1-2 секунди	Варьюється від 1 до 5 секунд	Варьюється від мілісекунд до декількох секунд	Варьюється від мілісекунд до декількох секунд	Залежить від задач. Генерація може займати до 20 секунд

YOLOv8 [32] – це модель глибокого навчання, спеціалізована на задачах розпізнавання об'єктів. Її унікальність полягає в здатності виконувати цю задачу в один етап (end-to-end), що значно підвищує продуктивність та швидкість обробки зображень. YOLOv8 використовує конволюційні нейронні мережі (CNN) для аналізу вхідних зображень і передбачення меж об'єктів та їх класів.

Модель має одноступневий процес розпізнавання. Модель ділить зображення на сітку і прогнозує межі та класи об'єктів для кожної комірки сітки одночасно. Завдяки своїй архітектурі, YOLOv8 здатна працювати в режимі реального часу, обробляючи відео з високою частотою кадрів. Також поліпшені алгоритми та архітектурні вдосконалення забезпечують високу точність розпізнавання навіть для дрібних об'єктів на складному фоні.

В контексті задачи по розпізнаванню об’єктів під час волейбольного матчу, для миттєвої видачі аналітики YOLOv8 має свої переваги. Ця модель здатна обробляти відео з низькою затримкою, що забезпечує своєчасне виявлення подій. Її висока точність дозволяє розпізнавати не тільки гравців і м'яч, але й визначати позиції гравців, відстежувати траєкторію м'яча і аналізувати різні події на полі.

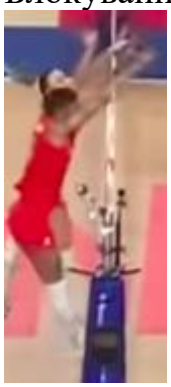
Порівняно з іншими методами розпізнавання об'єктів, такими як R-CNN, Fast R-CNN, Faster R-CNN та SSD (Single Shot MultiBox Detector), YOLOv8 є більш швидкою, оскільки всі розрахунки виконуються за один прохід через мережу, що значно підвищує швидкість обчислення, що є важливим при обробці в реальному часі. Вона не вимагає попереднього обчислення регіонів інтересу (RoI), що знижує складність і підвищує продуктивність. А також YOLOv8 добре працює з різноманітними типами даних і може бути налаштована для різних рівнів продуктивності та точності, що дозволяє адаптувати її для різних умов та апаратних ресурсів.

3.3 Аналіз подій на майданчику, що підлягають оцінці автоматизованою системою коментування

Провівши аналіз доступних у загальному доступі матчів з волейболу, можна зробити висновок, що гра відрізняється високою динамічністю, особливо

в розрізі професійних команд. Розіграші м'яча, як правило, не є затягнутими. Найчастіший сценарій – одна з команд заробляє ігрове очко протягом однієї-двох атак. У середньому це 3-5 атак обома командами за розіграш.

Таблиця 3.3 – Опис та класифікація ключових подій на майданчику для коментування системою.

Клас дії: Захисні дії		
Приклад події у класі	Блокування (blocking) 	Прийом м’яча (digging) 
Клас дії: Атакуючі дії		
Приклад події у класі	Передача м’яча (setting) 	Атака (spiking) 
Клас дії: Загальні дії		
Приклад події у класі	Падіння (falling) 	Відстеження рахунку 

3.4 Особливості датасетів спортивних подій

У процесі створення системи автоматичного коментування волейбольних матчів було проведено пошук готових датасетів для

використання їх у роботі. Більшість датасетів, представлених у загальному доступі, були створені для навчальних цілей, не мають достатньої кількості анотацій, а також класів, необхідних для системи.

Зважаючи на ці обставини, була спроба створення власного датасета, який у перспективі, міг би послужити опорною точкою для навчання моделі розпізнавання об'єктів на волейбольному полі.

Для цього було зібрано близько 300 скріншотів з кількох волейбольних матчів, які згодом були відфільтровані щодо великих планів окремого гравця, глядачів, порожнього поля тощо.

Зрештою, у датасет потрапили 270 зображень, які були вручну анотовані за допомогою платформи Roboflow [23]. Датасет був експериментальним, з метою визначити наскільки власноруч створений датасет покращить ефективність результатів навчання моделі.

Незважаючи на надзвичайно малий обсяг даних у датасеті – результат виявився досить задовільним, для розпізнавання гравців та м'яча у конкретних ракурсах (рисунок 3.9).

Рисунок 3.9 – Результат розпізнавання на базі кастомного датасету

Даний експеримент показав, що створення кастомного датасета під конкретне завдання, може бути гарною альтернативою іншим рішенням. Однак вимагає більше часу на створення та анотування, а також повинен

включати набагато більшу кількість зображень з різних ракурсів та залів для більш точної ідентифікації та класифікації.

Найбільш підходящим варіантом для цієї роботи став Deep Activity Recognition [24] датасет, який включає 4830 анотованих зображень, отриманих шляхом вилучення кадрів з 55 різних відео. Даний датасет включає 9 класів активності гравців і 8 класів активності команд, представлених на рисунку 3.10.

Group Activity Class	No. of Instances
Right set	644
Right spike	623
Right pass	801
Right winpoint	295
Left winpoint	367
Left pass	826
Left spike	642
Left set	633

Action Classes	No. of Instances
Waiting	3601
Setting	1332
Digging	2333
Falling	1241
Spiking	1216
Blocking	2458
Jumping	341
Moving	5121
Standing	38696

Рисунок 3.10 – Класи активності гравців та команд у датасеті

Далі у таблиці 3.4 наведені характеристики датасетів, які використовувалися для навчання та тренування моделі.

Таблиця 3.4 – Огляд використаних датасетів для навчання та тренування інтелектуальних моделей

	Кастомний датасет	Deep Activity Recognition
1	2	3
Розмір датасету	39.4 MB	672.3 MB
Кількість зображень	648 (після аугментації)	4830
Кількість категорій	3	9 по гравцям 8 по командам
Збалансованість датасету (розподіл кількості зображень по класах)	Train: 567 Valid: 54 Test: 27	Train: 2512 Valid: 1341 Test: 1337

Продовження таблиця 3.4

1	2	3
Збалансованість співвідношення зображень до лейблів	Збалансований	Незбалансований(img:label): Train: 2512:1960 Valid: 1341:1199 Test: 1337:1211
Співвідношення розподілу наборів даних (training set: validation set: test set)	88%:8%:4%	44%:28%:28%
Формат даних	.txt	.txt
Характеристики вхідних даних (роздільна здатність, тривалість…)	640x640	1920x1080 (8 imgs) 1280x729 (rest)
Ліцензійні умови використання	-	BSD 2-Clause license
Різноманітність умов реєстрації даних (шуми, освітлення, кут зйомки….)	Кут зйомки поза лінією довжини поля зверху (рівень шуму високий) Кут зйомки повторів поза лінією ширини поля зверху (тільки повтори, рівень шуму низький)	Кут зйомки поза лінією довжини поля зверху (рівень шуму високий)

4 РЕАЛІЗАЦІЯ ПОСТАВЛЕНОЇ ЗАДАЧІ

4.1 Модель розпізнавання динамічних об’єктів

У роботі для ідентифікації динамічних об'єктів було використано модель YOLO v8 [32]. Для цього завдання YOLO використовує конволюційні нейронні мережі. Основна ідея полягає в поділі зображення на сітку та передбачені межі об'єктів та їх класів у кожному осередку сітки. Далі представлене математичне подання алгоритму роботи даної моделі:

Нехай I - це вхідне зображення розміром $W \times H$. Зображення ділиться на сітку розміром $S_w \times S_h$, де S – кількість комірок в сітці. Кожна комірка в сітці відповідає за передбачення B обмежувальних рамок (bounding boxes) та відповідних їм значень впевненості (confidence scores). Таким чином, для кожної комірки ми маємо:

$$b_i = (x_i, y_i, w_i, h_i, c_i),$$

де x_i та y_i – кординати центру рамки, w_i і h_i – ширина та висота рамки, а c_i – значення впевненості для i-ї рамки. Також кожна комірка передбачає ймовірність приналежності до класів для об'єктів C класів:

$$p(class_i|object).$$

Вихідний тензор має розмір $S_w \times S_h \times (B \times 5 + C)$, де кожна комірка містить B передбачень граничних рамок (кожне з них має 5 значень: x, y, w, h, c) і C значень ймовірностей класів.

Обчислення значень втрат (loss function) відбувається у три етапи.

Обчислення втрат об'єктності:

$$Loss_{conf} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{obj} \left(C_i - \widehat{C_i}\right)^2 + \lambda noobj \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{noobj} \left(C_i - \widehat{C_i}\right)^2$$

Обчислення втрат класу:

$$Loss_{class} = \sum_{i=0}^{S^2} 1_{i=0}^{obj} \sum_{c \xi classes} (p(c) - \widehat{p}(c))^2$$

Разом загальна функція втрат виглядає наступним чином:

$$Loss = \lambda_{coord} Loss_{coord} + Loss_{conf} + Loss_{class}.$$

В архітектурі YOLOV8 можна виділити кілька основних компонентів:

- backbone. Основа архітектури YOLOV8 відповідає за вилучення карт (maps) об'єктів із вхідного зображення. Зазвичай вона складається із серії згорткових шарів, які фіксують ієрархічні функції на кількох рівнях абстракції. Загальні магістралі, що використовуються у моделях YOLO, включають варіанти CSPDarknet53, MobileNet та ResNet. YOLOv8 використовує оптимізовану версію цих магістралей, щоб збалансувати швидкість та точність;

- neck. Цей компонент призначений для покращення карт ознак, які були витягнуті попереднім компонентом, а також для підготовки їх перед передачею наступному. До нього входять додаткові шари, які поєднують і уточнюють функції різних масштабів. Тут часто використовується структура пірамідальної мережі функцій (FPN) або мережі агрегації шляхів (PAN), щоб гарантувати, що для виявлення використовують як високорівневі, так і низькорівневі функції;

- head. Глава виявлення YOLOv8 відповідає за прогнозування обмежуючих рамок, ймовірностей класів і балів об'єктності для кожного поля прив'язки. Голова обробляє карти ознак, що виводяться шиєю(Neck), і

генерує остаточні прогнози. Як правило, цей компонент включає згорткові шари, за якими слідує ряд шарів виявлення, які видають вихідні дані для різних масштабів;

- anchor boxes. YOLOv8 використовує попередньо визначені поля прив'язки (або точки прив'язки) у різних масштабах для прогнозування обмежуючих рамок, навколо об'єктів. Ці поля прив'язки розроблені таким чином, щоб відповідати формам та розмірам об'єктів у наборі навчальних даних. Модель вчиться регулювати ці поля прив'язки під час навчання, щоб відповідати виявленим об'єктам;

- loss функції. Функція втрат в архітектурі YOLOV8 об'єднує кілька компонентів для оптимізації моделі під час навчання. Основні компоненти функції втрат включають:

- втрати регресії обмежуючої рамки, що обмежує: вимірює різницю між прогнозованими рамками, що обмежують, і рамками основної істини;

- втрати об'єктності: оцінюється ступінь достовірності прогнозованих обмежуючих рамок, що містять об'єкти;

- втрати ймовірності класу: оцінюється точність прогнозованих міток класів для кожної обмежуючої рамки;

- post-processing. Після того, як модель генерує прогнози, використовуються етапи постобробки для уточнення результатів. Ці кроки включають:

- немаксимальне придушення (NMS): усуває надмірні рамки, що обмежують, зберігаючи тільки ті, які мають найвищі оцінки достовірності;

- порогове значення. Відфільтровує прогнози з низьким рівнем достовірності, щоб зменшити кількість хибних спрацьовувань.

- training and inference. YOLOV8 навчається на великомасштабних наборах анотованих даних з використанням контрольованого навчання. Під час навчання модель навчається виявляти та класифікувати об'єкти, мінімізуючи функцію втрат. Під час виведення YOLOv8 обробляє вхідні зображення в режимі реального часу та генерує обмежувальні рамки та

позначки класів для виявлених об'єктів. Така архітектура забезпечує швидке і точне виявлення, що підходить для різних програм. Більш детальне уявлення архітектури YOLO v8 зображено на рисунку 4.1.

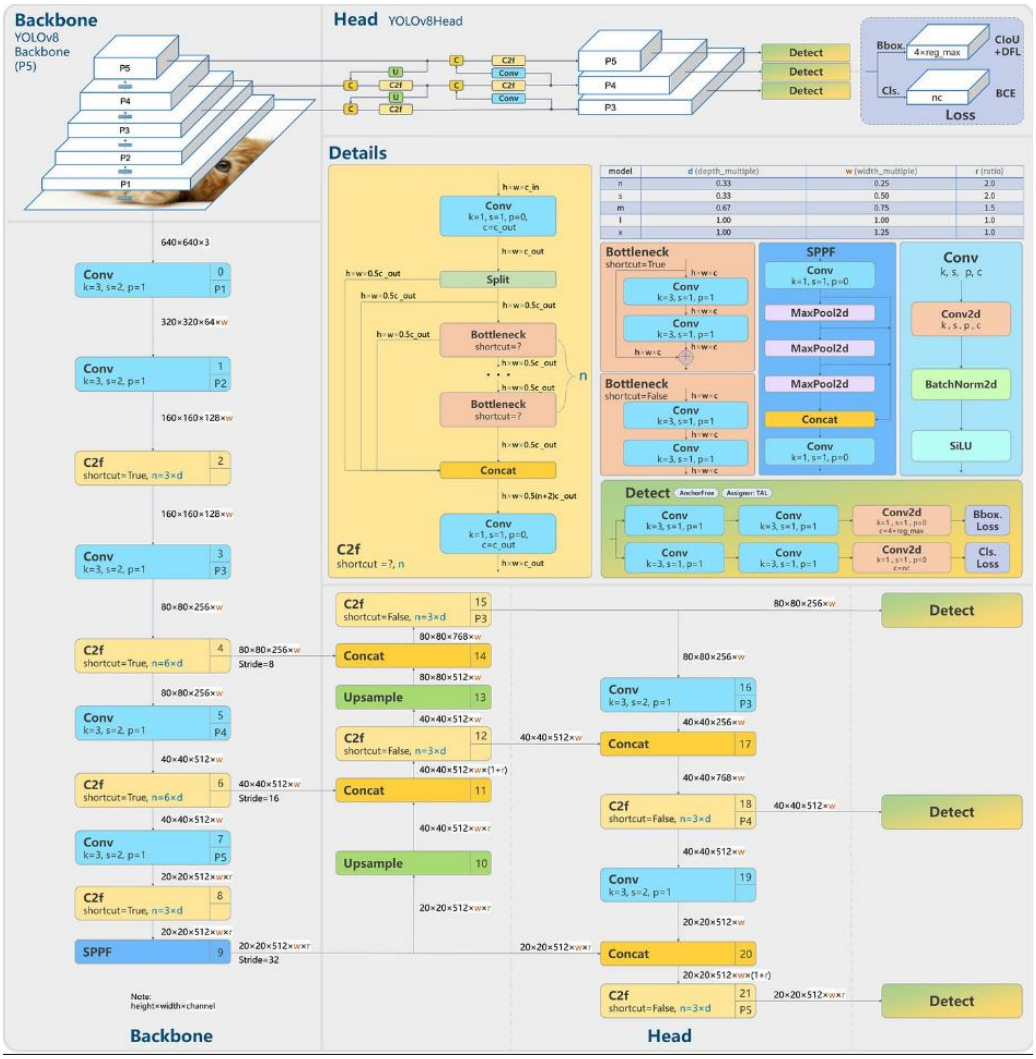
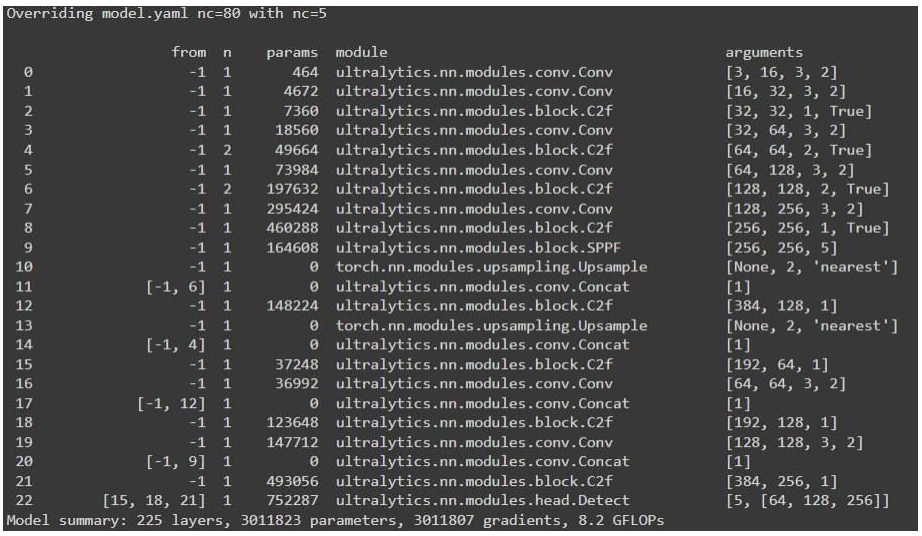


Рисунок 4.1 – Архітектура YOLOv8

Інтегруючи ці компоненти, YOLOv8 досягає високої продуктивності з точки зору швидкості та точності, що робить його підходящим вибором для завдань виявлення об'єктів у реальному часі, таких як виявлення ключових дій у волейбольних матчах.

Більш практичне уявлення даних компонентів, як шарів CNN, можна отримати в процесі роботи моделі (рисунок 4.2)



```
Overriding model.yaml nc=80 with nc=5

                   from  n    params  module                                       arguments
  0                  -1  1       464  ultralytics.nn.modules.conv.Conv             [3, 16, 3, 2]
  1                  -1  1      4672  ultralytics.nn.modules.conv.Conv             [16, 32, 3, 2]
  2                  -1  1      7360  ultralytics.nn.modules.block.C2f             [32, 32, 1, True]
  3                  -1  1     18560  ultralytics.nn.modules.conv.Conv             [32, 64, 3, 2]
  4                  -1  2     49664  ultralytics.nn.modules.block.C2f             [64, 64, 2, True]
  5                  -1  1     73984  ultralytics.nn.modules.conv.Conv             [64, 128, 3, 2]
  6                  -1  2    197632  ultralytics.nn.modules.block.C2f             [128, 128, 2, True]
  7                  -1  1    295424  ultralytics.nn.modules.conv.Conv             [128, 256, 3, 2]
  8                  -1  1    460288  ultralytics.nn.modules.block.C2f             [256, 256, 1, True]
  9                  -1  1    164608  ultralytics.nn.modules.block.SPPF            [256, 256, 5]
 10                  -1  1         0  torch.nn.modules.upsampling.Upsample         [None, 2, 'nearest']
 11            [-1, 6]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 12                  -1  1    148224  ultralytics.nn.modules.block.C2f             [384, 128, 1]
 13                  -1  1         0  torch.nn.modules.upsampling.Upsample         [None, 2, 'nearest']
 14            [-1, 4]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 15                  -1  1     37248  ultralytics.nn.modules.block.C2f             [192, 64, 1]
 16                  -1  1     36992  ultralytics.nn.modules.conv.Conv             [64, 64, 3, 2]
 17           [-1, 12]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 18                  -1  1    123648  ultralytics.nn.modules.block.C2f             [192, 128, 1]
 19                  -1  1    147712  ultralytics.nn.modules.conv.Conv             [128, 128, 3, 2]
 20            [-1, 9]  1         0  ultralytics.nn.modules.conv.Concat           [1]
 21                  -1  1    493056  ultralytics.nn.modules.block.C2f             [384, 256, 1]
 22       [15, 18, 21]  1    752287  ultralytics.nn.modules.head.Detect           [5, [64, 128, 256]]
Model summary: 225 layers, 3011823 parameters, 3011807 gradients, 8.2 GFLOPs
```

Рисунок 4.2 - Скріншот з процесу навчання моделі YOLOv8

4.2 Визначення ключових кадрів

Після обробки відео за допомогою моделі YOLOv8 ми маємо на виході анотовані кадри, які необхідно обробити для пошуку ключових подій. Для цього відео розбивається на сегменти, у кожному з яких виділяється ключовий момент (Додаток А). Надалі ці ключові кадри будуть використовуватись для озвучення коментатором.

У цьому підході використовується евристичний підхід припущенням, що базується на двох метриках. Найбільша кількість граничних рамок (bounding boxes) ідентифікованих на кадрі та найбільший середній показник впевненості (average confidence). Метод не є ідеальним з погляду кінцевого результату, проте націлений на згладжування прогалин у визначенні дій на полі в умовах роботи з відносно невеликим датасетом.

В якості альтернативи може бути використаний класифікатор YOLO, який теоретично здатний підвищити ефективність системи коментування.

4.3 Генерація текстових коментарів

Для генерації текстових коментарів у роботі була обрана модель GPT-4o [31]. Ключовою особливістю якої є вбудована підтримка

мультимодальності даних, це дозволяє передавати в модель не тільки промпт згенерований методом розпізнавання об'єктів, але також зображення. В цілому, всі моделі GPT 4-ї версії підтримують мультимодальність, проте GPT-4o, вона ж Turbo, є більш сбалансованою версією, що поєднує в собі більш високу швидкість та помірні витрати за користування.

Архітектура моделі GPT-4o, ґрунтується на архітектурі трансформерів, яка властива більшості моделей обробки природного мовлення. Основними компонентами її архітектури є:

- трансформери (Transformer). Включають енкодери і декодери;
- подання вхідних даних (Input Representation). Подання із послідовності токенів. У випадку тексту токеном може бути слово або частина слова. У разі зображення – кодування послідовності токенів є візуальними ознаками;
- ембеддінги (Embeddings). Служать для збереження семантики шляхом представлення токенів у вигляді багатовимірного вектора;
- позиційні енкодинги (Positional encodings). Відповідають безпосередньо за позиціонування токенів у послідовності;
- шари трансформера (Transformer Layers) – служать підвищення точності результатів, захоплення складних залежностей, і оптимізації процесу навчання;
- вихідний шар (Output Layer) – перетворює кінцеві векторні уявлення токенів на ймовірність наступних токенів за допомогою Softmax функції.

Вхідний текст обробляється шляхом перетворення слів у токени, які далі перетворюються в ембеддінги. Для кожного токена x_i, ембеддінг представляється вектором $E(x_i)$.Оскільки модель трансформера не має вбудованої інформації про порядок токенів, додаються позиційні енкодінги $P(i)$, які представляють позицію токена в послідовності. Загальний векторний представник для токена x_i є сумою його ембеддінга і позиційного енкодінга:

$$z_i = E(x_i) + P(i)$$

Ключовим компонентом трансформера є механізм самоспрямування (Self-Attention), який використовується для обчислення залежностей між токенами в послідовності.

Для кожного вектору z_i обчислюються три вектори: запит Q, ключ K та значення V:

$$Q_i = W_Q z_i,\ K_i = W_K z_i,\ V_i = W_V z_i,$$

де W_Q , W_K , W_V – матриці ваг.

Після обчислення самоспрямування вираховується міра уваги. Вона обчислюється як скалярний добуток запиту Q_i, і ключа K_j, нормований на розмірність векторів:

$$Attention(Q_i, K_j) = \frac{Q_i K_j^T}{\sqrt{d_k}},$$

де d_k – розмірність векторів ключів.

Для того щоб отримати зважені значення – обчислюється добуток нормованих оцінок уваги і відповідних векторів значень:

$$Attention(Q, K, V) = softmax(\frac{Q_i K_j^T}{\sqrt{d_k}})V$$

.

Для захоплення залежностей на великих відстанях використовується механізм багатоглового самоспрямування (Multi-Head Self-Attention), використовуючи кілька наборів запитів, ключів і значень для забезпечення більшої потужності моделі:

$$MultiHead(Q, K, V) = Concat(head_1, \ldots, head_h)W_O,$$

де кожна голова обчислюється як:

$$head_i = Attention(QW_Q^i, KW_K^i, VW_V^i),$$

а W_O – матриця ваг, для об’єднання голів.

Також кожен з шарів трансформера містить повнозв’язні шари, які застосовуються до кожного вектору окремо:

$$FFN(x) = max(0, xW_1 + b_1)W_2 + B_2,$$

де W_1, W_2 і b_1, b_2 – параметри повнозв’язних шарів.

GPT-4o складається з кількох шарів трансформера, кожен з яких включає механізм багатоголового самоспрямування і повнозв'язні шари. Кожен шар використовує обхідні з'єднання (residual connections) і нормалізацію шарів (layer normalization) для стабілізації навчання.

$$LayerNorm(x + MultiHead(Q,K,V)) \rightarrow LayerNorm(x + FFN(x))$$

Остаточний векторний представник кожного токена перетворюється у ймовірності наступних токенів за допомогою софтмакс-функції:

$$P(x_{i+1}|x_1, \ldots, x_i) = softmax(z_i W_e^T),$$

де W_e – матриця ембеддінгів.

Таким чином архітектура GPT-4o базується на трансформері і включає механізми багатоголового самоспрямування та повнозв'язних шарів. Основні оптимізації включають ефективне управління пам'яттю і паралельну обробку даних, що забезпечує високу продуктивність і знижує вартість використання моделі. Мультимодальність підтримується шляхом інтеграції текстових і візуальних даних, що розширює можливості використання моделі в різних доменах.

4.4 Модель генерації аудіо

Для генерації аудіоряду була використана модель XTTS-2 [33]. Це передова модель Text To Speech, яка спеціалізується на багатомовній генерації мовлення. Вона підтримує 17 мов. Нажаль, українська мова поки не входить до цього переліку.

З основних характеристик можна ще виділити клонування голосі, яке на базі короткого аудіозразку дозволяє створювати нові мовні виходи, що максимально точно імітують оригінальний голос. Модель здатна передавати емоції та стиль мовлення і має підвищену якість звуку.

Архітектура цієї моделі складається з наступних компонентів:

- текстовий аналізатор (Text Analysis). Відповідає за попередню обробку тексту, включаючи токенізацію, визначення частин мови та створення лінгвістичного представлення тексту. У математичному представленні токенізація виглядає як перетворення тексту у послідовність токенів. По аналогії з GPT моделлю:

$$Tokenization(T) = \{t_1, t_2, \ldots, t_n\}$$

- лінгвістична передача (Linguistic Processing). Виконує фонетичний аналіз тексту, створюючи фонетичні представлення слів і речень. Це включає визначення правильних вимов, інтонацій, ритму та наголосів;

- акустична модель (Acoustic Model). Перетворює фонетичні представлення в акустичні характеристики, що необхідні для синтезу мови. Використовує глибокі нейронні мережі для моделювання людської мови. Акустична модель є ключовим компонентом. Вона перетворює текстове представлення у мел-спектрограму. Це включає в себе використання трансформерів для обробки тексту і генерації акустичних характеристик. Як і у моделі GPT у трансформерах XTTS-2 використовується механізм самоспрямування(Self-Attention), в якому запит Q, ключ K та значення V обчислюються наступним чином:

$$Q = XW_Q , K = KW_K , V = XW_V,$$

де X – вхідна матриця токенів, а W_Q , W_K , W_Q – вагові матриці. За аналогічною функцією обчислюються і ваги самоспрямування:

$$Attention(Q, K , V) = softmax(\frac{Q_i K_j^T}{\sqrt{d_k}})V .$$

Далі генерується мел-спектрограмма, яка є частотним представленням звуку, підлаштовану під людський слух. Вона обчислюється в два етапи. Перетворення Час-Фреквенція(STFT):

$$S(t, f) = |STFT(x(t))|^2,$$

де $STFT$ – короткотермінове перетворення Фур'є, яке обчислює спектр для кожного короткого інтервалу часу t. І перетворення безпосередньо на Мел-Шкалу:

$$M(t, m) = \sum_f H(m, f)S(t, f),$$

де $H(m, f)$ – фільтрбанк мел-шкали, який конвертує частоти у мел-шкалу.

- вокодер (Vocoding). Перетворює акустичні характеристики, створені акустичною моделлю, в синтезовану мову. З технічного боку, це перетворення мел-спектрограми у аудіосигнал. XTTS-2 використовує сучасні вокодери для забезпечення високої якості звуку, такі як WaveNet. Це нейронний вокодер,який моделює умовну ймовірність аудіосигналу на основі його попередніх значень:

$$p(x_t|x_{t-1}, x_{t-2}, \dots) = softmax(f(x_{t-1}, x_{t-2}, \dots)),$$

де f – нейронна мережа, яка обчислює умовну ймовірність наступного значення сигналу.

Останнім етапом роботи системи автоматичного генерування голосових коментарів є заміна оригінального аудіоряду згенерованим за допомогою фреймворку ffmpeg [34]. Кінцевий результат зберігається у mp4 форматі, та являє собою повне відео з синхронізованими коментарями(Додаток Б).

4.5 Узагальнена модель системи

Усі компоненти системи можна умовно зобразити у вигляді узагальненої системи автоматичного коментування спортивних подій (рис 4.3).

Запропонована концептуальна модель системи автоматизованого коментування спортивних змагань включає п’ять взаємопов’язаних функціональних модулів:

- модуль визначення дій гравця на полі. Задачею цього модулю є попереднє визначення дій спортсменів під час матчу, для подальшої передачі анотованого зображення в наступний модуль. Він зроблений на базі YOLOv8, навченої заздалегідь на підготовленому датасеті;
- модуль визначення ключових кадрів у відео. Цей модуль відповідає за вибір ключового кадру з певної послідовності, емпіричним методом. Критеріями кращого кадру є найбільша кількість боксів з найвищим середнім коефіцієнтом впевненості. Ключовий кадр передається далі;
- модуль генерації тексту за промптом. Цей модуль приймає кадр з анотацією, та робить запит на OpenAI API за для генерації тексту на базі картинки та промпту. В якості моделі обрана GPT-4o, оскільки вона є мультимодальною, та швидкою. Отриманий текст передається на обробку далі;
- модуль конвертації текстового опису в голосовий коментар. Задача яка перед ним стоїть полягає в генерації аудіо на основі отриманого тексту.

Для цього використовується модель TXXS - 2. Основними перевагами якої є відкритий програмний код та потужний функціонал;

- модуль синхронізації згенерованого аудіо та відео. Він відповідальний за синхронізацію окремих аудіозаписів відповідно до належного місця у відео. Він підготовлює кінечний файл відео з коментарями.

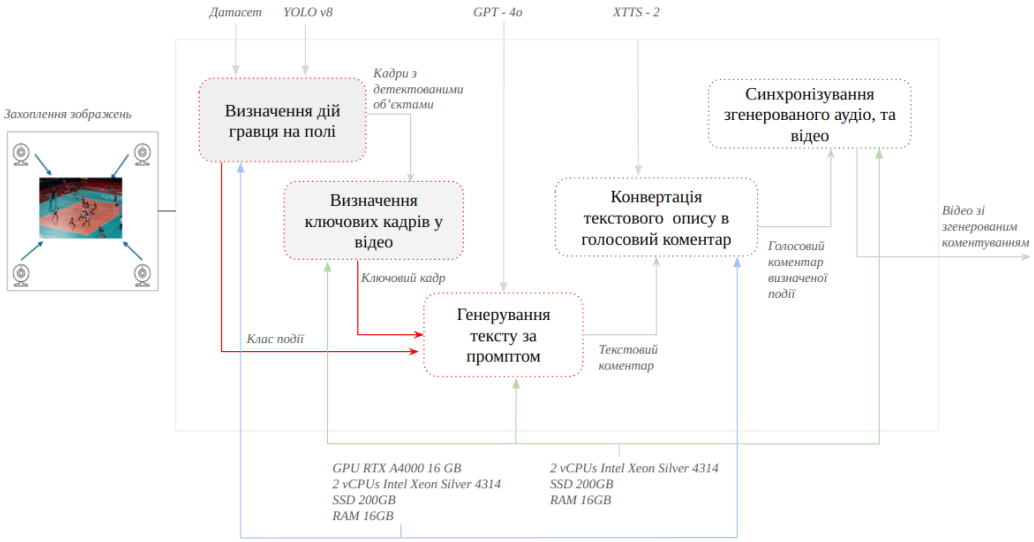


Рисунок 4.3 – Концептуальна модель системи автоматизованого коментування спортивних змагань

Дана система представлена у вигляді мінімально працездатного продукту (MVP) і має певні недоліки, які можуть бути виправлені в наступних версіях:

- тренування моделі YOLO. Оскільки тренування було обмеженим - модель YOLO може не захоплювати всі ключові моменти матчу. Більш посилена тренування на більшому і всесторонньому датасеті може підвищити точність вихідного результату;

- класифікація подій. У цій версії YOLO захоплює лише певну частину дій гравця, але не охоплює весь контекст матчу. Використання YOLO в якості класифікатора подій або комбінації подібних моделей для окремих піддоменів може виправити дане обмеження;

- вибір ключового кадру. Евристичний підхід у виборі ключового кадру є не самим надійним. Реалізація більш софістичного підходу потенційно здатна покращити цей процес;

- повторючісь коментарі. Модель для коментування може давати повторювані результати, оскільки не містить у собі послідовності попередніх коментарів матчу. Включення минулих ключових подій і розширення обмежень на коментарі дозволяють створити безперервний потік тексту, який буде виглядати як повноцінний текст коментатора;

- емоційна якість моделі TTS. Іноді модель TTS генерує плоску, беземоційну річ. Потенційно, апробація альтернативних спікерів з більш експресивними характеристиками може вирішити цю проблему;

- відставання коментарів від подій. Підхід із сегментуванням, заснований на пошук ключових кадрів у визначеній групі кадрів, іноді ключова подія може бути децентрованою, що викликає затримки. Необхідно мінімізувати розмір сегмента, або переглянути цей підхід визначення ключових подій у таймлайні.

4.5 Проведення експерементальних досліджень. Аналіз результатів

4.5.1 Експеримент 1. Дослідження характеристик та можливостей нейромережевого детектора та класифікатора подій у відеопослідовності

У якості детектора та класифікатора використовується нейромережева модель YOLOv8. Ціллю даного експерименту виступає відстеження змін показників при використанні різних оптимізаторів, запропонованих інтерфейсом YOLOv8. Вони спрямовані покращити точність, та швидкість навчання. Результати експерименту представлені у таблиці 4.1

Для зручності ці дані представлені у графічному вигляді по динаміці зміни метрик: Recall (рис. 4.4), Bbox loss (рисунок 4.5) та часу на навчання(рисунок 4.6).

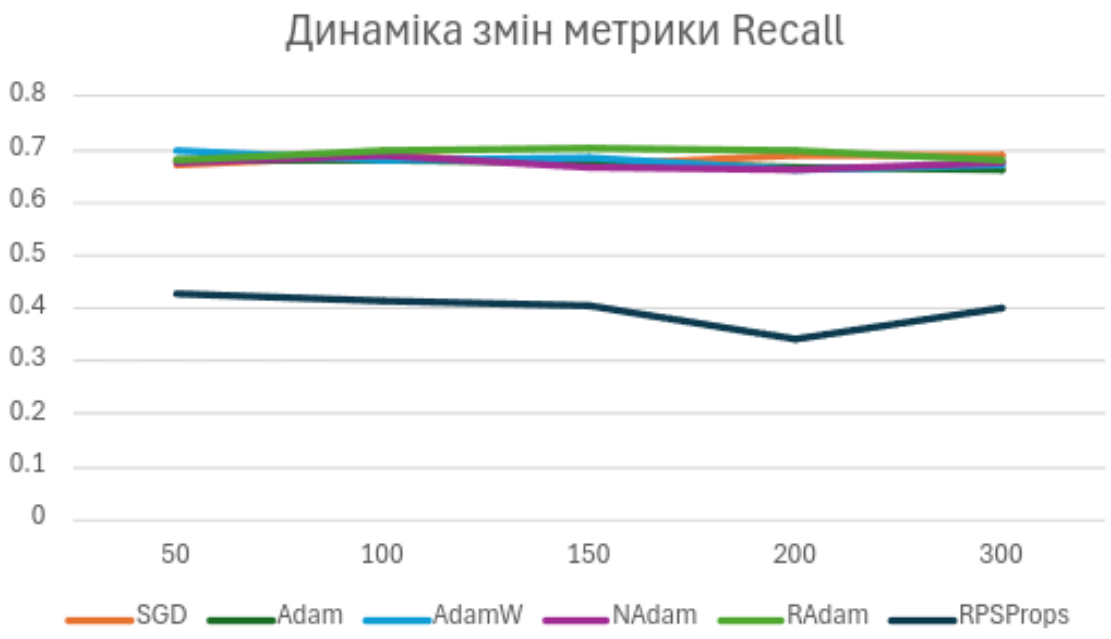


Рисунок 4.4 – Динаміка змін показників Recall при навчанні моделі з пропорціями датасету 44:28:28, в залежності від кількості епох, розподілена за оптимізаторами

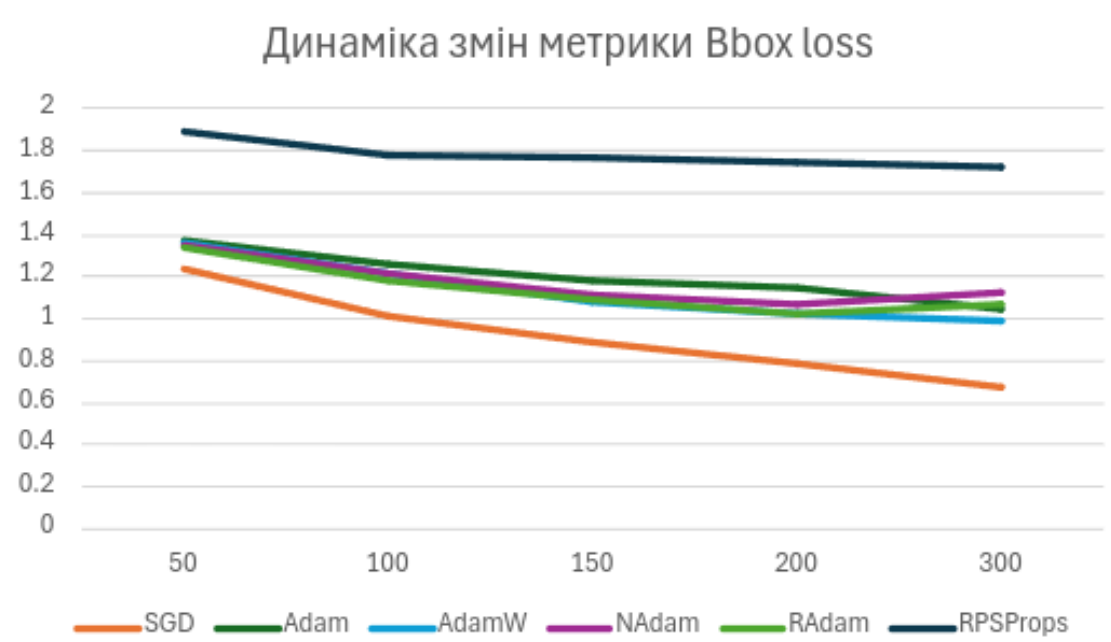


Рисунок 4.5 - Динаміка змін показників Bbox loss при навчанні моделі з пропорціями датасету 44:28:28, в залежності від кількості епох розподілена за оптимізаторами

Таблиця 4.1 – Вплив оптимізатора на точність детектування об’єктів в кадрі (Recal), втрати обмежувальної рамки, час навчання у залежності від кількості епох для класифікації подій на основі моделі YOLOv8

Оптимізатор	Кількість епох тренування	Recall	Bounding box loss	Час навчання (год)
1	2	3	4	5
Начальная: тренувальна : валідаційна = 44:28:28				
SGD	50	0.672	1.239	0.385
	100	0.688	1.016	0.769
	150	0.671	0.8891	1.129
	200	0.687	0.79	1.462
	300	0.687	0.6685	2.197
Adam	50	0.68	1.374	0.401
	100	0.678	1.261	0.781
	150	0.675	1.179	1.168
	200	0.665	1.143	1.540
	300	0.664	1.049	2.298
AdamW	50	0.699	1.367	0.364
	100	0.68	1.21	0.742
	150	0.686	1.079	1.131
	200	0.664	1.019	1.500
	300	0.673	0.9921	2.037
NAdam	50	0.675	1.356	0.397
	100	0.688	1.213	0.766
	150	0.668	1.116	1.179
	200	0.662	1.067	1.492
	300	0.674	1.126	2.013
RAdam	50	0.682	1.337	0.382
	100	0.698	1.178	0.774
	150	0.702	1.087	1.173
	200	0.699	1.023	1.475
	300	0.68	1.067	2.218 (Early stopping)

Продовження таблиця 4.1

1	2	3	4	5
RMSProp	50	0.429	1.896	0.380
	100	0.413	1.784	0.797
	150	0.403	1.763	1.132
	200	0.342	1.747	1.482
	300	0.401	1.725	2.199
Навчальна: тренувальна : валідаційна = 70:15:15				
SGD	50	0.759	1.242	0.444
	100	0.742	1.056	0.903
	150	0.755	0.9431	1.366
	200	0.731	0.8544	1.789 (early stop)
	300(298)	0.731	0.8544	1.789 (early stop)
Adam	50	0.723	1.38	0.472
	100	0.748	1.274	0.922
	150	0.715	1.217	1.371
	200	0.776	1.17	1.823
	300	0.735	1.107	2.685
AdamW	50	0.737	1.347	0.466
	100	0.748	1.204	0.917
	150	0.735	1.111	1.362
	200	0.732	1.029	1.802
	300(252)	0.732	1.062	2.305 (early stop)
NAdam	50	0.709	1.345	0.473
	100	0.744	1.223	0.925
	150	0.784	1.16	1.340
	200	0.772	1.109	1.812
	300	0.771	1.06	2.665
RAdam	50	0.734	1.331	0.482
	100	0.777	1.212	0.908
	150	0.766	1.142	1.342

Продовження таблиця 4.1

1	2	3	4	5
	200	0.772	1.109	1.820
	300	0.749	1.042	2.719
RMSProp	50	0.505	1.85	0.477
	100	0.478	1.748	0.929
	150	0.365	1.725	1.359
	200	0.478	1.72	1.807
	300(179)	0.347	1.954	1.598 (early stop)

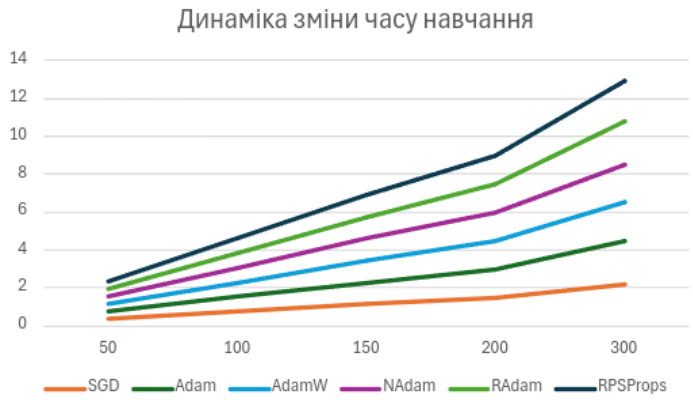


Рисунок 4.6 – Динаміка зміни часу навчання моделі з пропорціями датасету 44:28:28, в залежності від кількості епох розподілена за оптимізаторами

Таким чином оптимізатор SGD [35] є найбільш доцільним у використанні у контексті навчання моделі з пропорціями тренувальні:валідні:тестові 44:28:28. Він має позитивну динаміку підвищення показнику найбільшої точності об’єктів, з кожною епохою втрати обмежувальних рамок прагнуть до нулю, а час виконання є найбільш швидким навіть при великій кількості епох. Розглянемо динаміку зміни показників для більш розповсюдженого сету пропорцій 70:15:15 (рис. 4.7 - 4.9).

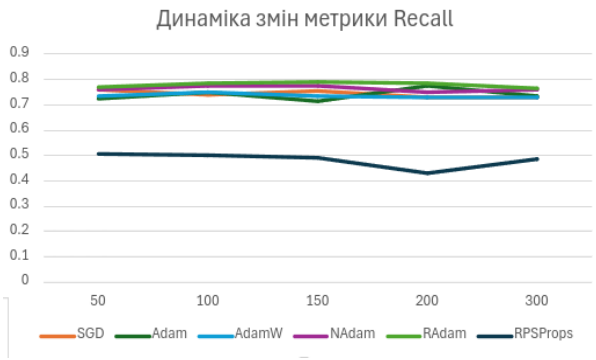


Рисунок 4.7 – Динаміка змін показників Recall при навчанні моделі з пропорціями датасету 70:15:15, в залежності від кількості епох розподілена за оптимізаторами

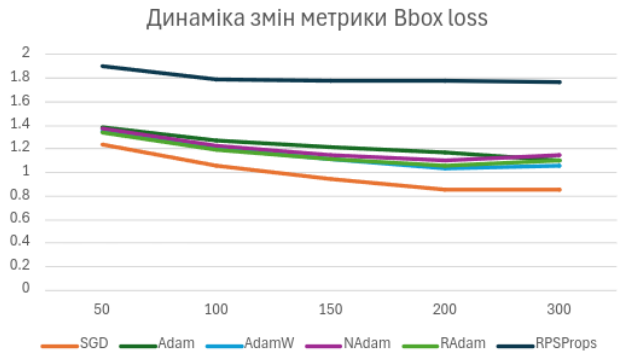


Рисунок 4.8 – Динаміка змін показників Bbox loss при навчанні моделі з пропорціями датасету 70:15:15, в залежності від кількості епох розподілена за оптимізаторами

Рисунок 4.9 - Динаміка зміни часу навчання моделі з пропорціями датасету 70:15:15, в залежності від кількості епох розподілена за оптимізаторами

Спираючись на результати експериментів, робимо висновок, що найкращу точність детектування динамічних об’єктів в кадрі вдається отримати при використанні оптимізатора SGD та розподілі датасету у співвідношенні 44:28:28 для начальної: тренувальної : валідаційної вибірок відповідно при 300 епохах навчання. Висновок зроблено на основі найнижчого значення Bounding box loss, який становить 0.6685 і є майже втричі кращим показником, аніж точність детектування при використанні оптимізатора RMSProp.

Найвищу точність класифікації детектованого об’єкту вдається досягти при розподілі датасету у співвідношенні 70:15:15 для начальної: тренувальної : валідаційної вибірок відповідно при 100 епохах навчання для оптимізатора RAdam. Показник Recall становить 0.777.

Графіки 4.7–4.9 підтверджують доцільність використання оптимізатора SGD, як найбільш ефективного з наявних для цього датасету. Що відрізняє їх – це рівень значень. Модель з пропорціями датасету 70:15:15, навіть у експериментах з невеликою кількістю епох, має більш високі показники стосовно кожної з метрик. Що означає що розподілений таким чином датасет здатен більш точно визначати об’єкти, але також має більш великі втрати обмежувальної рамки, та навчається більш довго.

4.5.2 Експеримент 2. Вплив мультимодальності на точність та повноту згенерованого коментаря

Метою даного дослідження є підтвердження теорії, щодо необхідності використовувати підхід з застосуванням мультимодальності при генеруванні коментарів. Для цього, в якості інпуту у запиті до моделі GPT-4o були використані: лише текст, лише зображення, текст + зображення з анотацією. Результати даного експеримента представлені на таблиці 4.2.

Таблиця 4.2 – Вплив мультимодальності на повноту і точність згенерованого коментаря

	Максимальна довжина вихідного коментаря	Вхідні дані	Згенерований коментар
1	2	3	4
Image (without annotation)	6		Unbelievable block! Turkey fights back!
	11		Incredible block from Turkey! What an electrifying rally moment!
	16		What an intense moment! Brazil is setting up for a powerful spike as Turkey fiercely defends!
	21		What an electrifying rally! TUR sets up for a powerful spike, but BRA's blocking might make all the difference!
Text	6	1 blocking, 1 spiking	Epic showdown: Block vs. Spike!
	11	1 blocking, 1 spiking	What a moment! Incredible spike, perfectly blocked! Unbelievable defense!
	16	1 blocking, 1 spiking	A thrilling showdown! One player blocks, while another goes for a powerful spike! Unbelievable action!
	21	1 blocking, 1 spiking	An electrifying moment! One player goes for an incredible block while the opposing side attempts a powerful spike. Intense showdown!
Image + Text annotation	6		High-flying spike! The tension rises!

Продовження таблиця 4.2

1	2	3	4
	11		Tense moment! Brazil spikes hard; Turkey fiercely blocks at the net!
	16		The Brazilian spiker soars high, but Turkey's blocker meets her! Intense volleyball action!
	21		Brazil spikes with power, but Turkey's block rises strong! What an intense rally in Hong Kong!

Відповідно до отриманих результатів можна дати висновок, що завдяки можливостям GPT-4o – кожен з підходів, відображених в експериментах, міг би бути використаним для генерування аудіо коментарів. Однак, наглядно видно, що в випадку з текстом – моделі приходиться багато “дофантазовувати”, щоб видати повноцінний коментар на основі анотацій. Що до використання зображення без анотації - цей варіант також є робочим, однак інколи модель плутається у діях гравців. Час від часу ключова подія, яка відбувається на полі, описується не корректно. Помилки виникають у створення відповідності дії, до гравця або команди, яка цю дію виконала. Також ідентефікація самих дій є не чіткою і інколи видає невірні результати.

Експерименти з анотованим зображенням і текстом завжди завершувалися точним результатом (за субєктивною оцінкою людини експерта). Щодо повноти згенерованого коментаря - виходячи з експериментів, можна сказати, що мультимодальність в даному випадку не грає ролі. Усі коментарі, незважаючи на вхідні дані були повними і розкритими. Однак у прикладі з текстом, є вірогідність що коментарі будуть час від часу повторюватись, оскільки генеруються на базі одного-двох слів.

Проведений аналіз результатів таблиці 4.2 показує переваги та недоліки кожного із трьох підходів:

- використання лише тексту на вході моделі генерування текстового повідомлення GPT-4o коментує лише загальні коментарі, не деталізуючи і не привязуючись до подій на майданчику. Це призводить до дублювання коментарів;

- використання лише зображень без анотування на вході моделі генерування текстового повідомлення GPT-4o призводить до некоректного визначення подій та гравців/команд на майданчику та подальшого формування хибного коментаря;

- використання мультимодальних даних на вході моделі генерування текстового повідомлення GPT-4o є обгрунтованим, бо дозволяє генерувати повні, точні коментарі, які не дублюються.

4.5.3 Експеримент 3. Вплив вибору кількості та частоти ключових кадрів на синхронізацію згенерованого коментаря та події в часі

Оскільки кадри поділяються на батчі, які в подальшому використовуються для знаходження ключового кадру у цій послідовності. Було вирішено визначити, як саме розмір батчу впливає на результуюче відео, а саме, чи зменшує розсинхронізацію між аудіо доріжкою и фактичним відео. З кожним експериментом розмір батчу подвоювався. Результати дослідження представлені у таблиці 4.3.

Під батчем, в цьому контексті, ми розуміємо позначення групи кадрів, які обробляються разом для знаходження серед них ключового.

Оскільки в експерименті ми спираємося саме на затримку, а помилки відповідності виступають лише додатковим критерієм, кращий за показниками вийшов результат з розміром батчу у 192 кадри. При такому батчі, затримок в коментуванні подій не відмічається. Оскільки ще три варіанти розміру батча показали такий саме результат, для порівняння був

використаний другорядний критерій, щодо помилок відповідності до події, який складає лише 10% від загальної кількості коментарів. Такі результати досягаються завдяки більш рідкому коментуванню, при великих розмірах батчу кадрів.

Таблиця 4.3 – Вплив вибору кількості та частоти ключових кадрів на синхронізацію згенерованого коментаря та події в часі

Розмір батчу	Щільність коментування	Помилки відповідності до події, %	Затримки коментування від подій, %	Кількість коментарів (тривалість відео 2хв)
3	Щільно	9%	4%	22
6	Щільно	9%	4%	22
12	Середнє	10%	5%	20
24	Середнє	13%	0%	20
48	Середнє	5%	10%	20
96	Рідко	13%	0%	15
192	Дуже рідко	10%	0%	10

Експерименти проводилися на одному відео з передачею різного розміру батчу, для виявлення залежностей. Помилки відповідності подій відносяться більше до моделі розпізнавання. Кожного разу зображення на рисунку 4.10 було трактовано, як подія “блокування”. До колонки стосовно помилок відповідності відносяться саме такі помилки.

Рисунок 4.10 – Момент, розпізнаний системою, як блок зі сторони команди в білому

Кількість затримок не виглядає критичною. Не можна адекватно відстежити закономірність у її змінах, оскільки генерування коментаря додає елемент ймовірності. Довжина згенерованого промпту може впливати на те, що деякі коментарі будуть пропущені, з метою органічного накладання одного аудіо на інше.

Виходячи з отриманих показників, можна заключити, що розмір батчу має бути або достатньо великим(192), або зовсім малим(3 або 6), для отримання результату з найменшою кількістю помилок(9-10%) та затримок(0-4%). Вибір буде залежати від потреб. Малий розмір батчу підходить під задачі зі щільним коментуванням, в той час як великий батч гарантує помірний темп коментування, з мінімальною кількістю помилок.

ВИСНОВКИ

Дослідження, які проводились в рамках цієї роботи, підтверджують актуальність проблеми автоматизації коментування спортивних подій. Існуючі рішення, у своїй більшості, не пропонують глядачам готової системи автоматичного генерування голосових повідомлень у спорті, яка могла б повноцінно замінити коментатора.

На підставі вивчених підходів, які застосовуються у існуючих рішеннях, була сформульована поверхнева модель альтернативної системи генерації коментарів для такого виду спорту, як волейбол.

Згідно то сформульованої моделі було проаналізовано та обрано технології для реалізації кожного окремого модуля в моделі системи та сформовано задачі, які мала вирішувати ця система.

Розробка власного застосунку мала на меті апробацію теоритичної складової, отриманої в результаті попередніх досліджень. Побудована система пропонує вирішення наступних задач: розпізнавання гравців на волейбольному полі, вибір ключового кадру з послідовності, генерування тексту коментаря, на основі отриманих даних, створення аудіо зі згенерованого тексту, синхронізація аудіо з оригінальним відео.

Результатом роботи стало впровадження мінімально життєздатної версії застосунку, здатного у перспективі повноцінно замінити коментатора волейбольних змагань.

Дослідження, які проводились на готовій системі визначили нові вектори, вздовж яких потрібно рухатись, для покращення вихідних результатів.

Так, продовжуючи роботу в даному напрямку можна збільшити рівень точності розпізнавання та класифікації об’єктів на полі, розширити кількість моделей для більш комплексного аналізу кадрів, додати класифікатор жестів

рефері, підключити аналіз аудіо, додати користувацький інтерфейс і можливість обирати мову та стилістику коментування, тощо…

Створений фундамент системи та його подальший розвиток може відкрити перед комерційними організаціями нові горизонти інтеграції бізнесу у спортивній індустрії, а перед вболівальниками – нові шляхи здобуття глядацького досвіду.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Horvat T., Job J. The use of machine learning in sport outcome prediction: A review //Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery. – 2020. – Т. 10. – №. 5. – С. e1380.

2. Merullo, Jack, et al. "Investigating sports commentator bias within a large corpus of American football broadcasts." arXiv preprint arXiv:1909.03343 (2019).

3. Chen, Zhutian, et al. "Sporthesia: Augmenting sports videos using natural language." IEEE transactions on visualization and computer graphics 29.1 (2022): 918-928.

4. Adam, Edriss Eisa Babikir. "Deep learning based NLP techniques in text to speech synthesis for communication recognition." Journal of Soft Computing Paradigm (JSCP) 2.04 (2020): 209-215.

5. Kim B. J., Choi Y. S. Automatic baseball commentary generation using deep learning //Proceedings of the 35th Annual ACM Symposium on Applied Computing. – 2020. – С. 1056-1065.

6. Дослідження генеративного ШІ у спорті та активностях [Электронный ресурс] – Режим доступа : www/ URL: https://research.ibm.com/projects/generative-ai-for-sports-and-entertainment#publications – 10.07.2024 г. – Generative AI for Sports and Entertainment.

7. Система збору та обробки спортивних даних [Электронный ресурс] – Режим доступа : www/ URL: https://www.statsperform.com/opta-vision/ – 10.07.2024 г. – GENERATIVE AI POWERED FOOTBALL INSIGHTS: AT SCALE.

8. Система декодування та візуалізації спортивних подій [Электронный ресурс] – Режим доступа : www/ URL: https://tracab.com/ – 10.07.2024 г. – UNLOCKING THE DNA OF SPORTS.

9. Система трекінгу спортсменів Sport LogIq [Электронный ресурс] –

Режим доступа : www/ URL: https://sportlogiq.com/ – 10.07.2024 г. – AI Powered Sports Analytics.

10. MoCA Project: Analysis and Retargeting of Ball Sports Video [Электронный ресурс] – Режим доступа : www/ URL: https://pi4.informatik.uni-mannheim.de/pi4.data/content/projects/moca/Project-SportsAdaptation.html – 10.07.2024 г. – The MoCA Project.

11. Olesia BARKOVSKA, Oleksandr BILICHENKO, Heorhii UVAROV, Tymur MAKUSHENKO Improved rendering method of skeletal animation on control points base. Computer systems and information technologies, 2024, (1), 71-81. https://doi.org/10.31891/csit-2024-1-9

12. Loland, Sigmund. "Classification in sport: A question of fairness." European journal of sport science 21.11 (2021): 1477-1484.

13. Manenti, Giorgio, et al. "Functional modelling and IDEF0 to enhance and support process tailoring in systems engineering." 2019 International Symposium on Systems Engineering (ISSE). IEEE, 2019.

14. Mataruna-Dos-Santos, Leonardo Jose, et al. "Big data analyses and new technology applications in sport management, an overview." Proceedings of the 2020 International Conference on Big Data in Management. 2020.

15. Chacoma, Andrés, and Orlando V. Billoni. "Simple mechanism rules the dynamics of volleyball." Journal of Physics: Complexity 3.3 (2022): 035006.

16. Sharma, Rahul, Ram Bilas Pachori, and Pradip Sircar. "Automated emotion recognition based on higher order statistics and deep learning algorithm." Biomedical Signal Processing and Control 58 (2020): 101867.

17. Ketkar, Nikhil, et al. "Convolutional neural networks." Deep Learning with Python: Learn Best Practices of Deep Learning Models with PyTorch (2021): 197-242.

18. Pauker, Stephen G., and Jerome P. Kassirer. "Decision analysis." Medical uses of statistics (2019): 159-179.

19. Chandra, Mayank Arya, and S. S. Bedi. "Survey on SVM and their application in image classification." International Journal of Information

Technology 13.5 (2021): 1-11.

20. Датасет Volleyball Match Tracking Computer Vision Project створений на платформі Roboflow – Режим доступа : www/ URL: https://universe.roboflow.com/vollleyball-tracking/volleyball-match-tracking – 10.07.2024 г. – Volleyball Match Tracking Computer Vision Project.

21. Датасет від Ultralytics. Open Images v7 [Электронный ресурс] – Режим доступа : www/ URL: https://docs.ultralytics.com/datasets/detect/open-images-v7/ – 10.07.2024 г. – Open Images v7 Dataset.

22. Персональний датасет volleyball_gameplay_detection створений за допомогою платформи Roboflow [Электронный ресурс] – Режим доступа : www/ URL: https://app.roboflow.com/volleyballgameplaydetection/volleyball_gameplay_detection/1 – 10.07.2024 г. – volleyball_gameplay_detection Image Dataset.

23. Платформа Roboflow для створення та роботи з датасетами [Электронный ресурс] – Режим доступа : www/ URL: https://universe.roboflow.com/ – 10.07.2024 г. – Explore the Roboflow Universe.

24. Mostafa S. Ibrahim and Srikanth Muralidharan and Zhiwei Deng and Arash Vahdat and Greg Mori. A Hierarchical Deep Temporal Model for Group Activity Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

25. Сервіс від AWS для штучного інтелекту — Amazon Rekognition [Электронный ресурс] – Режим доступа : www/ URL: https://aws.amazon.com/rekognition/ – 10.07.2024 г. – Amazon Rekognition.

26. Google Cloud. Cloud Vision Api [Электронный ресурс] – Режим доступа : www/ URL: https://cloud.google.com/vision – 10.07.2024 г. – Extract insights from images, documents, and videos.

27. Сервіси штучного інтелекту Azure [Электронный ресурс] – Режим доступа : www/ URL: https://azure.microsoft.com/en-us/products/ai-services – 10.07.2024 г. – Azure AI Services.

28. IBM Watson Visual Recognition service [Электронный ресурс] –

Режим доступа : www/ URL: https://mediacenter.ibm.com/media/IBM+Watson+Visual+Recognition/0_jbsmp6lq – 10.07.2024 г. – IBM.

29. Бібліотека для Computer Vision - OpenCV [Электронный ресурс] – Режим доступа : www/ URL: https://opencv.org/ – 10.07.2024 г. – OpenCV is the world’s biggest computer vision library.

30. Alzubaidi, Laith, et al. "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions." Journal of big Data 8 (2021): 1-74.

31. API Для роботи з моделями GPT [Электронный ресурс] – Режим доступа : www/ URL: https://openai.com/index/openai-api/ – 10.07.2024 г. – OpenAI API.

32. Система роспізнавання та класифікації об’єктів [Электронный ресурс] – Режим доступа : www/ URL: https://www.ultralytics.com/yolo – 10.07.2024 г. – Train AI models in seconds with Ultralytics YOLO.

33. Модель TTS, документація [Электронный ресурс] – Режим доступа : www/ URL: https://docs.coqui.ai/en/stable/models/xtts.html – 10.07.2024 г. – XTTS

34. Фреймворк для роботи з аудіо та відео записами FFmpeg [Электронный ресурс] – Режим доступа : www/ URL: https://ffmpeg.org/ – 10.07.2024 г. – FFmpeg.

35. Zhou, Pan, et al. "Towards theoretically understanding why sgd generalizes better than adam in deep learning." Advances in Neural Information Processing Systems 33 (2020): 21285-21296.