

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Харківський національний університет радіоелектроніки
Факультет Комп'ютерних наук
Кафедра Програмної інженерії

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

другий (магістерський)

(рівень вищої освіти)

Дослідження моделей та алгоритмів машинного навчання для автоматичного
анотування зображень спортивних заходів

Виконав:

студентка 2 курсу групи ІПЗм-21-3

Лаптії А.А.

(прізвище, ініціали)

Спеціальність 121 – Інженерія програмного

забезпечення

Тип програми Освітньо-наукова

Керівник д.т.н. Смеляков К.С.

(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____

З.В. Дудар

2023 р.

Харківський національний університет радіоелектроніки

Факультет _____ Комп'ютерних наук _____
(повна назва)

Кафедра _____ Програмної інженерії _____
(повна назва)

Рівень вищої освіти _____ другий (магістерський) _____

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

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

Освітня програма _____ Інженерія програмного забезпечення _____
(повна назва)

ЗАТВЕРДЖУЮ:

Зав. кафедри _____
(підпис)

«_____» _____ 20 ____ р.

ЗАВДАННЯ**НА КВАЛІФІКАЦІЙНУ РОБОТУ**

студентки _____ Лантій Анастасії Андріївни _____
(прізвище, ім'я, по батькові)

1. Тема роботи «Дослідження моделей та алгоритмів машинного навчання для автоматичного анотування зображень спортивних заходів» _____

затверджена наказом університету від «03» квітня 2023 р. № 83Стз

2. Термін подання роботи до екзаменаційної комісії «15» травня 2023 р.

3. Вихідні дані до роботи методи виявлення та розпізнавання обличчя та номерів, датасети з спортивних заходів, метрики, критерії їх оцінки, Python, JavaScript, TypeScript, C# _____

4. Перелік питань, що потрібно опрацювати в роботі вступ, аналіз предметної області, аналіз способів виявлення та розпізнавання обличчя та номерів, проведення експериментів та аналіз їх результатів, програмна реалізація прототипу, формування подальших рекомендацій. _____

5. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Спецчастина	д.т.н. Смеляков К.С.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Аналіз предметної області	25.02.2023	виконано
2	Постановка задачі	10.03.2023	виконано
3	Проведення дослідження	01.04.2023	виконано
4	Підготовка пояснювальної записки	28.04.2023	виконано
5	Підготовка презентації та доповіді	02.05.2023	виконано
7	Перевірка на академічний плагіат	03.05.2023	виконано
8	Нормоконтроль	04.05.2023	виконано
9	Рецензування	05.05.2023	виконано
10	Знесення диплома в електронний архів	07.05.2023	виконано
11	Допуск до захисту у зав. кафедри	14.05.2023	виконано

Дата видачі завдання 17.01.2023 р.

Студент _____

(підпис)

Керівник роботи _____

(підпис)

д.т.н. Смеляков К.С.

РЕФЕРАТ / ABSTRACT

Кваліфікаційна робота магістра містить: 58 стор., 15 рис., 3 табл., 17 джерел, 5 додатків.

СПОРТИВНІ ЗАХОДИ, СПОРТ, МАРАФОН, РОЗПІЗНАВАННЯ ОБЛИЧЧЯ, РОЗПІЗНАВАННЯ НОМЕРІВ, НЕЙРОМЕРЕЖІ, ДАТАСЕТ, ПОШУК ЗА ЛИЦЕМ, ПОШУК ЗА НОМЕРОМ.

Об'єктом дослідження є способи виявлення та розпізнавання обличчя та номерів за фотографією.

Метою дослідження є аналіз різних за складністю та структурою способів для виявлення та розпізнавання обличчя та номерів за фотографією.

Методи розробки базуються на основі навчання нейромереж, подальшого їх перенавчання шляхом подання великою кількістю даних з датасетів.

У результаті роботи було проведено аналіз існуючих способів та методів, що можуть розпізнавати та виявляти по фото номери та обличчя. Результати експерименту було продемонстровано в створеному прототипі.

SPORT EVENTS, SPORT, MARATHON, FACE RECOGNITION, NUMBER RECOGNITION, NEURAL NETWORKS, DATASET, SEARCH BY FACE, SEARCH BY NUMBER.

The object of the study is methods of detecting and recognizing faces and numbers from a photograph.

The purpose of the study is to analyze methods of different complexity and structure for detecting and recognizing faces and numbers from photographs.

The development methods are based on the training of neural networks, their subsequent retraining by presenting a large amount of data from datasets.

As a result of the work, an analysis of existing methods and methods that can recognize and detect numbers and faces from a photo was carried out. The results of the experiment were demonstrated in the created prototype.

Я, Лаптії Анастасія Андріївна, студентка групи ІПЗм-21-3, здобувачка вищої освіти на другому (магістерському) рівні, кафедра Програмної інженерії, заявляю: моя кваліфікаційна робота на тему «Дослідження моделей та алгоритмів машинного навчання для автоматичного анотування зображень спортивних заходів», що буде представлена до ЕК для публічного захисту, виконана самостійно, в ній не містяться елементи плагіату і вона може бути опублікована в електронному архіві відкритого доступу ElArKhNURE. Всі запозичення з друкованих та електронних джерел мають відповідні посилання.

Я ознайомлений з діючим положенням «Про протидію академічному плагіату в ХНУРЕ», згідно з яким виявлення плагіату є підставою для відмови в допуску кваліфікаційної роботи до захисту та застосування дисциплінарних заходів.

ЗМІСТ

ВСТУП	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ	10
1.1 Загальний аналіз галузі.....	10
1.2 Виявлення проблем.....	16
2 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ТА АЛГОРИТМІВ	20
2.1 Огляд теоретичних відомостей машинного та глибокого навчання	20
2.2 Огляд методів анотування зображень.....	23
2.3 Аналіз і порівняльна оцінка ML-моделей і нейромереж	24
2.4 Інформаційна підготовка прийняття рішень	26
2.5 Вибір алгоритму	33
3 ЕКСПЕРИМЕНТ	35
3.1 Розробка вимог до системи та реалізація дослідження	35
3.2 Обрання датасету, метрик, моделі, нейромережі	37
3.3 Опис експерименту	40
4 ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОТОТИПУ	43
4.1 Проектування прототипу.....	43
ВИСНОВКИ	50
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	51
ДОДАТОК А	54
ДОДАТОК Б	55
ДОДАТОК В	56
ДОДАТОК Г	63
ДОДАТОК Д	64

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

ШІ	Штучний інтелект
CNN	Згорткова нейронна мережа (Convolutional neural network)
OCR	Optical Character Recognition
COCO	Common Objects in Context
MNIST	Graphics Interchange Format
YOLO	You Only Look Once

ВСТУП

Метою реалізації проєкту за темою машинного навчання з розпізнавання обличчя та номерів на спортивних заходах є формування у населення культури здорового способу життя, створення умов для розвитку сучасної і доступної спортивно-оздоровчої інфраструктури, збільшення кількості населення, яке регулярно займається руховою активністю. Є кілька причин, чому такий підхід буде мати ряд переваг над іншими: підвищення ефективності за рахунок автоматизованих підходів (за допомогою методів машинного та глибинного навчання можна швидко оброблювати великі обсяги даних, дозволяючи швидше доставляти кінцеву вихідну інформацію користувачеві), більша масштабованість, обробка великої кількості даних, аналіз, збереження даних, постійний моніторинг, автоматизація процесів дозволяє контролювати стан конкретного функціоналу з будь-якою частотою [7], покращена відтворюваність, результати ML є послідовними, оскільки вони пропонують повторювані та відтворювані результати, на відміну від ручних підходів, що полегшує відтворення і перевірити результати, тестування відносно дешево в порівнянні з ручними підходами, функціонал можна застосовувати до більшості джерел даних, навіть до тих, які не підлягають вручну, фотоінтерпретація є дуже гарною для сприйняття, можливе покращення та збільшення точності з часом, можлива комбінація з традиційними підходами, щоб забезпечити доповнення інформації.

Створення дослідницького додатку, надасть змогу людям, що цікавляться спортом і безпосередньо учасникам відповідних заходів (спортсменам і офіційним особам), глядачам, тренерам, журналістам, сценаристам, фотографам отримати доступ до цифрових матеріалів з чемпіонатів, Олімпійських ігор, благодійних марафонів, учасниками яких вони були чи, навпаки, їх пропустили, та таким чином покращити якість своїх спогадів, репортажів, аналітичних звітів, спортивних прогнозів завдяки централізованій системі зберігання фото з місць проведення

змагань. Головною особливістю системи є пошук та розпізнавання людини по обличчю або ж номеру (так званий, bib-код).

Першою задачею роботи є дослідження датасетів, нейронних мереж, завдяки яким можливе розпізнавання, відстежування та детект об'єктів на фото. Для обробки та порівняння результатів дослідження треба обрати метрики, за якими буде проводитись дослід (наприклад, accuracy, precision, recall). Для прийняття найкращого рішення у якості вихідних даних дослідження мають бути представлені порівняльні результати нейронних мереж, щоб визначити ступінь складності їх навчання та його часові параметри (чи складно їх навчати, чи довго), щоб визначитись з тим, чи навчати нейронну мережу власноруч чи може слід краще використовувати якусь із перенавчених; варто також визначити, які методи розпізнавання краще підійдуть для обраної предметної області.

Другою задачею роботи є створення системи, яка зможе знаходити та або розпізнавати людину за фото, часом або ж її номером – у бігунів, наприклад, такий клеїться на спортивний одяг на будь-якій зручній частині тіла, такі особливі номери зазвичай містять чіп, за допомоги якого потім можна відстежувати як точний час перетину бігуном межі фінішу, так і зафіксовані дані про стан роботи його організму: пульс, тиск та багато іншого. За ідеальних умов треба винайти найоптимальніший алгоритм.

1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ

1.1 Загальний аналіз галузі

Спортивні заходи дуже популярні у всьому світі як серед спортсменів, так і серед спортивних туристів. Найвідоміші з них - зимові та літні Олімпійські ігри, чемпіонати світу з футболу і хокею. На це вказують зростання обсягів продажу вхідних квитків на змагання та офіційні заходи, пов'язані з ними. Переважна кількість уболівальників на таких змаганнях є досить заможною частиною планети, але є і люди зі скромним статком (студенти, робітники), які є фанатами, наприклад, футбольних команд, а, значить, хотіли б також отримати якісні спогади зі спортивних миттєвостей.

Один із найбільш поширених серед будь-яких верств населення світу видів спорту – гірськолижний, де одні люди відпочивають, інші – змагаються. Найпопулярнішими гірськолижними країнами вважаються: Австрія, Андорра, Франція, Швейцарія, Болгарія, Італія, Словаччина, Туреччина, Фінляндія, Швеція, Норвегія. Вони мають розвинені сервіси для занять таким різновидом спорту та відпочинку, проте лижні курорти є і в США, Канаді і, навіть, в Ізраїлі, Індії та Японії.

Крім цього, щороку у світі проводяться континентальні, міжнародні спортивні змагання з різних видів спорту, Ігри героїв, Ігри нескорених, Ігри Співдружності, Арктичні зимові ігри, міжнародні спортивно-масові заходи серед студентів, відбіркові чемпіонати на найвідоміші турніри. Про масовість та популярність спортивних заходів свідчить кількість учасників лише одного Лондонського марафону, який відбувся 5 жовтня 2020, що перевищила 40 тисяч осіб. Про динаміку зростання ролі спорту свідчить порівняння кількості учасників в Олімпійських іграх 1904 року (Сент-Луїс, США), де змагалось 62 спортсмена, та 2016 року, де їх було 11544 чоловік (Ріо-де-Жанейро, Бразилія).

Навіть за наявності «ковідних» протиепідемічних обмежень лише у зимових Олімпійських іграх в лютому 2022 (Пекін, Китай) взяло участь понад 4 000 атлетів.

На тлі пандемії квитки на цю Олімпіаду не продавали, а тому за відсутності іноземних уболівальників, відвідали змагання тільки близько 150 тисяч китайських глядачів за спеціальними запрошеннями. (Довідка: перша зимова Олімпіада стартувала 25 січня 1924 року у французькому Шамоні за участю 300 спортсменів із 16 країн, під час якої канадці стали незаперечними лідерами в хокеї, закинувши 110 шайб і пропустивши лише 3).

Сьогодні на земній кулі існує велике різноманіття видів спорту, лише в Україні постійно йде підготовка спортсменів з 53 олімпійських видів спорту (40 – літніх, 13 – зимових) та 71 виду спорту, що не входить до програми олімпійських ігор («наказ державного комітету молодіжної політики, спорту і туризму України, № 261 від 07.02. 2001»).

Сучасний спорт – це певний феномен масової культури. Він об'єднує суспільство, створюючи певні комунікативні групи, він служить задоволенню потреб людини, впливає на її імідж та відповідний стиль життя. Це не тільки прибуткова сфера життя, але й цікава з точки зору накопичення та обробки даних. Завдяки досягненню сучасних інформаційно-комунікаційних технологій та стрімкому розвитку відповідних сервісів та послуг, спортом охоплені найширші аудиторії. Разом з популяризацією спорту в різних засобах масової інформації зростає і його комерційна складова, він стає частиною «індустрії розваг», видовищем, яке заслуговує особливої уваги медійників. А це, в свою чергу, швидкими темпами наповнює інформаційний простір цифровими зображеннями як «картинами подій», так і фото чи відео відповідних особистостей.

Майже всі події сучасного світу освітлюють в медіа просторі, жодна подія не відбувається без спалахів камер, тому аналітика цих даних та керування ними є областю мого інтересу.

Марафони стали популярними з різних причин:

– здоров'я та фітнес: у зв'язку зі збільшенням уваги до здоров'я та фітнесу все більше і більше людей займаються бігом як способом залишатися у формі. Марафони дають мету, до якої потрібно працювати, і відчуття досягнення після завершення гонки.

– виклик: пробігти марафон – це ціле досягнення, яке вимагає відданості, дисципліни та витривалості. Багатьох людей приваблює виклик і відчуття успіху, яке приходить із завершенням гонки.

– спільнота: марафони створюють відчуття спільності та товариства серед бігунів. Тренувальний процес і сам забіг дають можливість зустрітися з однодумцями та подружитися.

– благодійність: організовується багато марафонів, щоб зібрати гроші на благодійність. Це дає можливість бігунам підтримати справу, у яку вони вірять, одночасно досягаючи особистої мети.

– подорожі: марафони проводяться в містах по всьому світу, що дає можливість бігунам подорожувати та пізнавати різні культури, беручи участь у змаганнях.

У світі є багато популярних марафонів, які приваблюють бігунів з усіх куточків світу. Ось кілька прикладів найпопулярніших марафонів:

Бостонський марафон: Бостонський марафон — один із найстаріших і найпрестижніших марафонів у світі, який щорічно проводиться в День патріота в Бостоні, штат Массачусетс, США. Змагання приваблюють кращих бігунів з усього світу та мають суворий відбірковий час для учасників.

Нью-Йоркський марафон: Нью-Йоркський марафон – це найбільший марафон у світі, у якому беруть участь понад 50 000 бігунів з усього світу. Траса проходить через усі п'ять районів Нью-Йорка і закінчується в Центральному парку.

Лондонський марафон: Лондонський марафон є одним із найбільших і найвідоміших марафонів у світі, який щорічно проводиться в Лондоні, Великобританія. Курс пролягає вулицями Лондона та повз багато відомих пам'яток, зокрема Букінгемський палац і Лондонський Тауер.

Токійський марафон: Токійський марафон є одним із найбільших і найпрестижніших марафонів в Азії, який щорічно проводиться в Токіо, Японія. Перегони приваблюють кращих бігунів з усього світу та мають унікальну трасу, яка проходить як сучасними, так і традиційними районами міста.

Берлінський марафон: Берлінський марафон є одним із найшвидших і найпопулярніших марафонів у світі, який щорічно проводиться в Берліні, Німеччина. Траса відома своєю плоскою та швидкою трасою, на якій було встановлено кілька світових рекордів.

Це лише кілька прикладів багатьох популярних марафонів у всьому світі. Серед інших відомих марафонів – Чиказький марафон, Паризький марафон і Марафон Гонолулу.

Щоб відстежувати марафонця програмним шляхом, потрібно буде використовувати систему відстеження або додаток, який надає інформацію про прогрес бігуна в реальному часі.

Кілька поширених способів програмного відстеження марафонців: GPS-відстеження: багато марафонців мають пристрій GPS або додаток для смартфона, який може відстежувати їхнє місцезнаходження та надавати оновлення в режимі реального часу щодо прогресу під час забігу. Ви можете використовувати API або веб-сервіс, який підключається до системи відстеження, щоб отримувати дані про місцезнаходження бігуна, темп і приблизний час прибуття на різні контрольні точки.

Системи хронометражу: у більшості марафонів використовується система хронометражу, яка фіксує час бігуна на різних контрольних точках протягом дистанції. Ви можете використовувати API або веб-службу, яка підключається до системи хронометражу, щоб отримати дані про поточну позицію бігуна та очікуваний час прибуття до фінішу.

Соціальні мережі. Багато марафонців використовують платформи соціальних мереж, щоб ділитися оновленнями та фотографіями під час забігу. Ви можете використовувати API, надані платформами соціальних мереж, щоб контролювати публікації бігунів і відстежувати їх прогрес.

Загалом, програмне відстеження марафонця потребує доступу до системи відстеження або програми, яка надає інформацію в реальному часі про його місцезнаходження, темп і приблизний час прибуття. Залежно від системи, яка

використовується, ви можете отримати доступ до цієї інформації через API або веб-службу.

Попри наднавантажений графік життя сучасної людини, вже декілька десятиріччя поспіль невід'ємною складовою повсякденного життя сьогодення є спорт. Турбота про своє тіло має бути такою ж невід'ємною частинною нашого життя, як і піклування про ментальне здоров'є. В умовах війни понад 90% українців мають хоча б один із симптомів тривожних розладів, але 31% не вважають свої ментальні проблеми вагомими для звернення по допомогу і важливими для інших. Фізичні вправи дійсно покращують психічний стан людини, але панацеєю та чарівною пігулкою від депресії вони не є. Фізична активність позитивно впливає не тільки на м'язи, але і є корисними фактором активізації мозкової діяльності. Люди які займаються спортом, більш стійкі до стресу.

Зростає кількість досліджень, які свідчать про те, що спорт і фізичні вправи та фізична активність можуть мати позитивний вплив на психічне здоров'я, включаючи депресію.

Нижче наведено декілька фактів, чому спорт і фізичні вправи можуть бути корисними:

- збільшення ендорфінів: фізичні вправи викликають вивільнення ендорфінів, природних хімічних речовин, які виробляє організм, які допомагають зменшити біль і підвищити відчуття задоволення та щастя. Це може допомогти полегшити симптоми депресії.

- поліпшення сну: доведено, що регулярні фізичні вправи покращують якість сну, що може бути особливо корисним для людей з депресією, які можуть страждати від порушень сну.

- зменшення стресу: фізична активність може допомогти зменшити стрес і тривогу, які є загальними симптомами депресії.

- соціальна взаємодія. Багато видів спорту та фізичних вправ включають соціальну взаємодію, яка може бути важливою складовою лікування депресії. Спілкування з іншими за допомогою вправ може допомогти покращити настрій і зменшити відчуття ізоляції.

– почуття виконаного завдання: постановка та досягнення цілей за допомогою спорту чи фізичних вправ може допомогти розвинути впевненість у собі та дати людям відчуття виконаного обов'язку, що може бути особливо корисним для тих, хто бореться з депресією.

– відволікання: фізична активність може допомогти відволікти розум від негативних думок і емоцій, забезпечуючи тимчасовий перепочинок від симптомів депресії.

Загалом, хоча фізичні вправи та фізична активність не повинні розглядатися як заміна професійного лікування депресії, вони можуть бути ефективним способом доповнити лікування та покращити загальне психічне здоров'я та самопочуття.

Щорічно українські обласні департаменти молоді та спорту спрямовують максимальні зусилля в сфері молодіжної політики щодо популяризації, набуття необхідних компетенцій, активного залучення міжнародних організацій і підвищення загального престижу спортивних заходів у молоді.

Суспільні відносини у всьому світі характеризуються процесами глобалізації, інтернаціоналізації, стрімким розвитком та проникненням у всі сфери життя сучасних цифрових технологій. Не є виключенням й публічне управління. Цифровізація всіх сфер публічного управління, у тому числі фізичної культури та спорту, дає змогу суттєво підвищити відкритість, прозорість й ефективність публічного управління, а також значно покращити якість публічно-управлінських послуг. Цікавим напрямком у досягненні цього є цифрова трансформація сфери фізичної культури і спорту. Міністерством молоді та спорту України розроблено чотири проекти у сфері молоді та спорту, які стали складовою частиною 94-х проектів цифровізації держави, потрапивши на публічну сторінку цифрової трансформації.

1.2 Виявлення проблем

Існують різні моделі та алгоритми анотування зображень. Один із основних застосувань знаходження об'єктів є пошук та ідентифікація обличчя. За допомогою сучасних алгоритмів ми можемо відслідковувати та знаходити людські обличчя на фото та відео. Необхідно визначити перелік алгоритмів для дослідження їх корисності при автоматичному анотуванні зображень спортивних заходів.

Проблематика задачі постає у множині можливих варіантів при обранні моделі та нейромережі [1]. Існує дуже багато різних досліджень, з приводу того, яка ж модель є найкращою, але всі мережі та моделі особливо себе розкривають на конкретних предметних областях, у конкретному середовищі, тому спиратися у загальному плані на більшість досліджень можна, але, щоб зрозуміти, що найкраще підійде для власних даних, треба провести декілька експериментів на однакових даних, проте на різних моделях та мережах.

Розглянемо методи вирішення поставлених задач:

Марафонців можна легко розпізнати за кількома фізичними та поведінковими характеристиками. Ось деякі з поширених способів ідентифікації марафонця:

Бігове спорядження: марафонці зазвичай носять кросівки, шорти або колготки, а також легку сорочку чи майку. Вони також можуть носити капелюх або козирок, щоб захистити обличчя від сонця.

Тип статури: марафонці, як правило, худі та м'язисті, з сильними ногами та високим рівнем серцево-судинної витривалості. Вони також можуть мати характерну «статуру бігуна» з довгими худими кінцівками та вузькою талією.

Поведінка: марафонці можуть проявляти певну поведінку, наприклад розтягуватися або розминатися перед забігом, носити воду або енергетичні гелі, носити хронометраж або спортивний нагрудник.

Зовнішній вигляд: марафонці можуть мати видимі ознаки фізичного навантаження, такі як піт, почервоніння шкіри та прискорене дихання. Вони також можуть проявляти ознаки втоми або виснаження після завершення гонки.

Для вирішення першої задачі – дослідження нейронних мереж, слід обрати топ 5 таких мереж і в кожному сегменті обрати метрики, завдяки яким буде можливе порівняння.

Розпізнати людину за зображенням може бути складно, але є деякі прийоми, за допомогою яких можна підвищити точність.

Ось кілька способів розпізнати людину за картинкою:

Програмне забезпечення для розпізнавання облич: програмне забезпечення для розпізнавання облич використовує алгоритми для визначення та зіставлення рис обличчя на зображенні з базою даних відомих облич. Останніми роками ця технологія стає все більш досконалою, і її можна використовувати для ідентифікації осіб з високим ступенем точності.

Контекстна інформація. Іноді для ідентифікації людини на фотографії потрібна додаткова контекстна інформація. Наприклад, якщо фотографія була зроблена під час певної події, як-от марафону, на людині може бути спортивний нагрудник або інший ідентифікаційний одяг чи аксесуари. Ця інформація може допомогти звужити коло потенційних збігів і підвищити точність.

Порівняння з іншими фотографіями: якщо є доступ до інших фотографій людини, ви можете порівняти риси обличчя та інші характеристики з цією фотографією. Це може допомогти підтвердити збіг або виявити розбіжності, які можуть вказувати на іншу особу.

Розпізнати людину за зображенням може бути складно, але використання програмного забезпечення для розпізнавання обличчя, контекстна інформація, знайомство з людиною та порівняння з іншими фотографіями можуть допомогти підвищити точність.

Для вирішення другої задачі - створення системи для розпізнавання та детекту людей за певними ознаками, треба дослідити мови програмування, які найраше для цього підходять, написати програмний код та протестувати все на реальних даних

хоча б у тестовому середовищі. Окремо варто приділити увагу розпізнаванню та детекту по обличчю та по біб-номеру.

Задачі хоча і є самодостатніми і можуть вирішуватися окремо одна від одної, їх все ж таки варто поєднати. Краще за все буде почати з маленького дослідження щодо нейронних мереж, а потім перейти до самого експерименту, де до переднавченої чи навченої власноруч системи буде надходити фото, яке система буде обробляти та видавати кінцевий результат.

Для огляду моделей та алгоритмів машинного навчання оберемо декілька та порівняємо. Наприклад, можна використати алгоритм Дейкстри [2] для оптимізації пошуку за структурами даних у задачах обробки зображень [14, 15], включаючи оптимізацію пошуку за структурами після використання нейронних мереж [16]. Почати варто з загальної класифікації, де моделі є вже навчені – часто фреймворки представляють такі готові рішення і треба навчитися лише ними користуватися, а є моделі ненавчені [8], тобто це сукупність нейронів, які можна навчити, а їх вчителем безпосередньо буде виступати дослідник. Навчені моделі зазвичай мають можливість та потребують «донавчання» за власним розсудом, бо інколи розв’язку з-під коробки не вистачає.

Виявлені проблеми:

- немає загального ресурсу
- користування вже наявними альтернативами
- неявний збір даних іншими корпораціями та підприємствами
- створення платформи, що дасть можливість координувати дії між меценатами та отримувачами меценатської допомоги
- створення електронного “Єдиного календарного плану” спортивних заходів з бюджетуванням
- розроблення інформаційних технологій, що дають можливість адмініструвати пілотний проект
- оцифрування показників діяльності федерацій з видів спорту, розроблення уніфікованого протоколу для збору даних федераціями, розроблення захищеного механізму передачі даних та механізму автоматизованого формульного розподілу

державних коштів між видами спорту;

- створення стимулів для участі населення у проведенні рейтингів шляхом впровадження мотиваційних проектів, використання ігрових практик та механізмів

- створення стимулів для розвитку вітчизняних ініціатив у розробленні та впровадженні спортивних цифрових технологій шляхом виконання таких завдань:

- розроблення механізму заохочення вітчизняних ініціатив щодо створення та розвитку спортивних новітніх стартапів шляхом залучення аудиторії для тестування продукту;

- розроблення механізму для залучення потенційних інвесторів; використання вітчизняних новітніх цифрових інформаційних продуктів під час реалізації пілотних проектів сфери фізичної культури і спорту

- обґрунтування, розроблення, впровадження інноваційної технології збору та аналізу статистичної інформації щодо зайнятих фахівців у сфері фізичної культури і спорту

- прийняття сучасної концепції розвитку фахової освіти у сфері фізичної культури і спорту, де формальна освіта є першим етапом, другим - входження фахівця у професію на основі неформальної освіти, зокрема наставництва, наступними етапами є безперервний професійний розвиток шляхом неформальної освіти та самоосвіти

2 АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ТА АЛГОРИТМІВ

2.1 Огляд теоретичних відомостей машинного та глибокого навчання

Машинне навчання включає в себе глибоке навчання. Це галузь інформатики, яка зосереджена на самонавчанні та вдосконаленні через вивчення комп'ютерних алгоритмів, регресивним чином моделі можуть перенавчатися та краще працювати. Глибоке навчання, на відміну від машинного, використовує штучні нейронні мережі, призначені для відтворення того, як люди думають і навчаються. Донедавна складність нейронних мереж була обмежена через обмеження потужності процесорів, але прогрес у аналізі великих даних дозволив створювати більші та складніші нейронні мережі, і тепер комп'ютери мають здатність контролювати, навчатися та швидко реагувати в складних обставинах. Навчання базується на багаторівневих штучних нейронних мережах. Глибокі нейронні мережі (DNN) — це мережі, які можуть виконувати складні операції, такі як представлення та абстрагування, щоб зрозуміти зображення, звуки та текст. Глибоке навчання, предмет машинного навчання, який найшвидше розвивається, є дійсно рушійною цифровою технологією, яку все більше компаній використовують для створення нових бізнес-моделей.

Експерти зі штучного інтелекту завжди зустрічаються з іншими дослідниками, які мають дуже цінні експериментальні дані, ретельно зібрані в лабораторіях, але обмеження часу та ресурсів зменшують життєздатність застосування методів машинного навчання або комп'ютерного зору до цих даних.

На створення машинного навчання надихнуло математичне моделювання нейронних мереж. У 1943 році логік Уолтер Піттс і невролог Уоррен МакКаллох опублікували есе, в якому вони намагалися систематично скласти карту розумових процесів і прийняття рішень у людському пізнанні.

У 1950 році Алан Тюрінг створив тест Тюрінга, який став лакмусовим папірцем для визначення того, які машини є «розумними» чи «неінтелектуальними». Щоб вважатися «розумною», машина повинна була

переконати людину в тому, що вона теж є людиною. Невдовзі літня дослідницька програма Дартмутського коледжу була визнана місцем народження ШІ.

З цього моменту почали розвиватися «інтелектуальні» алгоритми машинного навчання та комп'ютерні програми, які могли виконувати дії, починаючи від організації маршрутів подорожей продавців до гри з людьми в настільні ігри, наприклад у шашки.

Термін «глибинний» зазвичай використовується для позначення кількості прихованих шарів у нейронній мережі.

Глибокі мережі можуть мати до 150 прихованих шарів, тоді як традиційні нейронні мережі мають лише 2-3. Моделі глибокого навчання навчаються з використанням величезних обсягів мічених даних і топологій нейронних мереж, які вивчають функції безпосередньо з даних, таким чином усуваючи вимогу до вилучення функцій людиною.

(CNN або ConvNet) – це тип глибокої нейронної мережі, який широко використовується. CNN згортково поєднує вивчені функції з вхідними даними, що робить його ідеальним для обробки 2D-даних, як-от фотографій, тут у Saiwa два CNN надаються для навчання на спеціальному наборі даних користувача.

CNN зменшують вимоги до виділення ознак вручну, що означає, що вам не доведеться визначати характеристики класифікації зображення. CNN отримує характеристики безпосередньо з фотографій. Критичні характеристики не навчені заздалегідь; скоріше, вони виявляються, коли мережа тренується на пакеті зображень. Завдяки автоматизованому виділенню функцій моделі глибокого навчання є особливо успішними для програм комп'ютерного бачення, таких як класифікація об'єктів [4]. CNN вчать розрізняти різні візуальні сигнали за допомогою десятків чи сотень прихованих шарів. Кожен прихований шар ускладнює вивчені характеристики зображення. Наприклад, перший прихований шар може навчитися розпізнавати краї, тоді як останній шар навчиться виявляти складніші форми, характерні для геометрії предмета, який ми хочемо розпізнати.

Машинне навчання є гарячою темою в академічній та промисловій сферах, де постійно з'являються нові методи. Через швидкість і складність сфери

професіоналам важко встигати за новими стратегіями, тоді як новачки можуть бути перевантажені.

Модель машинного навчання, яку також називають «моделлю», — це математичне представлення, яке описує дані в контексті проблеми, як правило, комерційного завдання.

- найпоширеніші алгоритми глибокого навчання:
- нейронні мережі з використанням згорток (CNN)
- мережі довготривалої пам'яті (LSTM)
- повторювані нейронні мережі (RNN) (RNN)
- змагальні генеративні мережі (GAN)
- мережі з використанням радіальних базисних функцій (RBFN)
- перцептрони з кількома шарами (MLP)
- карти, які організуються самі (SOM)
- мережі глибокої віри (DBN)
- машини Больцмана з обмеженнями (RBM)
- автокодери

Глибоке навчання — це підкатегорія методів машинного навчання, заснованих на штучних нейронних мережах, які автоматично витягують функції високого рівня з необроблених вхідних даних.

Існують різні варіанти надання навчальних даних:

- безпосереднє завантаження з комп'ютера користувача або хмарного простору
- додавання посилання на загальнодоступний набір даних
- автоматичне навчання на даних користувача
- перевірка навчальної моделі за допомогою додаткових послуг, таких як виявлення об'єктів
- навчання на загальнодоступних наборах даних без вручення користувача.
- відстеження навчальних завдань за допомогою діаграми та інших даних з панелі користувача

– після того, як модель була необхідної точності для досягнення перевірки, кількість епох можна змінити.

2.2 Огляд методів анотування зображень

Згорткові нейронні мережі (CNN) є одними з найбільш ефективних і широко використовуваних архітектур глибокого навчання.

На момент лютого 2022 року, є такі відомості щодо бібліотек для розпізнавання тексту на зображенні. Такі бібліотеки потрібні та можна використовувати для дослідження виявляти та розпізнавати номери бігунів або інших спортсменів.

На Рисунку 2.1 представлені результати дослідження.

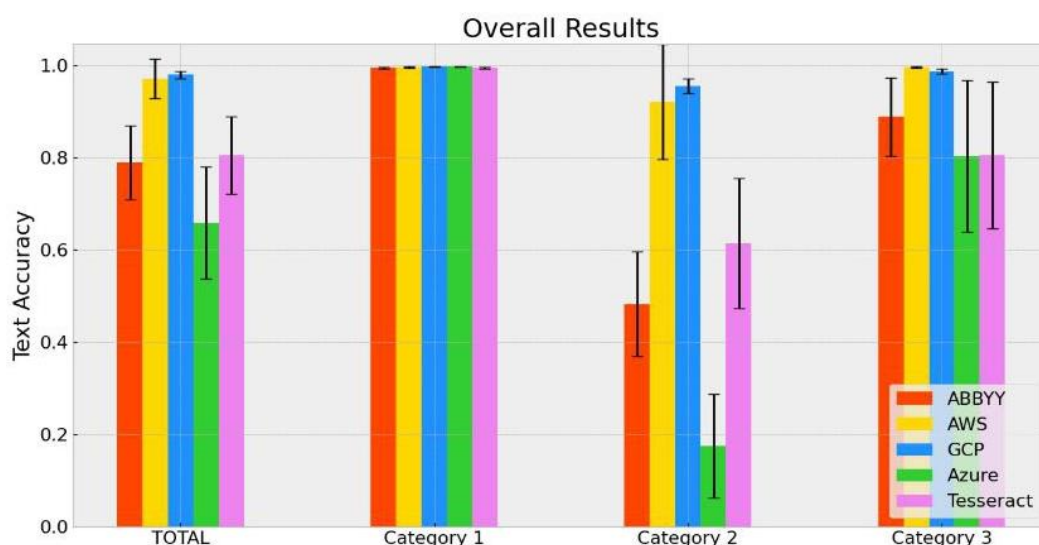


Рисунок 2.1 – Результати порівняння сучасних бібліотек машинного навчання

На Рисунку 2.1 представлена Категорія 1 – скріншоти з веб-сторінок, які включають в себе рандомно обрані сторінки з пошукової системи Гугл та сторінки з Вікіпедії. Всі запити також обрані рандомно. Категорія 2 представляє з себе

результати з рукописними написами на зображеннях. На Категорії 3 представлені виписки з зарплатнею, доходами, довідками.

Моделі глибокого навчання іноді називають «глибокими нейронними мережами», оскільки більшість підходів до глибокого навчання використовують топології нейронних мереж. У роботі [10] запропоновано ефективне рішення для пошуку даних на основі фотографій.

Глибоке навчання зробило важливий внесок у класифікацію зображень, мовний переклад і розпізнавання мовлення, а також його можна використовувати для вирішення діагностичних проблем без участі людини.

Топ інструментаріїв за результатами дослідження:

- ABBYY FineReader 15
- Amazon Textract
- Google Cloud Platform Vision API
- Microsoft Azure Computer Vision API
- Tesseract OCR Engine

2.3 Аналіз і порівняльна оцінка ML-моделей і нейромереж

З моменту популяризації навчання нейронних мереж (CNN) має місце безперервний процес покращення якості алгоритмів, які використовуються для виявлення об'єктів. Треба проаналізувати найпопулярніші з них і вибрати найкращі. Для цього необхідно:

- сформувавати альтернативи
- сформувавати критерії, за якими будемо оцінювати ваші альтернативи
- сформувавати шкали для оцінки по зазначеним критеріям
- провести оцінку, на базі якої буде прийматись рішення

Виявлення об'єктів — це завдання виявлення екземплярів об'єктів певного класу на зображенні. Сучасні методи можна розділити на два основні типи, де є

одноетапні методи та двоетапні методи. Одноетапні методи надають пріоритет швидкості висновку, і приклади моделей включають YOLO, SSD і RetinaNet.

Двоетапні методи надають пріоритет точності виявлення, і приклади моделей включають Faster R-CNN, Mask R-CNN і Cascade R-CNN. Моделі зазвичай оцінюються відповідно до показника середньої точності.

Множина досліджуваних альтернатив складається з таких алгоритмів виявлення об'єктів:

- YOLO v5: за останні кілька років з'явилося багато нових моделей виявлення об'єктів, кожна з яких має свої переваги та недоліки, але досі найкращою моделлю виявлення об'єктів з точки зору швидкості та точності є YOLOv4, YOLOv5, YOLOv7 та YOLO-R (це одне сімейство, з різними роками випуску), тому поки що працюватимемо з 5 версією, що вийшла у 2020 році.

Усі версії YOLO чудово інтегруються, забезпечують хорошу частоту кадрів та використовують небагато пам'яті [6].

- SSD: зокрема, нейронні мережі YOLO і SSD можуть розпізнавати екземпляри різних об'єктів і виділяти їх обмежувальні рамки на фотографіях, навіть якщо таких екземплярів кілька, вони мають різні розміри і частково перекриваються.

- Keras retinaNet

- Fast R-CNN

- Hog

Опис критеріїв вибору:

- популярність – частотність пропозицій щодо використання певного алгоритму при запиті в Google. Розставимо нашим алгоритмам бали за частотністю пропозицій щодо їх використання при запиті в Google за шкалою від 1 до 5 (найбільший бал – вища частотність)

- якість архітектури – важливий критерій, який впливає на час виконання обчислень, їх ефективність в певних сценаріях (можливість виявлення в обмеженому просторі чи швидкому переміщенні). Певні алгоритми потребують використання великих обчислювальних ресурсів, підвищуючи таким чином

загальну вартість процесу.

- швидкість виконання обчислень (кадри/с) – важлива для розрахунку часу отримання оптимального результату в режимі реального часу
- точність виявлення об'єктів – можливість забезпечення високої точності розпізнавання об'єктів, наявність чи відсутність фонових помилок, можливість розпізнавання декількох об'єктів, здатність розпізнавати дрібні об'єкти
- обмеження: кожен із алгоритмів має свої обмеження при використанні, наприклад, зниження роздільної здатності зображення при підвищеній продуктивності, залежність швидкості від операційної системи

Треба зазначити, що усі вищезазначені платформи доволі коштовні, тому у власному дослідженні будуть використані безкоштовні аналоги, що може менш ефективні, бо власноруч потрібно буде навчати чи донавчати моделі, проте це не потребуватиме матеріальних витрат. Проведемо дослідження з таким інструментарієм, як: TensorFlow, TensorFlow.js, Terran та інші.

2.4 Інформаційна підготовка прийняття рішень

Існує ряд інструментів штучного інтелекту, які дозволяють розпізнавати людей на фотографіях.

YOLO (You Only Look Once) — це популярний алгоритм глибокого навчання для виявлення об'єктів, зокрема розпізнавання людей на фотографіях. Однак для розпізнавання людей існують і інші алгоритми та методи.

Fast R-CNN: Fast R-CNN — ще один популярний алгоритм виявлення об'єктів, який використовує глибоке навчання для ідентифікації об'єктів на зображенні. Порівняно з YOLO, Fast R-CNN загалом точніший, але повільніший. Його можна використовувати для розпізнавання людей на фотографіях, але може вимагати більше часу та ресурсів, ніж YOLO.

OpenCV: OpenCV — це бібліотека комп'ютерного зору з відкритим кодом, яка надає ряд можливостей аналізу зображень і відео, зокрема розпізнавання

обличчя. Порівняно з YOLO, OpenCV може бути простішим у використанні та не потребує стільки навчальних даних чи обчислювальних ресурсів. Однак це може бути менш точним для завдань виявлення об'єктів, таких як розпізнавання людей.

Amazon Rekognition: Amazon Rekognition — це хмарна служба аналізу зображень і відео, яка використовує машинне навчання для аналізу зображень і відео для різноманітних цілей, зокрема для розпізнавання облич. Він може ідентифікувати людей на фотографіях і зіставляти їх із базою даних відомих облич, щоб забезпечити високу точність.

Microsoft Azure Face API: Microsoft Azure Face API — це хмарна служба, яка надає розширені можливості розпізнавання облич, зокрема розпізнавання облич, перевірку облич і групування облич. Його можна використовувати для ідентифікації людей на фотографіях і відео та зіставлення їх із базою даних відомих облич.

Google Cloud Vision API: Google Cloud Vision API – це хмарна служба аналізу зображень, яка використовує машинне навчання для аналізу зображень і надання статистичних даних. Він включає в себе можливості розпізнавання облич, які можуть ідентифікувати людей на фотографіях і зіставляти їх із базою даних відомих облич.

OpenCV: OpenCV — це бібліотека комп'ютерного зору з відкритим кодом, яка надає ряд можливостей аналізу зображень і відео, зокрема розпізнавання обличчя. Він містить попередньо підготовлені моделі, які можна використовувати для розпізнавання обличчя, а також можна використовувати для побудови власних моделей розпізнавання.

Це лише кілька прикладів інструментів AI, доступних для розпізнавання людей на фотографіях. Інші інструменти та служби включають Face++, Kairos і Clarifai. Використовуючи ці інструменти, важливо враховувати такі фактори, як точність, простота використання та вартість.

Методи розпізнавання людей на фотографіях за допомогою нейронної мережі Python, ви можете застосувати кілька підходів.

Згорткові нейронні мережі (CNN): CNN — це тип нейронних мереж, які добре підходять для завдань розпізнавання зображень, зокрема розпізнавання людей на фотографіях [17]. Ви можете навчити CNN на наборі даних позначених зображень, який містить приклади людей, а потім використовувати навчену модель для розпізнавання людей на нових фотографіях.

OpenCV: OpenCV — це бібліотека комп'ютерного зору з відкритим кодом, яка надає ряд можливостей аналізу зображень і відео, зокрема розпізнавання обличчя. Ви можете використовувати OpenCV для виявлення облич на зображеннях, а потім використовувати нейронну мережу для розпізнавання облич. Є кілька попередньо навчених моделей, доступних для використання з OpenCV, або ви можете навчити свою власну модель, використовуючи набір даних зображень із мітками.

Техніка навчання — це техніка, яка включає використання попередньо навченої нейронної мережі та її тонке налаштування для конкретного завдання. Ви можете використати попередньо підготовлену CNN, таку як VGG або ResNet, а потім налаштувати її на наборі даних із зображеннями з мітками, які містять приклади людей. Це може бути швидшим і ефективнішим способом навчання нейронної мережі розпізнаванню людей, ніж навчання з нуля.

Deep Learning Frameworks: існує кілька платформ глибокого навчання для використання з Python, зокрема TensorFlow, PyTorch і Keras [3]. Ці фреймворки надають інструменти для створення та навчання нейронних мереж, зокрема для розпізнавання людей на фотографіях.

Вибір методу залежатиме від низки факторів, включаючи конкретне завдання, доступні дані та необхідний рівень точності та швидкості.

Інтелектуальний аналіз даних та машинне навчання відноситься до складних наукових принципів і методів, які використовуються в області ШІ. Технологія розвивається завдяки постійному впливу цифрової інформації в кіберпросторі. У цій роботі варто розглянути фундаментальні концепції з анотування, детекту та розпізнавання обличчя та біб-кодів на зображеннях.

Векторний опис наших альтернатив, що є метриками виглядає наступним чином:

Таблиця 1 – Альтернативи з оцінкою за метриками

	Популярність	Якість архітектури	Швидкість виконання обчислень	Точність виявлення об'єктів	Обмеження
YOLO v5	4	Ефективна навчованість та	висока	Висока точність з найменшою	відсутні
SSD	3	Гірша за Fast R-CNN	35 кадрів/с	Висока при виявленні	Рекомендується для
Keras retinaNet	3	Краща за SSD	35 кадрів/с	Можливий неправильний вибір кандидатів	відсутні
Fast R-CNN	2	Потребує використання великих обчислювальних	7 кадрів/с Залежність швидкості від системи	Можливий неправильний вибір кандидатів	Затримки в пропозиції об'єктів Використа
Hog	1	Дуже трудоємна для складних	низька	Висока при розпізнаванні	Неефективна при обмеженні

Векторний опис містить якісні та кількісні показники.

Формуємо шкалу оцінок за критеріями. Наведемо опис шкали оцінок за критеріями. Переведемо якісні оцінки критеріїв у кількісні:

Якість архітектури:

- ефективна навчованість – 10 балів
- потребує використання

- великих обчислювальних ресурсів – 3 бали
- краща за SSD – 5 балів
- гірша за Fast R-CNN – 4 бали
- дуже трудомісна – 3 бали

Швидкість виконання обчислень (кадри/с)

- висока – 7 балів
- 35 кадрів/с – 5 балів
- 7 кадрів/с – 3 бали
- низька – 2 бали

Замінюємо якісні показники на кількісні.

Таблиця 2 – Бали альтернатив

	Популярність	Якість архітектури	Швидкість виконання обчислень	Точність виявлення об'єктів	Обмеження
YOLO v5	4	10	7	7	2
SSD	3	4	5	5	1
Keras retinaNet	3	5	5	8	2
Fast R-CNN	2	3	3	3	1
Hog	1	3	2	3	1

Точність виявлення об'єктів:

- висока точність з найменшою кількістю фонових помилок – 7 балів
- висока при виявленні крупних об'єктів – 5 балів
- можливий неправильний вибір кандидатів – 8 балів
- ефективно працює з супутниковими.
- можливий неправильний вибір кандидатів – 3 бали
- висока при розпізнаванні орієнтирів обличчя – 3 бали

Обмеження:

- наявні – 1 бал
- відсутні – 2 бали

На кількісних оцінках отримуємо корисність наших альтернатив.

Будемо вважати, що найкращими для вашого подальшого дослідження є ті альтернативи, які мають більшу корисність.

Пошук Парето-оптимального рішення, згідно з аксіомою Парето з двох векторних оцінок:

$$K(a) = (k_1(a), k_2(a), \dots, k_m(a)) \text{ і } K(b) = (k_1(b), k_2(b), \dots, k_m(b))$$

$K(a) P K(b)$ ($K(a) > K(b)$), якщо існує хоча б одне j від 1 до m таке що:

для будь-якого i не рівного j $k_i(a) \geq k_i(b)$ ($k_i(a)$ буде рівно $k_i(b)$), або $k_i(a) > k_i(b)$ ($k_i(a) > k_i(b)$), а $k_j(a) < k_j(b)$ ($k_j(a) < k_j(b)$).

Варіант a називається парето-оптимальним рішенням, якщо немає такого варіанту b , що $b > a$ по Парето.

Оптимум за Парето, також відомий як ефективність за Парето або оптимальність за Парето, — це концепція в економіці та теорії ігор, яка описує стан, у якому жоден індивід чи група не можуть бути покращені, не погіршивши становище інших.

Більш конкретно, оптимум Парето - це ситуація, в якій розподіл ресурсів є таким, що жодний подальший перерозподіл ресурсів не може покращити становище будь-якої особи, не погіршивши становище інших. Іншими словами, розподіл ресурсів є оптимальним у тому сенсі, що він максимізує загальний добробут, поважаючи індивідуальні переваги.

Оптимум за Парето часто використовується в контексті задач оптимізації з кількома цілями, де є кілька цілей, які потрібно оптимізувати одночасно. У таких проблемах може бути кілька оптимальних за Парето рішень, кожне з яких представляє компроміс між різними цілями.

Застосування теорії корисності: після перевірки за принципом Парето використаємо теорію корисності. Знайдемо таку згорткову модель $w(x)$, ($0 \leq w(x) \leq 1$), яка інтерпретується як корисність нашої альтернативи, а нашим прийнятним рішенням буде альтернатива з найбільшою корисністю.

Здійснимо нормування показників в таблиці з урахуванням \min і \max , введемо вагові коефіцієнти для наших критеріїв і обчислимо згорткову оптимізаційну модель Z^* , використовуючи лінійну адитивну згортку з ваговими коефіцієнтами.

Таблиця 3 – Результати оцінки

	Популярність	Якість архітектур	Швидкість виконання обчислень (кадри/с)	Точність виявлення об'єктів	Обмеження	Результат згортки з ваговими
YOLO v5	1	1	1	4/5	1	0,95
SSD	2/3	1/7	3/5	2/5	0	0,36
Keras retinaNet	2/3	2/7	3/5	1	1	0,63
Fast R-CNN	1/3	0	1 /5	0	0	0,08
Hog	0	0	0	0	0	0
Вагові коефіцієнти	0,1	0,33	0,25	0,25	0,07	

Альтернативою з найбільшою корисністю є «YOLO v5».

Отже, сфокусуємо дослідження на YOLO v5.

Модель YOLOv5 можна також модифікувати декількома способами, такими як розширення модуля часткової мережі (CSP), покращення ядра вибору (SK) у модулі уваги та модифікація механізму вилучення багатомасштабних ознак, який відіграє значну роль у виявленні та класифікації малих і великих розмірів поставлених лейблів на зображенні. Продуктивність моделі можна підтвердити завдяки середньомасштабному набору даних щодо виявлення лейблів (обличчя, біб-номер). Щоб перевірити ефективність моделі, треба порівняти результати запропонованої моделі з кількома варіантами моделі YOLOv5, де запропонована модель досягла найкращих результатів в експериментах. Таким чином, запропонована модель має потенціал для застосування в реальних додатках.

Концепція Парето-оптимуму названа на честь італійського економіста Вільфредо Парето, який вперше запропонував її наприкінці 19 століття як спосіб аналізу розподілу багатства та доходу в суспільстві. Відтоді ця концепція широко використовується в різних галузях, включаючи економіку, теорію ігор, інженерію та соціальні науки, як спосіб розуміння оптимального розподілу ресурсів і прийняття рішень.

Видалимо з нашого векторного опису неефективні альтернативи за Парето:

- альтернатива «Fast R-CNN» > Альтернативи «Hog»
- альтернатива «Keras retinaNet» > Альтернативи «Fast R-CNN»
- альтернатива «Keras retinaNet» > Альтернативи «SSD»

Таким чином, Парето-оптимальним рішенням для вибору моделей та алгоритмів анування зображень є 2 альтернативи:

- альтернатива «Keras retinaNet» та Альтернатива «YOLO v5»

2.5 Вибір алгоритму

Проаналізувавши на практиці які проблеми потрібно вирішувати для розпізнавання та знаходження обличчя, номерів мною було зроблено висновок, що для цієї задачі підходить саме імплементація через використання Tensorflow. Ефективна реалізація нейронних мереж: TensorFlow надає ефективну та просту у використанні структуру для побудови та навчання нейронних мереж, у тому числі згорткових нейронних мереж (CNN) [13], які зазвичай використовуються для розпізнавання зображень і виявлення об'єктів. TensorFlow надає доступ до попередньо навчених моделей, навчених на великих наборах даних, таких як ImageNet, які можна використовувати як відправну точку для побудови спеціальних моделей розпізнавання або виявлення.

В додаток до цього TensorFlow дозволяє створювати спеціальні моделі, які можна адаптувати до конкретних завдань розпізнавання або виявлення, що може бути важливим для досягнення високої точності та ефективності.

TensorFlow має велику та активну спільноту розробників, які роблять внесок у його розвиток і надають підтримку користувачам. Це означає, що доступно багато ресурсів, таких як навчальні посібники, документація та приклади коду, які можуть допомогти користувачам розпочати роботу з TensorFlow і вирішити будь-які проблеми, що виникають [5].

TensorFlow сумісний із низкою апаратних засобів, включаючи ЦП, графічні процесори та TPU, що дозволяє проводити ефективні обчислення та масштабувати моделі до більших наборів даних.

3 ЕКСПЕРИМЕНТ

3.1 Розробка вимог до системи та реалізація дослідження

Промоделювати та вирішити задачу за темою магістерської роботи «Дослідження моделей та алгоритмів машинного навчання для автоматичного анотування зображень спортивних заходів».

Аби навчити алгоритм класифікації, треба подати моделі позначені шаблони (розмітка на них має бути, звісно, можна, робити розмітку власноруч [6], але це дуже довго і краще взяти вже готові сети для цього). Задачі комп'ютерного розпізнавання полягають в класифікації та або можливості відрізнити зображення. Нейронні мережі можуть виконувати багато надскладних завдань обробки зображень. Одним із таких завдань є розпізнавання об'єктів, а також подальшої логіки розпізнавання саме конкретного об'єкта, унікального.

Система може бути виконана на будь-якій мові програмуванні, у будь-якій еко-системі. Важливим критерієм є дослідити ефективність використання декількох різних датасетів, що пов'язані зі спортом чи спортивними заходами, також порівняти декілька різних моделей для розпізнавання. Додаток має приймати на вхід зображення та на вихід видавати задетекчених людей чи обличчя в залежності від того, що треба відобразити.

Загалом завдання ділитися на дві частини – дослідницьке та практичне. З точки зору апаратного забезпечення: системі потрібен комп'ютер або сервер із достатньою обчислювальною потужністю для запуску алгоритмів розпізнавання зображень. Для швидкої обробки зображень доцільно використовувати потужний процесор або графічний процесор [11]. Камера (не входить в зону дослідження, але обов'язковий вхідний параметр): для зйомки учасників знадобиться камера високої роздільної здатності з хорошими характеристиками в умовах слабкого освітлення. Камеру слід розташувати у відповідному місці, щоб отримати чіткі зображення обличчя учасників. Програмне забезпечення для розпізнавання зображень: потрібна бібліотека програмного забезпечення для розпізнавання зображень, наприклад

OpenCV, TensorFlow або PyTorch. Ці бібліотеки надають попередньо підготовлені моделі або дозволяють розробляти власні моделі для розпізнавання зображень. Навчальні дані: для навчання моделі розпізнавання зображень знадобиться набір даних із зображеннями учасників марафону Python. Цей набір даних має включати різноманітний набір зображень із різними умовами освітлення, позами та виразами обличчя. Інструменти розробки програмного забезпечення: Python і пов'язані бібліотеки, такі як NumPy, SciPy і Matplotlib, знадобляться для впровадження програмного забезпечення для розпізнавання зображень. Крім того, середовище розробки програмного забезпечення, наприклад Jupyter Notebook або PyCharm, було б корисним для розробки та тестування коду. Інтерфейс користувача: для відображення зроблених зображень і розпізнавання учасників потрібен інтерфейс користувача. Це може бути простий графічний інтерфейс або веб-інтерфейс. Зберігання даних: для зберігання зображень і міток визнаних учасників потрібна база даних або файлова система. Це дозволить згодом аналізувати та відстежувати прогрес учасників. Мережа: якщо система розгорнута в мережі, мережева інфраструктура, така як комутатори, маршрутизатори та кабелі, може знадобитися для підтримки трафіку даних. Важливо зазначити, що розпізнавання обличчя є складним завданням, і на точність можуть впливати різні фактори, такі як умови освітлення, якість зображення та поза. Тому вкрай важливо ретельно оцінити продуктивність моделі та розглянути можливість використання додаткових методів, як-от вирівнювання обличчя та нормалізація, щоб покращити точність. Крім того, слід брати до уваги етичні міркування щодо розпізнавання обличчя і вживати належних заходів конфіденційності та безпеки для захисту особистої інформації учасників.

3.2 Обрання датасету, метрик, моделі, нейромережі

Щоб оцінити моделі, що класифікують, використовуємо такі метрики оцінки. Усі показники розглядаються у двійковій класифікації. Матриця плутанини — це таблиця, яка використовується для оцінки ефективності моделі машинного навчання для проблем класифікації. Це матриця фактичних і прогнозованих міток класів, яка допомагає візуалізувати кількість справжніх позитивних результатів, помилкових позитивних результатів, справжніх негативних результатів і помилкових негативних результатів [12], створених класифікатором. Матриця плутанини складається з чотирьох комірок або квадрантів наступним чином:

- справжній позитивний (TP): фактичний клас позитивний, і прогнозований клас також позитивний.
- хибнопозитивний (FP): фактичний клас негативний, але прогнозований клас позитивний.
- істинно негативний (TN): Фактичний клас негативний, і прогнозований клас також негативний.
- помилково негативний (FN): Фактичний клас позитивний, але прогнозований клас негативний.

Ось приклад матриці плутанини для проблеми бінарної класифікації, де фактичний клас є позитивним або негативним, а прогнозований клас моделі також позитивний або негативний: рядки матриці відповідають фактичним міткам класу, тоді як стовпці відповідають прогнозованим міткам класу. Загальна кількість екземплярів (вибірок) у наборі даних визначається сумою всіх клітинок у матриці.

		Predicted	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Рисунок 3.1 – Матриця плутанини

Матриця плутанини є корисним інструментом для оцінки ефективності

класифікатора, оскільки вона надає більш детальну інформацію про прогнози моделі, ніж просто дивитися на точність. Його можна використовувати для розрахунку різних показників оцінювання, таких як точність, запам'ятовування, оцінка F1 і точність. Наприклад, точність обчислюється як $TP/(TP+FP)$, а відкликання обчислюється як $TP/(TP+FN)$, де TP, FP і FN є значеннями в матриці плутанини.

Accuracy є одним із основних показників оцінки класифікації є точність. Він вимірює, як мітки людини були передбачені правильно. Тобто ставка TN плюс TP поділена на загальну суму кількість проб:

$$Accuracy = \frac{TN + TP}{TP + TN + FP + FN}.$$

Рисунок 3.2 – Формула Accuracy

Precision – це показник для оцінки точності позитивних прогнозів. Таким чином, кількість справжніх позитивних результатів ділиться на кількість усіх позитивних класифікованих проб.

$$Precision = \frac{TP}{TP + FP}.$$

Рисунок 3.3 – Формула Precision

Recall описує відсоток виявлених позитивних результатів для всіх фактично позитивних зразків. Отже, це також називають справжнім позитивним показником (TPR). TPR обчислюється шляхом ділення всіх позитивних класифікацій на загальну кількість фактично позитивних зразків:

$$Recall = \frac{TP}{TP + FN}$$

Рисунок 3.4 – Формула Recall

- вибір набору даних для розпізнавання біб-кодів залежатиме від конкретних вимог і характеристик вашого завдання розпізнавання. Ось кілька факторів, які слід враховувати під час вибору набору даних:
 - розмір даних: розмір набору даних може вплинути на точність моделі розпізнавання. Більший набір даних може допомогти покращити продуктивність моделі, особливо якщо вона включає різноманітний набір зображень.
 - якість зображення. Якість зображення в наборі даних має відповідати зображенням, з якими буде працювати модель розпізнавання. Якщо зображення біб-кодів, які потрібно розпізнати, зроблені за слабого освітлення, мають розмитість у русі або взяті з різних ракурсів, набір даних має містити зображення зі схожими характеристиками.
 - різноманітність біб-кодів: важливо переконатися, що набір даних містить широкий спектр біб-кодів, включаючи різні шрифти, розміри та орієнтації. Це допоможе гарантувати, що модель здатна точно розпізнавати біб-коди в різних налаштуваннях.
 - анотації: набір даних має бути позначений анотаціями, які вказують на розташування та вміст біб-кодів на зображеннях. Це дозволить моделі навчатися на основі даних із мітками та підвищити точність розпізнавання.
 - етичні міркування: вибираючи набір даних, враховуйте будь-які етичні чи правові міркування. Переконайтеся, що набір даних було зібрано з дотриманням етичних норм і отримано всі необхідні дозволи на його використання.
 - подібність до сценарію розгортання: якщо у вас є уявлення про типи біб-кодів, які ви розпізнаватимете, спробуйте використати набір даних із подібними біб-кодами до тих, з якими ви працюватимете. Це може допомогти переконатися, що модель навчається на зображеннях, які дуже схожі на зображення, з якими вона

зіткнеться в сценарії реального світу.

Деякі популярні набори даних для задач OCR включають набір даних MNIST, який містить велику кількість рукописних цифр, і набір даних COCO, який включає широкий спектр анотованих зображень. Однак можуть бути доступні більш спеціалізовані набори даних, спеціально розроблені для розпізнавання біб-кодів. Ви можете шукати існуючі набори даних в Інтернеті або створити власний набір даних, збираючи зображення біб-кодів і позначаючи їх вручну.

3.3 Опис експерименту

Python надає багато бібліотек і інструментів для реалізації цих кроків, включаючи TensorFlow, Keras, PyTorch, OpenCV і scikit-learn. Типовий експеримент може включати написання сценаріїв для завантаження та попередньої обробки даних, визначення архітектури моделі та циклу навчання та використання інструментів візуалізації для аналізу ефективності моделі.

Тестування та впровадження існуючих алгоритмів відбувається на існуючих датасетах. Використовуються переднавчені моделі, а не голі, бо їх навчання вимагає набагато більшої обчислювальної потужності та значно більшої кількості навчальних даних (зазвичай рекомендується більше 10 тисяч прикладів). Замість виявлення нових алгоритмів, віддається перевагу більш детальному дослідженню вже існуючим алгоритмам, що здатні знайти об'єкти на зображенні. Логічна складова програмного додатку ж дозволить далі співставляти об'єкти.

Програмний продукт також має містити скрипт, що надасть можливість подальшого моделей.

При подачі фотографії як вхідний параметр на вихід треба отримати фотографію, де у квадратику буде виділено обличчя, чи біб-код в залежності від того, що потрібно показати тільки потрібне у квадраті автоматично позначаючи решту зображення як фон (тобто всі інші типи об'єктів які не є потрібними і не були

окреслені як такі), ігноруються.

Експеримент із виявлення біб-кодів у зображеннях марафону за допомогою Python, включатиме наступні кроки:

- збір даних: збір набору даних зображень марафону з анотованими біб-кодами. Цей набір даних можна отримати з різних загальнодоступних баз даних або створити вручну, додавши до зображень коди bib.

- попередня обробка даних: попередня обробка набору даних для підготовки до навчання моделі. Це може передбачати зміну розміру зображень, нормалізацію значень пікселів і розширення набору даних за допомогою таких перетворень, як обертання, переклад і масштабування.

- вибір моделі: вибір відповідної архітектури моделі для завдання виявлення біб-кодів у марафонських зображеннях. Це може бути попередньо навчена модель виявлення об'єктів, наприклад Faster R-CNN, YOLO або RetinaNet.

Навчання моделі: навчання вибраної моделі на анотованому наборі даних. Це передбачає проходження зображень через модель і коригування параметрів моделі, щоб мінімізувати різницю між прогнозованими біб-кодами та основними мітками істинності.

- оцінка моделі: Оцінка навченої моделі на окремому тестовому наборі даних для вимірювання її ефективності з точки зору точності, точності, запам'ятовування та оцінки F1.

- розгортання моделі: розгортання навченої моделі для виявлення біб-кодів у нових зображеннях марафону. Це може передбачати інтеграцію моделі з веб-або мобільним додатком, щоб користувачі могли завантажувати зображення та отримувати прогнози в реальному часі.

Важливо зазначити, що виявлення та розпізнавання облич є складними завданнями, і на точність можуть впливати різні фактори, як-от умови освітлення, якість зображення та поза. Тому вкрай важливо ретельно оцінити продуктивність моделі та розглянути можливість використання додаткових методів, як-от вирівнювання обличчя та нормалізація, щоб покращити точність. Експеримент із

виявлення обличчя на зображеннях марафону за допомогою Python, включатиме наступні кроки:

- збір даних: збір набору даних зображень марафону з анотованими обличчями. Цей набір даних можна отримати з різних загальнодоступних баз даних або створити вручну, додавши анотації до зображень за допомогою обмежувальних рамок навколо облич.

- попередня обробка даних: попередня обробка набору даних для підготовки до навчання моделі. Це може передбачати зміну розміру зображень, нормалізацію значень пікселів і розширення набору даних за допомогою таких перетворень, як обертання, переклад і масштабування.

- вибір моделі: вибір відповідної архітектури моделі для завдання виявлення обличчя на марафонських зображеннях. Це може бути попередньо навчена модель розпізнавання обличчя, як-от каскади Хаара, гістограма орієнтованих градієнтів (HOG) або модель на основі глибокого навчання, як-от Single Shot MultiBox Detector (SSD) або You Look Only Once (YOLO).

- навчання моделі: навчання вибраної моделі на анотованому наборі даних. Це передбачає проходження зображень через модель і коригування параметрів моделі, щоб мінімізувати різницю між прогнозованими обличчями та базовими мітками істинності.

- оцінка моделі: Оцінка навченої моделі на окремому тестовому наборі даних для вимірювання її ефективності з точки зору точності, точності, запам'ятовування та оцінки F1.

- розгортання моделі: розгортання навченої моделі для виявлення обличчя на нових зображеннях марафону. Це може передбачати інтеграцію моделі з веб-або мобільним додатком, щоб користувачі могли завантажувати зображення та отримувати прогнози в реальному часі.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ ПРОТОТИПУ

4.1 Проектування прототипу

TensorFlow — це бібліотека машинного навчання, спочатку розроблена Google і доступна як для Python, так і для JavaScript.

TensorFlow для Python — це найпоширеніша версія TensorFlow, яка надає широкий спектр інструментів машинного навчання та алгоритмів для побудови та навчання моделей. Версія TensorFlow на Python має більшу спільноту розробників і користувачів, а це означає, що вона має більше ресурсів, навчальних посібників і документації, доступних онлайн.

З іншого боку, TensorFlow.js — це JavaScript-версія TensorFlow, яка дозволяє розробникам запускати моделі машинного навчання безпосередньо в браузері або на Node.js. Це спрощує інтеграцію функцій машинного навчання у веб-програми без необхідності окремого внутрішнього сервера.

TensorFlow.js також надає низку попередньо навчених моделей, які можна використовувати з коробки для таких завдань, як виявлення об'єктів, класифікація зображень і генерація тексту. Крім того, TensorFlow.js можна використовувати в поєднанні з іншими популярними фреймворками веб-розробки, такими як React і Vue.js.

Обидві версії TensorFlow мають свої сильні та слабкі сторони, і вибір версії для використання залежить від конкретних вимог проекту та знання мови розробником. Якщо проект передбачає складніші завдання машинного навчання, Python може бути кращим вибором, а якщо проект передбачає інтеграцію машинного навчання у веб-додаток, TensorFlow.js може бути більш придатним.

Розглянемо додаток TensorFlow (TensorFlow.js) – версію для JavaScript, Terran.io – версію для Python та розширення для .Net.

TensorFlow.js Object Detection



Рисунок 4.1 – Детект людей за допомоги TensorFlow.js (1)

Порівняємо результат виконання детекту людей різними бібліотеками. На Рисунках 4.1-4.3 представлена вхідна фотографія та результат завдяки Tensorflow.js. Результат є доволі вдалим та гарним, мережа добре навчена. Скрипт є доволі простим та наданий на офіційному сайті додатку.



Рисунок 4.2 – Детект людей за допомоги TensorFlow.js (2)

На Рисунку 4.3 задетекчені люди з Харківського марафону 2021 року.



Рисунок 4.3 – Детект людей за допомоги TensorFlow.js (3)

Спробуємо задетектити людські обличчя за допомоги Terran.io.

На Рисунках 4.4-4.7 надані результати роботи Terran.

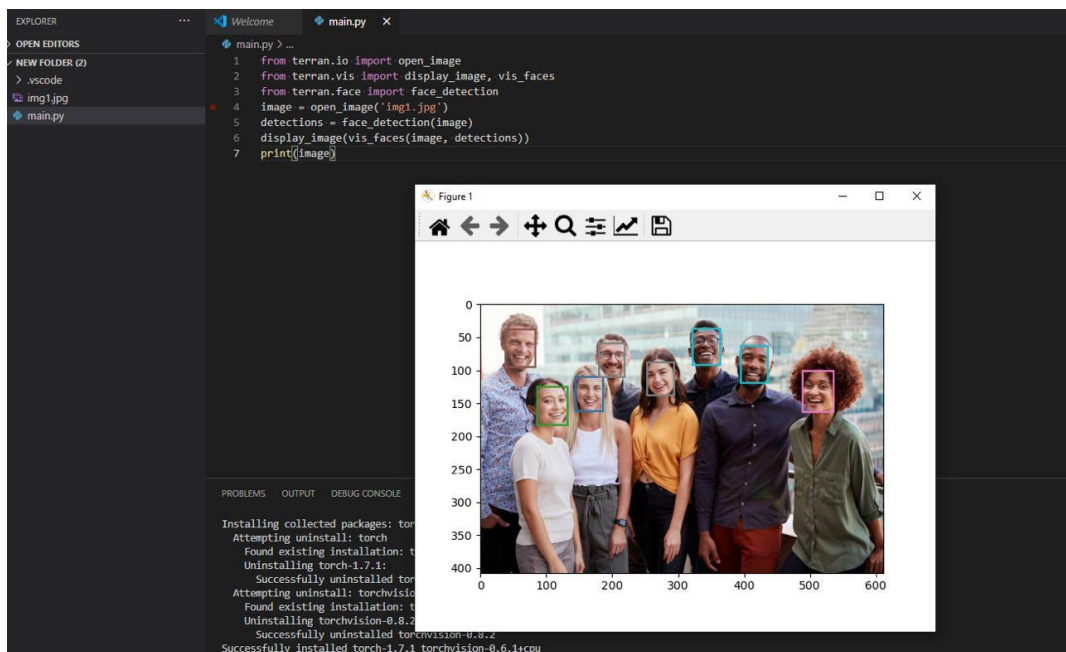


Рисунок 4.4 – Детект людей за допомоги Python, Terran library (1)

Бібліотека є доволі простою у використанні, але не має підтримки у спільноті, також оновлень немає з 2020 року.

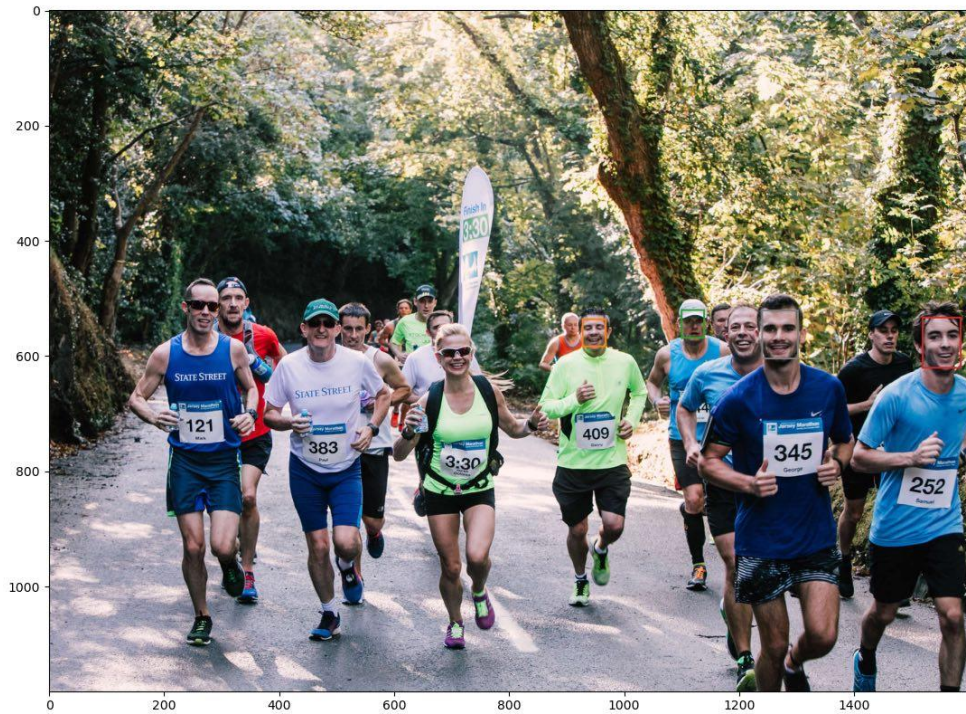


Рисунок 4.5 – Детект людей за допомоги Python, Terran library (2)

Варто зазначити, що бібліотека Терран доволі погано детектить обличчя здалеку. Тобто, якщо чисто повірізати зображення людей, яких виявили tensorflow додатком, наприклад, і додати кожен таку фотографію в окрему папку, а потім серед них перевіряти чи підходить обличчя тесерактом (ще один інструментарій, який працює в середовищі Python), то можна отримати доволі гарні результати, але знову ж таки, якщо у Terran на вхід дати більш наближене зображення, то він чудово розпізнає всіх. Тесеракт у себе всередині використовує openCV, так само як і більшість бібліотек для детекту чи розпізнавання, але деякі додатки мають більш навчені моделі.



Рисунок 4.6 – Детект людей за допомоги Python, Terran library (3)

У чистому вигляді `openCV` не рекомендовано використовувати, зручніше буде користуватися цим інструментарієм для усунення шумів на фотографії. Тесеракт у вигляді обгортки над `openCV` дозволить швидко та комфортно підготувати зображення: препроцесинг включає роботу з блюром, шумами, похибками, зміна розміру.

Розпізнавання числових значень у тесеракті буде гірше, бо дуже багато похибок, нюансів та є складнощі в налаштуванні `Page Segmentation Mode (--psm)` і `engine mode (--oem)`. Якщо підбирати параметри, то можна таки дійти до гарного результату, але мінус у затратності часового аспекту на тестування своїх припущень.

У літерах все дуже залежить від шрифту.

Фотографії завжди у сховищах (наприклад, хмарних) будуть зберігатися як медіа чи `blob`-об'єкт, а оброблюватися у програмі будуть як масив байтів.

ВИСНОВКИ

Робота була присвячена детальному дослідженню на тему «Дослідження моделей та алгоритмів машинного навчання для автоматичного анотування зображень спортивних заходів».

Розглянуто такі моделі, як YoloV5, SSD, kerasRetina Net, Fast R-CNN, Hog. В ході дослідження за рядом метрик та параметрів дослідницьким шляхом було виявлено, що найоптимальніше використовувати мережу YoloV5. Модель має вже навчені мережі, але є можливість їх довчити або ж перенавчити. У спільності Yolo будь-якої версії користується затребуваністю та популярністю, має багато варіацій та потужний функціонал. Загалом ця бібліотека активно розвивається, підтримується та має великий потенціал. Таке рішення доволі доречно використовувати у роботі, бо з плином часу рішення буде легко масштабуватися. Дослідження було проведено за використання Парето принципу, де для визначення найкращої альтернативи були обрані метрики та оцінені, а потім шляхом відсікання була обрана найліпша.

Інша частина дослідження була направлена на дослідження практичного інструментарію для детекту та розпізнавання обличчя чи людей (будь-яка класифікація). Експеримент було поставлено на тестових, навчальних та реальних даних, по 10 фото на кожний кейс. Використаний інструментарій: Terran.io (Python), Tesseract (Python), TensorFlow (TensorFlow.js version, JavaScript), завдяки мові програмування C# та безпосередньо платформи .Net були розроблені маленькі програми для демонстрації повноцінної роботи за схемою: завантаження зображення до системи, отримання вихідного результату з задетекченими обличчями. Є можливість знайти своє фото по своєму обличчю.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. X. Bai, L. Wang, Y. Hu, P. Li and Y. Zu, "Optimal Path Planning Method for IMU System-Level Calibration Based on Improved Dijkstra's Algorithm," in IEEE Access, vol. 11, pp. 11364-11376, 2023, doi: 10.1109/ACCESS.2023.3240518.
2. M. C. Brindise, B. A. Meyers, S. Kutty and P. P. Vlachos, "Automated Peak Prominence-Based Iterative Dijkstra's Algorithm for Segmentation of B-Mode Echocardiograms," in IEEE Transactions on Biomedical Engineering, vol. 69, no. 5, pp. 1595-1607, May 2022, doi: 10.1109/TBME.2021.3123612.
3. Орельен Жерон(2017). Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems с. 99-112, vol. 64, no. 3, pp. 856-864, June 2022, doi: 10.1109/TEMC.2021.3131492.
4. Distributed Deep Learning Using Synchronous Stochastic Gradient Descent [Електронний ресурс] / D. Das,S. Avancha, D. Mudigere, K. Vaidynathan, S. Sridharan, D. Kalamkar, B. Kaul, P. Dubey. – Режим доступу: <https://arxiv.org/pdf/1602.06709.pdf>. – Назва з титул. екрану.
5. D. Lopez-Pajares, E. Rojas, J. A. Carral, I. Martinez-Yelmo and J. Alvarez-Horcajo, "The Disjoint Multipath Challenge: Multiple Disjoint Paths Guaranteeing Scalability," in IEEE Access, vol. 9, pp. 74422-74436, 2021, doi: 10.1109/ACCESS.2021.3080931.
6. Statistical Validation of Image Segmentation Quality Based on a Spatial Overlap Index / H. Zou, S.K. Warfield,A. Bharatha, C.M. Tempany, M.R. Kaus, S.J. Haker, W.M. Wells, F.A. Jolesz, R. Kikinis. [https://doi.org/10.1016/S1076-6332\(03\)00671-8](https://doi.org/10.1016/S1076-6332(03)00671-8).
7. Y. Yu, O. S. Nduka and B. C. Pal, "Smart Control of an Electric Vehicle for Ancillary Service in DC Microgrid," in IEEE Access, vol. 8, pp. 197222-197235, 2020, doi: 10.1109/ACCESS.2020.3034496.

8. Y. Meng, Y. Wu, Q. Gu and L. Liu, "A Decoupled Trajectory Planning Framework Based on the Integration of Lattice Searching and Convex Optimization," in *IEEE Access*, vol. 7, pp. 130530-130551, 2019, doi: 10.1109/ACCESS.2019.2940271.
9. J. Han, D. Shen, D. Karbowski and A. Rousseau, "Leveraging Multiple Connected Traffic Light Signals in an Energy-Efficient Speed Planner," in *IEEE Control Systems Letters*, vol. 5, no. 6, pp. 2078-2083, Dec. 2021, doi: 10.1109/LCSYS.2020.3047605.
10. B. Confais, B. Parrein and A. Lebre, "Data Location Management Protocol for Object Stores in a Fog Computing Infrastructure," in *IEEE Transactions on Network and Service Management*, vol. 16, no. 4, pp. 1624-1637, Dec. 2019, doi: 10.1109/TNSM.2019.2929823.
11. F. Zhang, X. Zhou and M. Sun, "On-Demand Receiver-Centric Channel Allocation via Constrained VCG Auction for Spatial Spectrum Reuse," in *IEEE Systems Journal*, vol. 13, no. 3, pp. 2519-2530, Sept. 2019, doi: 10.1109/JSYST.2019.2912757.
12. T. Kulvicius, S. Herzog, M. Tamosiunaite and F. Wörgötter, "Finding Optimal Paths Using Networks Without Learning—Unifying Classical Approaches," in *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 12, pp. 7877-7887, Dec. 2022, doi: 10.1109/TNNLS.2021.3089023.
13. Darken C. Learning rate schedules for faster stochastic gradient search / C. Darken, J. Chang, J. Moody // *Neural Networks for Signal Processing II Proceedings of the 1992 IEEE Workshop*, 1992. September. – p. 1-11. <https://doi.org/10.1109/NNSP.1992.253713>.
14. K. Smelyakov, A. Chupryna, M. Hvozdiev and D. Sandrkin, "Gradational Correction Models Efficiency Analysis of Low-Light Digital Image," 2019 Open Conference of Electrical, Electronic and Information Sciences (eStream), 2019, pp. 1-6, doi: 10.1109/eStream.2019.8732174.
15. K. Smelyakov, A. Datsenko, V. Skrypka and A. Akhundov, "The Efficiency of Images Reduction Algorithms with Small-Sized and Linear Details," 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications,

Science and Technology (PIC S&T), 2019, pp. 745-750, doi: 10.1109/PICST47496.2019.9061250.

16. K. Smelyakov, A. Chupryna, O. Bohomolov and N. Hunko, "The Neural Network Models Effectiveness for Face Detection and Face Recognition," 2021 IEEE Open Conference of Electrical, Electronic and Information Sciences (eStream), 2021, pp. 1-7, doi: 10.1109/eStream53087.2021.9431476.
17. Open Sourcing TensorFlowOnSpark: Distributed Deep Learning on Big-Data Clusters [Електронний ресурс] / L. Yang, J. Shi, B. Chern, A. Feng. – Режим доступу: <https://developer.yahoo.com/blogs/157196317141/>. – Назва з титул. екрану. (дата звернення: 18.04.2023).