

Міністерство освіти і науки України  
Харківський національний університет радіоелектроніки  
Факультет Комп'ютерних наук  
Кафедра Програмної інженерії

**АТЕСТАЦІЙНА РОБОТА**  
**Пояснювальна записка**  
рівень вищої освіти – другий (магістерський)

Дослідження методів розпізнавання зображень для розробки програмної системи  
визначення тіла людини з різних ракурсів

Виконав: студент 2 курсу, групи ІІЗм-18-4

Білецький І.С.

спеціальності 121 – Інженерія програмного забезпечення

Освітньо-наукової програми  
Інженерія програмного забезпечення

Керівник проф. каф. ПІ Білоус Н.В.

Допускається до захисту

Зав. кафедри, проф. \_\_\_\_\_

З.В. Дудар

2020 р

## ХАРКІВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ РАДІОЕЛЕКТРОНІКИ

Факультет Комп'ютерних наукКафедра Програмної інженерії

Рівень вищої освіти - другий (магістерський)

Спеціальність 121-Інженерія програмного забезпечення

(код і повна назва)

Тип програми освітньо-наукова програмаОсвітня програма Інженерія програмного забезпечення

ЗАТВЕРДЖУЮ:

Зав. кафедри \_\_\_\_\_

(підпис)

« \_\_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ р.

### ЗАВДАННЯ НА АТЕСТАЦІЙНУ РОБОТУ

студентові Білецькому Ігорю Сергійовичу

(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження методів розпізнавання зображень для розробки програмної системи визначення тіла людини з різних ракурсів

затверджена наказом університету від "27" березня 2020 р № 473 Ст

2. Термін подання студентом роботи до екзаменаційної комісії  
20 травня 2020 р.

3. Вихідні дані до роботи методи розпізнавання зображень, методи визначення положення тіла та пози, пояснювальна записка, середовище об'єктно-орієнтованного проектування.

4. Перелік питань, що потрібно опрацювати в роботі дослідити методи розпізнавання визначення положення тіла з різних ракурсів, проаналізувати підходи до аналізу знаходження ракурсу, обґрунтувати вибір підходу та методу для визначення ракурсу

## 5. Консультанти розділів роботи

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розд	
		підпис	дат
Спецчастина	Проф. Білоус Н.В..		18.05.2020

**КАЛЕНДАРНИЙ ПЛАН**

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка*
1	Аналіз предметної галузі	10 квітня 2020 р.	виконано
2	Огляд варіантів систем, що дозволяють визначити тіло людини	15 квітня 2020 р.	виконано
3	Огляд методів та бібліотек, що визначають положення тіла людини з різних ракурсів	28 квітня 2020 р.	виконано
4	Підготовка пояснювальної записки	10 травня 2020 р.	виконано
5	Спецчастина	12 травня 2020 р.	виконано
6	Підготовка презентації та доповіді	14 травня 2020 р.	виконано
7	Попередній захист	15 травня 2020 р.	виконано
8	Нормоконтроль, рецензування	16 травня 2020 р.	виконано
9	Занесення диплома в електронний архів	17 травня 2020 р.	виконано
10	Допуск до захисту у зав. кафедри	25 травня 2019р.	

Дата видачі завдання \_\_\_\_\_ 2020 р.

Студент \_\_\_\_\_ Білецький І.С.  
(підпис)Керівник роботи \_\_\_\_\_ проф. каф.ІІІ Білоус Н.В.  
(підпис) (посада, прізвище, ініціали)

## РЕФЕРАТ / ABSTRACT

Атестаційна робота магістра містить 80 с., 29 рис., 3 табл., 30 джер.

РОЗПІЗНАВАННЯ ПОЗИ ЛЮДИНИ, ЗГОРТКОВІ НЕЙРОННІ МЕРЕЖІ, API, KINECT, OPENPOSE, ЗОБРАЖЕННЯ ГЛИБИНИ, ДАНІ СКЕЛЕТА.

Об'єкт дослідження – процес розпізнавання тіла та пози людини. Предметом дослідження – методи розпізнавання рухів людини за допомогою ключових точок.

Мета роботи – дослідження методів розпізнавання пози людини для розробки програмної системи визначення його тіла з різних ракурсів. Методи рішення базуються на технологіях OpenPose, OpenCV з використанням мов програмування C# та Python. В результаті розробки створено систему, що дозволяє за допомогою стаціонарних камер або камери мобільного телефону визначати тіло людини з різних ракурсів. Також був проведений аналіз існуючих методів обробки і класифікації відеопотоків і розпізнавання пози людини у вигляду 2D скелету, запропоновано гібридний підхід, який використовує Kinect, OpenPose та згорткову нейронну мережу для оцінки пози людини.

HUMAN POSITION RECOGNITION, CONVOLUTIONAL NEURAL NETWORK, API, KINECT, OPENPOSE, DEPTH IMAGE, SKELETON DATA, PYTHON, C#.

The object of research is the process of recognizing the human body and posture. The subject of research are methods of recognizing human movements using key points.

The purpose of the work is to study the methods of recognizing a person's posture to develop a software system for determining his body from different angles. The solution method is based on OpenPose, OpenCV technologies using C # and Python programming languages. As a result of the development, a system was created that allows using fixed cameras or mobile phone cameras to determine the human body from different angles. An analysis of existing methods for processing and classifying video streams and recognizing human posture as a 2D skeleton was also performed, and a hybrid approach was proposed that uses Kinect, OpenPose, and a convolutional neural network to assess human posture.

## ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ .....	6
ВСТУП .....	7
1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ .....	9
1.1. Аналіз предметної галузі.....	9
1.2 Аналіз рішень щодо розпізнавання тіла та пози людини .....	10
2.1 Огляд методів розпізнавання пози людини .....	14
2.1.1 OpenPose .....	14
2.1.2 DeepCut .....	15
2.1.3 Hierarchical RNN for Skeleton Based Action Detection .....	16
2.1.4. Regional multi-person pose estimation(RMPE або AlphaPose) .....	17
2.1.5 R-CNN (Regions With CNNs) .....	18
2.1.6 Mask R-CNN .....	20
2.2. Огляд існуючих рішень для реалізації проекту .....	21
3 ВИЗНАЧЕННЯ ПОЛОЖЕННЯ ТІЛА З ВИКОРИСТАННЯМ КІНЕСТ ...	25
3.1 Створення центру ваги тіла, контуру та висоти тіла. ....	25
3.2 Локалізація голови .....	27
3.3 Навчання BPNN .....	30
4 ВИЗНАЧЕННЯ ТІЛА ТА ПОЗИ ЛЮДИНИ З ВИКОРИСТАННЯМ АЛГОРИТМУ OPENPOSE.....	32
5 ПРОГРАМНА РЕАЛІЗАЦІЯ. ЕКСПЕРИМЕНТАЛЬНІ РЕЗУЛЬТАТИ ...	40
ВИСНОВКИ.....	54
ДОДАТОК А Фрагменти похідного коду програми .....	58
ДОДАТОК Б Слайди презентації .....	74

## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

КІЧ – Карти Інтенсивності частин

ПАЧ – Поля Інтенсивності Частин

AP – Average Precision

AR – Average Recall

BPNN – Backpropagation Neural Network

BRNN – Bidirectional Recurrent Neural Network

CNN – Convolutional Neural Network

COCO – Common Objects in Context

NMS – Non-Maximum Suppression

R-CNN – Recurrent Convolutional Neural Networks

ReLU – Rectified Linear Unit

ResNet – Residual Network

RMPE – Regional multi-person pose estimation

SSTN – Symmetric Spatial Transformer Network

## ВСТУП

Здоров'я - безцінне надбання не тільки кожної людини, але і всього суспільства. Гарне здоров'я, розумно зберігається і зміцнюється самою людиною, забезпечує йому довгу і активне життя. Наукові дані свідчать про те, що у більшості людей при дотриманні ними гігієнічних правил є можливість жити до 100 років і більше. На жаль, багато людей не дотримуються найпростіших, обґрунтованих наукою норм здорового способу життя. Одні стають жертвами малорухливості (гіподинамії), що викликає передчасне старіння, інші надмірності в їжі з майже неминучим в цих випадках розвитком ожиріння, склерозу судин, а у деяких - цукрового діабету, треті не вміють відпочивати, відволікатися від виробничих і побутових турбот, вічно неспокійні, нервові, страждають безсонням, що в кінцевому підсумку призводить до численних захворювань внутрішніх органів.

Медичною наукою встановлено, що систематичні заняття фізичною культурою, дотримання правильного рухового і гігієнічного режиму є потужним засобом попередження багатьох захворювань, підтримання нормального рівня діяльності і працездатності організму. Формування людини на всіх етапах його еволюційного розвитку проходило в нерозривному зв'язку з активною м'язовою діяльністю. Організм людини розвивається в постійному русі. При дефіциті рухової активності знижується стійкість організму до застуди і дії хвороботворних мікроорганізмів.

Існує безліч систем тренінгу, але в принципі, не існує єдиного типу тренінгу для всіх - для кожного потрібен індивідуальний підхід. Єдиний тип тренінгу, який враховує всі ваші особливості фігури, який в залежності від вашого типу запропонує вам комплекс вправ, що включає в себе заняття з вагами і аеробіку - це фітнес. поняття. Фітнес» вже міцно увійшло в наше життя, створені фітнес-клуби, видаються журнали, проводяться фітнес турніри. Фітнес став способом життя, який веде до фізичного і ментального здоров'я людини. Однак, важливо правильне виконання всіх фізичних вправ.

Для цього потрібен або контроль тренера, професіонала, або самоконтроль за допомогою інтелектуального додатку, який перевіряє правильність виконання, порівнюючи з шаблоном вправи.

Актуальним на даний момент є визначення правильності виконання фізичних вправ та рухів людиною для подальшого порівняння з правильним шаблоном їх виконання. Тому метою атестаційної роботи є дослідження та удосконалення методу розпізнавання тіла та пози людини за допомогою ключових точок для розробки додатку, що дозволить аналізувати та відстежувати рухи людини з різних ракурсів.

Об'єктом дослідження є процес розпізнавання тіла та пози людини з різних ракурсів на зображенні або відеопотоці. Предметом дослідження є методи розпізнавання рухів людини за допомогою ключових точок.

Метою дослідження є удосконалення методу розпізнавання тіла людини через його позу з різних ракурсів. Для досягнення поставленої мети необхідно виконати наступні завдання:

- провести аналіз існуючих підходів, методів алгоритмів та програмних аналогів у предметній області;
- розробити систему розпізнавання тіла людини через визначення пози з різних ракурсів;
- провести дослідження ефективності розроблених алгоритмів та порівняти із існуючими аналогами на обраному наборі даних.

# 1 АНАЛІЗ ПРЕДМЕТНОЇ ГАЛУЗІ ТА ПОСТАНОВКА ЗАДАЧІ

## 1.1. Аналіз предметної галузі

Стрімкий розвиток сучасних комп'ютерних технологій призвів до появи потужних і доступних обчислювальних пристроїв, що дозволяють реалізовувати і виконувати складні алгоритми, що вимагають значних обчислювальних потужностей. Завдяки цьому стала можлива практична реалізація різних методів штучного інтелекту, в тому числі – ефективних методів машинного навчання, що дозволяють замінити людину в різних сферах діяльності, що вимагають тривалої монотонної обробки інформації, допомагають визначати фізичний стан, скоректувати виконання різних фізичних вправ при заняттях спортом, фітнесом та при реабілітації захворювань.

У сучасному світі, з урахуванням зростання безпілотних технологій, технологій віртуальної реальності і доповненої реальності, виникає питання про визначення стану речей у просторі, а саме про визначення положення тіла людини. Можливість визначати позу людини на зображенні чи відео у зазначених областях відіграє ключову роль.

Значні успіхи були досягнені у розпізнаванні пози людини завдяки використанню нейронних мереж. Одна з найпоширеніших моделей – перцептрон. Однак для вирішення задачі розпізнавання пози людини дана модель не підходить, адже великий розмір вхідних даних (зображення, відео) призводить до значного збільшення кількості синаптичних зв'язків, нейронів у мережі. Як результат, швидкість та обчислювальна складність розпізнавання значно збільшується. Топологія вхідних даних ігнорується у цій моделі нейронної мережі, чітка двомірна структура вхідних зображень не враховується. Для усунення цих недоліків, ми будемо використовувати згорткову нейронну мережу.

Існуючі моделі та методи розпізнавання тіла людини через знаходження пози на відео є достатньо складними та займають багато часу при оцінці пози у режимі реального часу. Наразі активно ведеться розробка спрощених алгоритмів, які

зможуть досить точно та швидко розпізнати позу людини у відеопотоці, що дозволить їх використовувати у автопілотних автомобілях у режимі реального часу, в різноманітних фітнес-додатках, при реабілітації пацієнтів при виконанні вправ лікувальної фізкультури тощо. Дана робота присвячена поліпшенню існуючих алгоритмів оцінки пози людини у переповнених непланованих сценах.

## 1.2 Аналіз рішень щодо розпізнавання тіла та пози людини

Основні рішення, що існують на ринку:

– Microsoft Kinect, розробники нової консолі створили велику кількість фітнес-ігор, які перетворюють домашні завдання спортом в реальні персональні тренування. У той час як користувач рухається перед екраном, Kinect зчитує інформацію, а консоль обробляє її, витрачаючи на це близько 10-15% процесорної потужності. Які ж дані передаються з бездротового контролера на консоль? Перш за все, кольорове (RGB) потокове відео з частотою 30 кадрів в секунду і роздільною здатністю VGA (640x480 пікселів). Потім – з такою ж здатністю монохромний відеопотік, «стежить» за глибиною зображення. І, на завершення, 16-бітове аудіо з частотою дискретизації 16 кГц. Користувач бачить його аватар, копію себе, який повторює за ним всі рухи. Для того, щоб визначати положення гравця, використовується приблизно та ж технологія, що і в далекомірах. Інфрачервоний проектор накладає на простір перед Kinect сітку з точок. 30 раз в секунду ІК-камера зчитує картинку і передає дані на консоль, де для кожної з цих точок визначається відстань від неї до камери;

– VirtualRehab. На сьогоднішній день VirtualRehab - це одне з найвидатніших відкриттів в області медичної реабілітації. VirtualRehab використовує технологію розпізнавання рухів на базі сенсора Microsoft® Kinect, що перетворює користувача в головну дійову особу всього процесу медичної

реабілітації. Кожен сеанс реабілітації реєструється завдяки технології Microsoft® Azure, яка зберігає інформацію в базі даних на хмарі, що дозволяє пацієнтові виконувати вправи в медичному закладі або вдома;

– SmartSpot. це розумне дзеркало з дисплеєм і контролером Microsoft Kinect всередині, яке дає зворотний зв'язок по правильності виконання вправ в процесі тренування. Відео та фотографії тренування потім завантажуються в акаунт користувача.

Крім ігрових підходів існують підходи, за допомогою смарт-годинників та трекерів. Такі пристрої являють собою сукупність кількох дуже складних і точних датчиків, які відстежують безліч параметрів руху, швидкості і положення в просторі. Вони містять і акселерометри, і гіроскопи, і магнітометри - вони збирають дані, які завдяки фірмовому алгоритму перетворюються в корисну інформацію. Наприклад, трекер компанії Moov не тільки відстежує процес виконання вправ, що входять до складу тренування, але і контролює правильність їх виконання, а також дає поради з приводу того, як правильно робити вправи. Ключовою відмінністю Moov Now від інших пристроїв є зворотній зв'язок. Під час тренування Moov буде аналізувати дії користувача, порівнюючи їх «еталонної» моделлю (яка записана при виконанні вправ професійними спортсменами) і буде повідомляти йому про помилку, даючи поради, про те як їх уникнути. При тренуваннях Moov Now може використовувати дані GPS з смартфона користувача.

Ще одне рішення, яке дозволяє контролювати ефект від виконання різних вправ – це «вумний» одяг. В конструкції «вумних» футболки вбудовані сенсори, які контролюють роботу м'язів, відстежують серцевий ритм та частоту дихання. Крім накопичення інформації з сенсорів є позиційний акселерометр та вбудований модуль Bluetooth, який дозволяє передавати всі отримані дані на мобільний пристрій зі спеціальним програмним забезпеченням.

Один із варіантів вирішення завдання розпізнавання пози людини був представлений ще в 1973 році - Pictorial Structures Model [1]. Основою для даного фреймворка виступала ідея про створення деякої математичної моделі і скелета людини у вигляді графа, де кістки - ребра, а суглоби - вершини даного графа.

Протягом останніх років найсучасніші методи оцінки пози ґрунтуються на конволюційних нейронних мережах. Вони перевершують традиційні методи, засновані на зображувальних структурах та моделях деформованих частин. Популярність глибокого навчання розпочалося з DeepPose [2], який використовує каскад згорткових мереж для оцінки пози цілого тіла. Тоді, замість прогнозування абсолютних місцеположень людських суглобів, деякі автори вдосконалюють оцінку пози, передбачуючи виправлення помилок при кожній ітерації або використовуючи мережу уточнення поз людини для використання залежностей між вхідними та вихідними просторами.

Усі ці підходи до оцінки пози людини можна згрупувати в методи «знизу вгору» та «згори вниз». Перший оцінює спочатку кожен суглоб тіла, а потім групує їх для створення унікальної пози. Останній запускає спочатку детектор людини і оцінює суглоби тіла в межах виявлених обмежувальних діапазонів. Методи «зверху вниз» особливо ефективні, коли пішоходи перекривають інших пішоходів, де стикаються обмежувальні діапазони. Попередні методи «знизу вгору» не обмежують поле, але все ще містять грубу карту функцій для локалізації. Запропонований метод не містить будь-яких обмежень на основі сітки щодо просторової локалізації суглобів і має здатність оцінювати численні пози, що перекриваються один одним.

### 1.3 Постановка задачі дослідження

Проведений аналіз рішень визначення тіла людини допоміг описати низку вимог до дослідження та побудови системи, яка дозволить через використання стаціонарних камер, камер мобільних пристроїв та Kinect виділяти ключові точки скелету людини. Призначенням розробки є система для розпізнавання пози людини у режимі реального часу. Система повинна використовувати Kinect SDK 2.0 для

отримання глибинного зображення та скелетних даних. Розпізнаватися буде шість різних поз:

- стояча;
- сидяча;
- сидяча з схрещеними ногами;
- стояча на колінах;
- лежача;
- згинаючись.

Основними цілями створення комплексу задач є:

- підвищення швидкості та точності розпізнавання пози людини.
- Для досягнення поставлених цілей потрібно вирішити такі завдання:
  - провести аналіз існуючих алгоритмів та бібліотек у предметній області;
  - розробити систему розпізнавання пози людини у режимі реального часу;
  - розробити серверну частину та адміністративну панель системи, за допомогою яких є можливість проводити визначення тіла людини з різних ракурсів;
  - розробити API для мобільного додатку клієнтів камери та полегшення використання бібліотеки OpenPose;
  - провести дослідження ефективності розроблених алгоритмів.

## 2 СУЧАСНИЙ СТАН РОЗПІЗНАВАННЯ ПОЗИ ЛЮДИНИ ЗА ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ

### 2.1 Огляд методів розпізнавання пози людини

Детальніше розглянемо існуючі методи розпізнавання пози людини.

#### 2.1.1 OpenPose

OpenPose [3] – це один із найпопулярніших підходів «знизу вгору» для оцінки пози багатьох людей у кадрі. Як і у більшості підходах знизу вгору, OpenPose спочатку виявляє частини(ключові точки), що належать кожній людині на зображенні, а потім призначає частини окремим особам. На рисунку 2.1 показана архітектура моделі OpenPose.

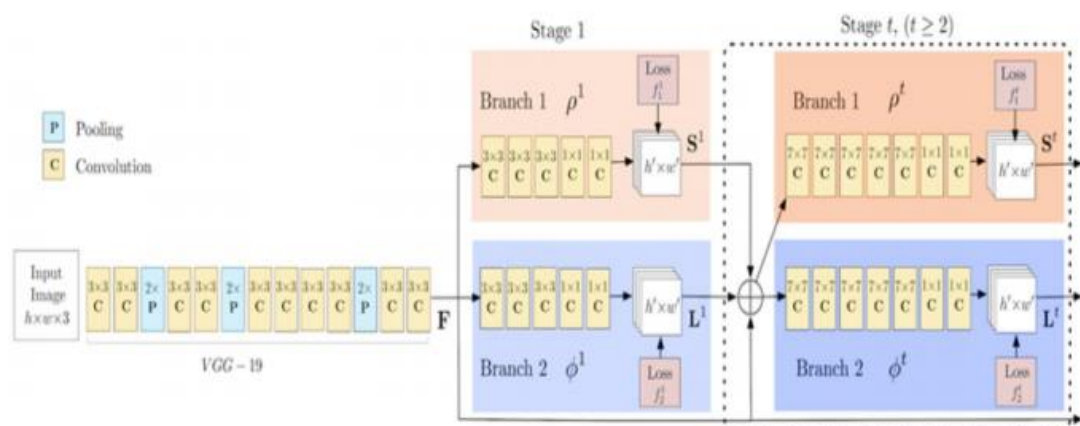


Рисунок 2.1 - Архітектура моделі OpenPose[3]

Мережа OpenPose спочатку витягує функції з зображення за допомогою перших кількох шарів(VGG-19 у вищевказаній блок-схемі). Потім риси подаються у дві паралельні гілки згорткових шарів. Перша гілка прогнозує набір з 18 карт

впевненості, при цьому кожна карта представляє певну частину скелета людини. Друга гілка передбачає набір з 38 полів спорідненості частин, який представляє ступінь асоціації між частинами.

Послідовні етапи використовуються для уточнення прогнозів, зроблених кожним шаром. За допомогою карт достовірності частин між двома частинами формуються двосторонні графіки. Використовуючи значення полів спорідненості частин, слабші ланки у двосторонніх графіках обрізаються. Через вищезазначені кроки людські скелети можуть бути оцінені та призначені кожній людині на зображенні.

### 2.1.2 DeepCut

DeepCut [4] – це підхід знизу вгору для оцінювання пози для людини. Автори при роботі визначили такі завдання:

- підготувати набір кандидатів  $D$  частини тіла. Цей набір представляє всі
- можливі розташування частин тіла для кожної людини на зображенні. Вибрати підмножину частин тіла з вищевказаного набору кандидатських частин тіла
- позначити кожну обрану частину тіла одним із класів частин тіла  $C$ . Класи частин тіла представляють типи частин тіла, такі як «рука», «нога», «тулуб» тощо;
- виокремлення частин тіла, які належать одній людині.

Вищезазначені проблеми були вирішені шляхом моделювання їх в задачу Integer Linear Programming (ILP). Моделюється система, розглядаючи триплекси  $(x, y, z)$  бінарних випадкових змінних з доменами.

Розглянемо двох кандидатів на частину тіла  $d$  і  $d'$  з набору кандидатів на частину тіла  $D$  та класів  $c$  і  $c'$  з набору класів  $C$ . Кандидати на частину тіла були

отримані через Faster RCNN або Dense CNN. Тепер ми можемо розробити наступний набір тверджень.

Якщо  $x(d, c) = 1$ , то це означає, що кандидат частини тіла  $d$  належить до класу  $c$ . Також  $y(d, d') = 1$  вказує на те, що кандидат частини тіла  $d$  і  $d'$  належать одній людині.

Вони також визначають  $z(d, d', c, c') = x(d, c) * x(d', c') * y(d, d')$ . Якщо вказане значення рівняння дорівнює 1, то це означає, що кандидат частини тіла  $d$  належить до класу  $c$ , кандидат частини тіла  $d'$  належить до класу  $c'$ , і, нарешті, кандидати в частину тіла  $d, d'$  належать тій же особі.

Останнє твердження може бути використане для розмежування поз тіла, що належать різним людям. Зрозуміло, що вищезазначені твердження можна сформулювати з точки зору лінійних рівнянь як функції ( $x, y, z$ ). Таким чином встановлюється ціла лінійна програма, за допомогою якої може бути оцінена поза кількох осіб.

### 2.1.3 Hierarchical RNN for Skeleton Based Action Detection

У роботі [5] автори використовують поділ людського скелета на п'ять анатомічних частин (тулуб, дві руки і дві ноги) замість того, щоб подавати на вхід нейронної мережі безпосередньо єдиний скелет. В кожній з п'яти частин виділяються кілька значимих точок, рух яких і відслідковуються. Кожна з п'яти частин подається на двосторонню нейронну мережу (Bidirectional Recurrent Neural Network - BRNN).

По мірі зростання кількості шарів образи, виділені підмережами, ієрархічно об'єднуються на вхід для наступних шарів. В кінці повнозв'язний шар і softmax шар завершують архітектуру, видаючи кінцеве уявлення для класифікації рухів. Архітектура даної ієрархічної нейронної мережі з описом частин тіла, за які відповідає кожен шар, представлена на рисунку 2.2.

Точність даного методу є досить високою, але він має істотний недолік: в якості вхідних даних використовуються дані захоплення рухів (motion capture), і це створює необхідність в спеціальному обладнанні для отримання інформації про рухи конкретної людини.

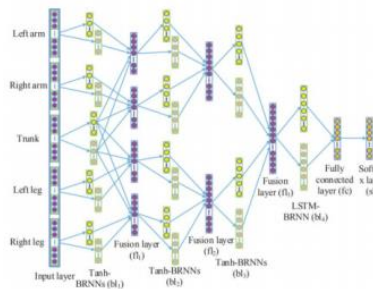


Рисунок 2.2 – Архітектура ієрархічної нейронної мережі [5]

#### 2.1.4. Regional multi-person pose estimation (RMPE або AlphaPose)

RMPE [6] - популярний метод «зверху-вниз» для оцінювання поз людини.

Автори стверджують, що методи зверху-вниз зазвичай залежать від точності детектора людини, оскільки оцінка позиції проводиться в області, де знаходиться людина. Отже, помилки в локалізації та передбачувані дублікати обмежувального поля можуть призвести до того, що алгоритм розпізнавання пози людини створюватиметься неоптимально.

Щоб вирішити цю проблему, автори запропонували використовувати Symmetric Spatial Transformer Network (SSTN) [6] для вилучення окремо взятої особи з неточного обмежувального поля. У цій області використовується Single Person Pose

Estimator (SPPE) для оцінки скелета людської пози для цієї людини. A Spatial DeTransformer Network (SDTN) використовується для трансформації оціночної пози людини до початкової системи координат зображень. Нарешті, параметрична

Non-Maximum Suppression(NMS) техніка використовується для вирішення питання надмірно створених поз.

Крім того, автори впроваджують Pose Guided Proposals Generator для розширення зразків навчання, які можуть краще допомогти тренувати мережі SPPE та SSTN. Важливою особливістю RMPE є те, що цю методику можна поширити на будь-яку комбінацію алгоритму виявлення людини та SPPE.

### 2.1.5 R-CNN (Regions With CNNs)

Архітектура мережі R-CNN [7] була розроблена командою з UC Berkley для застосування Convolution Neural Networks до задачі детектування об'єктів. Процедuru детектування об'єктів мережею R-CNN можна розбити на наступні кроки:

- виділення регіонів-кандидатів за допомогою Selective Search;
- перетворення регіону в розмір, який приймає CNN CaffeNet;
- отримання за допомогою CNN 4096-розмірного вектору ознак;
- проведення N бінарних класифікацій кожного вектору ознак за допомогою N лінійних SVM;
- лінійна регресія параметрів рамки регіону для більш точного охоплення
- об'єкта.

R-CNN в основному грає роль класифікатора, і він не передбачає межі об'єкта (крім уточнення за допомогою регресії обмежувальною рамкою). Його точність залежить від продуктивності модуля пропозиції регіону. У декількох роботах були запропоновані способи використання глибоких мереж для прогнозування обмежувальних рамок об'єктів.

Fast R-CNN[8] запропонували прискорити процес R-CNN за рахунок пари модифікацій: пропускати через CNN не кожен з 2000 регіоно-кандидатів окремо, а

цілком все зображення. Запропоновані регіони потім накладаються на отриману загальну карту ознак. Перетворення ознак, які потрапили в різні регіони, до фіксованих розмірів проводилося за допомогою процедури RoIPooling. Вікно регіону шириною  $w$  і висотою  $h$  поділялося на сітку, що має  $HW$  комірок розміром  $h/H$   $w/W$ . По кожній такій клітинці проводився Max Pooling для вибору тільки одного значення, даючи матрицю ознак  $HW$ .

Після покращень, зроблених в Fast R-CNN, найвужчим місцем нейронної мережі виявився механізм генерації регіонів-кандидатів. У 2015 команда з Microsoft Research змогла зробити цей етап значно швидшим. Вони запропонували обчислювати регіони не по початковому зображенню, а по карті ознак, отриманих з CNN (рисунок 2.3).

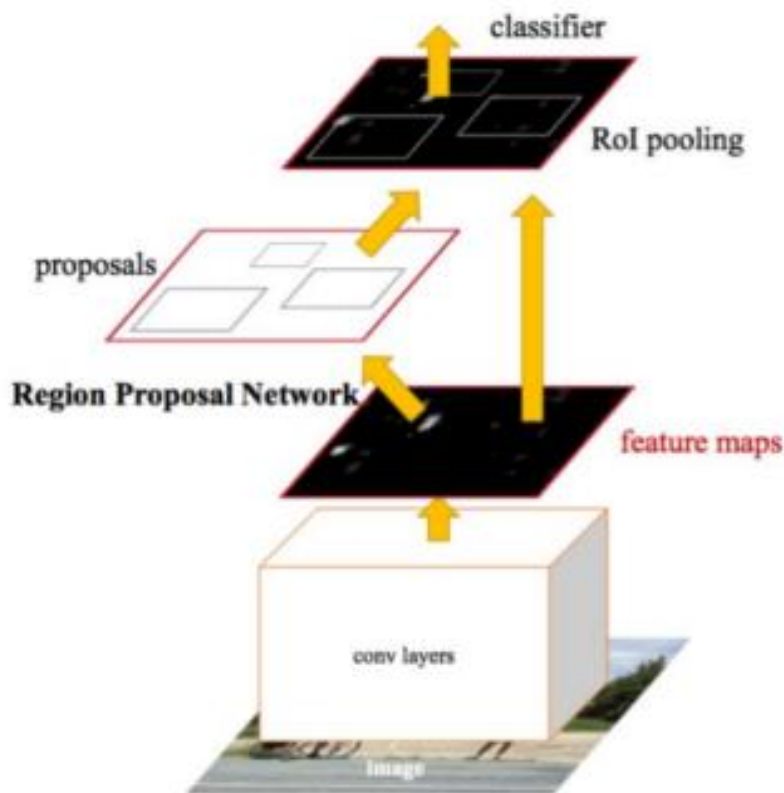


Рисунок 2.3 - Архітектура Faster R-CNN[8]

Для цього було додано модуль Region Proposal Network(RPN). В рамках RPN по витягнутих CNN ознаками проходять «міні-нейромережею» з 3x3 вікном. Отримані з її допомогою значення передаються в два паралельних повнозв'язних шари: boxregression layer(reg) і box-classification layer(cls).

### 2.1.6 Mask R-CNN

Mask R-CNN[7] продовжує архітектуру Faster R-CNN шляхом додавання ще однієї гілки, яка передбачає положення маски, що покриває знайдений об'єкт, і вирішує вже завдання instance segmentation(рисунок 2.4).

Маска являє собою просто прямокутну матрицю, в якій 1 на деякій позиції означає приналежність відповідного пікселя об'єкту заданого класу, 0 - що піксель об'єкту не належить. Виділення маски відбувається в class-agnostic стилі: маски передбачаються окремо для кожного класу, без попереднього знання, що зображено в діапазоні, і потім просто вибирається маска класу, яка перемогла в незалежному класифікаторі. Стверджується, що такий підхід більш ефективний.

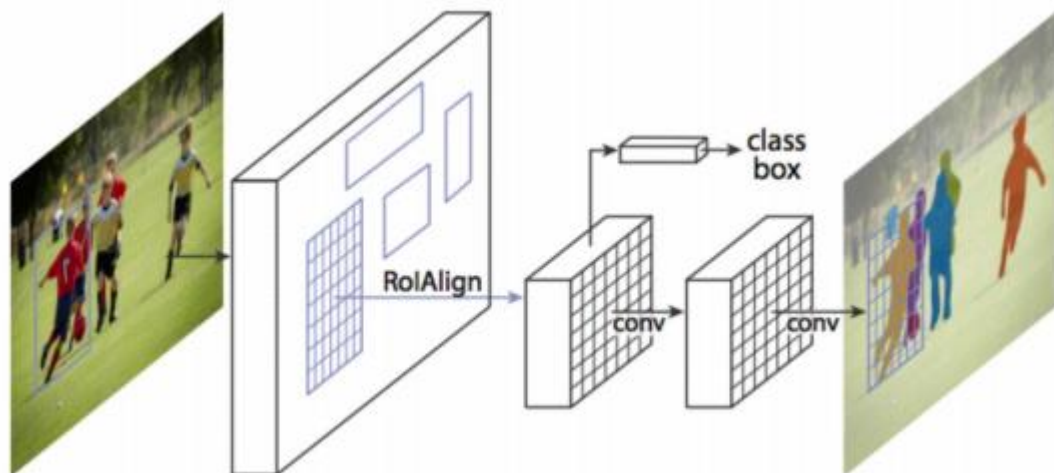


Рисунок 2.4 - Архітектура Mask R-CNN[7]

Одна з основних модифікацій, що виникла через необхідність передбачати маску - зміна процедури RoIPool(обчислює матрицю ознак для регіону-кандидата) на так звану RoIAlign. Справа в тому, що карта ознак, отримана з CNN, має менший розмір, ніж вихідне зображення, і регіон, який охоплює на зображенні цілочислену кількість пікселів, не виходить відобразити в пропорційний регіон карти з цілочисельною кількістю ознак.

RoIPool проблема вирішувалася просто округленням дрібних значень до цілих. Такий підхід нормально працює при виділенні обмежувальної рамки, але обчислена на основі таких даних маска виходить занадто неточною. В протипагу цьому, в RoIAlign не використовується округлення, всі числа залишаються дійсними, а для обчислення значень ознак використовується білінійна інтерполяція по чотирьом найближчим цілочисельним точкам.

Паралельно алгоритм виявлення об'єктів може бути навчений для визначення місця розташування осіб. Поєднуючи інформацію про місцезнаходження людини, а також їх набір ключових точок, ми отримуємо скелет-позу людини для кожної людини на зображенні.

Цей метод нагадує підхід «зверху вниз», але етап виявлення людини виконується паралельно етапу виявлення частин тіла. Іншими словами, етап виявлення ключових точок і етап виявлення людини не залежать один від одного.

## 2.2. Огляд існуючих рішень для реалізації проекту

Постава людини може проявляти змістовну виразність, і тому необхідний порівняно більш оптимальний підхід щодо вивчення визнання пози людей. Розпізнавання постави людей на основі обладнання, що визначає рух, поступово було зосереджено на досліджуваному полі, що стосується взаємодії між людьми та комп'ютером з останніх кількох років. Оскільки компанія «Майкрософт» представила недорогий пристрій зондування руху, Kinect, у 2010 році багато

установ та науковців проводили дослідження на основі Kinect та багатьох випадків взаємодії між людьми та комп'ютерами. Крім того, до цього соматосенсорного апарату вже застосовано розпізнавання постави людей, і досягнуто хороших ефектів.

Мало хто з дослідників використовував зображення глибини для розпізнавання постави людини. Попереднє дослідження [9] отримало контур людини за допомогою зображення глибини та детектора ребра Кенні. Після трансформації на відстані, що стосується розрахунку щодо положення голови, застосовувався підхід, який базується на моделі. Підхід реалізував виявлення людей завдяки додаванню до двовимірної моделі контуру голови, а також до тривимірного режиму поверхні голови для кінцевої мети здійснення виявлення тіла людей. План сегментації був висунутий з метою сегментації людей з оточуючого його середовища та отримання вичерпного цілого контурного зображення на основі точки виявлення. Потім були досягнуті цілі виявлення та відстеження людини. Дослідникам [10, 11] вдалося реалізувати передбачення положення суглоба людського тіла з тривимірної точки зору на основі єдиного зображення глибини. Крім того, вони також здійснили позначення проміжного представлення позиції з метою відображення оцінки пози, що є складною для відносно легкої проблеми стосовно класифікації на піксель, оцінюваних частин тіла за допомогою різноманітних наборів даних тренувань та шляхом застосування може бути отримано повторне проектування результатів щодо повторного проектування, а також пошук локальних зразків, а також оцінки 3D довіри для декількох суглобів тіла. У статті [12] запропоновано гібридний метод розпізнавання, а разом з деякими прийомами щодо обробки зображення, зображення глибини, створене датчиками Kinect, прийняті з метою визначення п'яти різних людей. П'ять поз людей стоять, присідають, сидять, згинаються і лежать. У [13] зображення Kinect було зафіксовано зображення глибини, а також оцінювали верхню кінцівку постави людини та рух верхньої кінцівки.

Кілька дослідників також використовували дані скелета для розпізнавання постави людини. Попереднє дослідження [14] отримало дані скелетів від Kinect

протягом семи різних експериментів. Існують чотири види характеристик, витягнутих із скелета людей, які були прийняті з метою усвідомлення розпізнавання поставок тіла у людей. Чотири характеристики - згинання, сидячи, стоячи та лежачи. Попереднє дослідження [15] запропонувало алгоритм щодо виявлення поз людей, а також розпізнавання поз людей, що є багатокласним на основі характеристик геометрії. Серйозні характеристики щодо кута були перетворені з даних Kinect щодо тривимірного скелета. Вдавшись до підтримуючої векторної машини або короткого SVM, що супроводжувалось ядром полінома, було здійснено класифікацію пози людей.

Що стосується попереднього дослідження [16], то існувало чотири різні підходи щодо класифікації пози людей, які були встановлені для порівняння:

- підтримуюча векторна машина, або SVM коротше;
- зворотна розмноження нейронної мережі, або Коротко кажучи, BPNN;
- наївний Байєс;
- дерево рішень.

Перевірка чотирьох вищезгаданих підходів була здійснена, перевірена за допомогою трьох поз (стоячи, сидячи та лежачи). Висновок полягав у тому, що точність BPNN сягала 100%, а середня точність чотирьох методів становила 93,72%.

Дані зображення глибини та скелета також поєднували численні дослідники для розпізнавання постави людини. Попереднє дослідження [17] реалізувало отримання тривимірних характеристик тіла на основі інформації координат, яка є тривимірною завдяки прийняттю зображення глибини. Крім того, ідентифікація пози людей, які є тривимірними, була здійснена на основі моделей відносно скелетного суглоба людей та багатовимірного набору даних. Попереднє дослідження [18] поєднало оцінки, що стосуються анатомії людського тіла, а також моделі скелета людей, виміряло відстані частин тіла за допомогою геодезичної відстані та здійснило оцінку постави людей за допомогою прийняття зображення глибини від Kinect.

Компанія Microsoft представила Kinect v2 в 2014 році, і вона продемонструвала значні поліпшення на відміну від Kinect v1, включаючи різні аспекти:

- Kinect v2 у два рази точніший за Kinect v1 у близькому діапазоні;
- точність 3D-реконструкції та відстеження людей значно покращується в різних середовищах із Kinect v2;
- Kinect v2 забезпечує підвищену стійкість до штучного освітлення та сонячного світла;
- дальність виявлення більша, ніж у Kinect v1;
- зображення глибини роздільної здатності Kinect v2 вище, ніж у Kinect v1;
- Kinect v2 може безпосередньо виводити дані про глибину людського тіла [19].

Отже, ми вирішили використати Kinect v2 для цього дослідження. Однак SDK 2.0 для розпізнавання скелетів є далеко не цілком правильним, незважаючи на те, що інформація, що стосується скелета, є правильною за умови, що голова людей розташована на найвищому місці тіла людей, а інші частини тіла не перетинаються. Коли скелетний суглоб перекривається іншим, наприклад, в місцях згинання та лежачи, інформація про скелет може бути невірною. Попереднє дослідження [20] запропонувало метод репарації оклюзії єдиної точки суглоба людини, але не один суглоб перекривався у багатьох положеннях. Дослідження [9, 13] не використовували скелетних даних і розпізнавали поставу лише за даними глибини. У попередньому дослідженні [12] дослідники не прийняли SDK від Kinect з метою визнання пози людей. Що стосується наближеної в сучасній літературі, то існують деякі недоліки, наприклад, обмеження, незначні пози людей, яких можна розпізнати, а також низький рівень розпізнавання.

### 3 ВИЗНАЧЕННЯ ПОЛОЖЕННЯ ТІЛА З ВИКОРИСТАННЯМ KINECT

Як описано в пункті розділу 2.2, загальна схема потоку щодо гібридного підходу можна побачити на рис. 3.1.

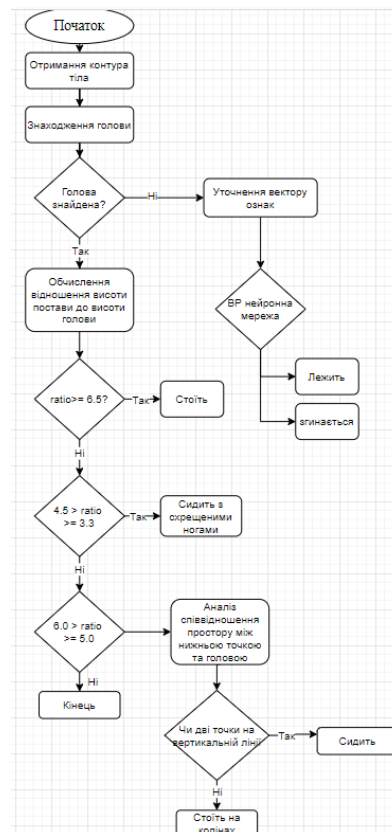


Рисунок 3.1 – Загальна схема потоку гібридного підходу

#### 3.1 Створення центру ваги тіла, контуру та висоти тіла.

Через технічні обмеження Kinect v2 він повинен працювати в простому середовищі. Однак, прийнявши Kinect v2, ідентифікація, а також вихід прямим способом глибинних даних щодо області гуманоїдів можуть бути реалізовані. Крім

того, диференціація між переднім планом та фоном стала непотрібною, а стопи та земля на одній і тій же ділянці також відрізняються відносно точно [21]. Ми використовуємо Kinect v2 та Kinect SDK 2.0 для Windows для витягування області гуманоїдів, але деякі шуми виникають через відображення на землі. Високочастотний фільтр використовувався для видалення шумів. Приклад роботи високочастотного фільтру описано в статті[22]. Але це дослідження не фокусується на фільтрі високих частот. Таким чином, фільтр низьких частот не описаний далі.

Контур людини отримує оператор Canny відповідно до площі тіла людини, як показано на рис. 3.2. На цьому малюнку, що стосується червоного прямокутника, який можна побачити в (a), він може вважатися центром тяжіння відносно площі тіла людей; контур людини показаний у (b), а червоний прямокутник можна розглядати як центр тяжіння контуру людей.

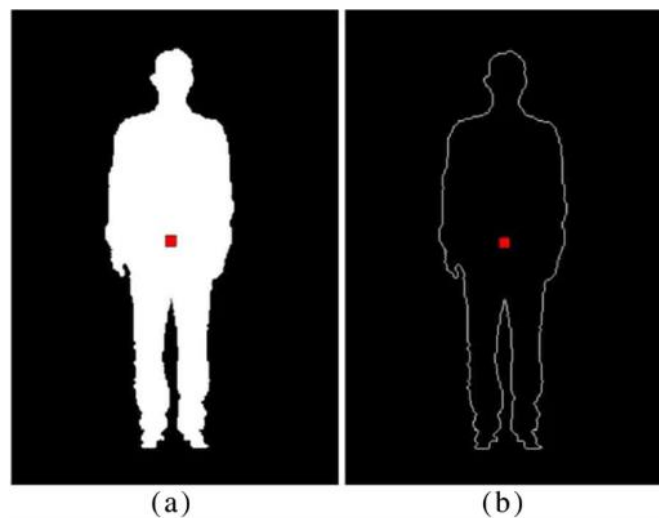


Рисунок 3.2 – Центр ваги в області людського тіла та контуру людини

Трапляються дві ситуації. Правильним є центр ваги по відношенню до площі тіла людей. Крім того, неправильний може вважатися центром тяжкості за межами площі тіла людей, тому що функції векторів є невірними. У цій неправильній ситуації центр ваги робиться як джерело; малюються горизонтальні та вертикальні лінії; вибирається лінія, яка точно відповідає двом точкам перетину на лінії контуру

тіла. Новий центр тяжіння змінюється в центрі положення стосовно двох точок, які перетинаються, як показано на рис. 3.3.

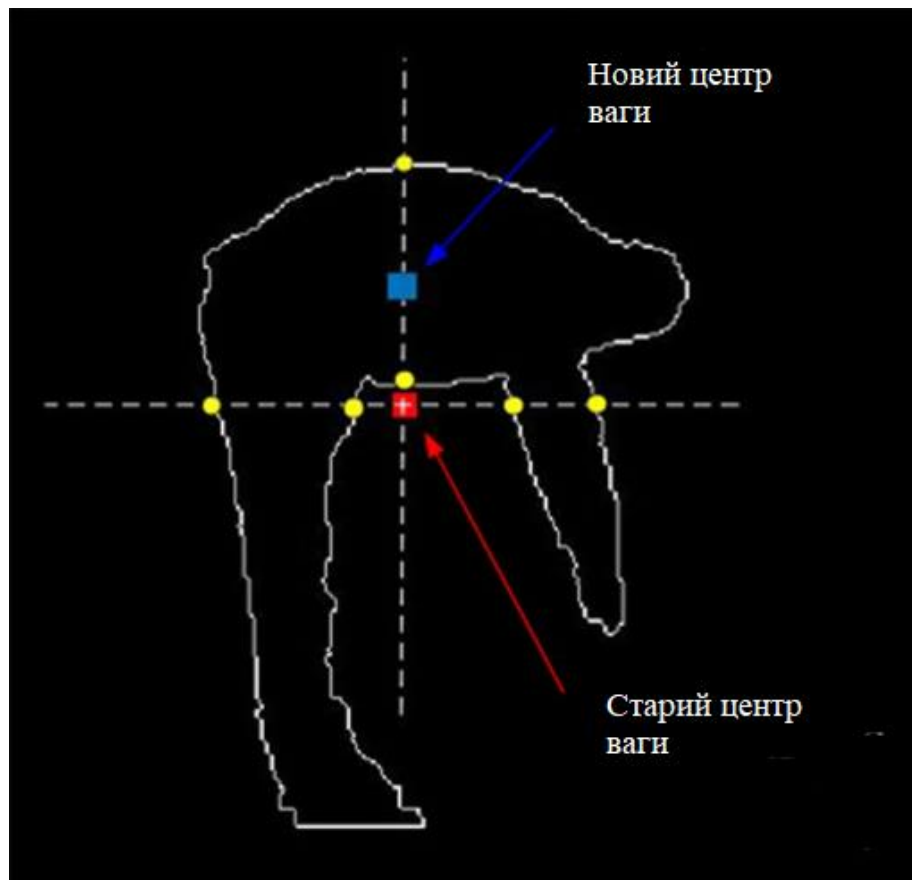


Рисунок 3.3 – Заміна центру ваги людського контуру

### 3.2 Локалізація голови

У цьому дослідженні постави попередньо оцінюються за положенням голови та висотою голови. Отже, локалізація голови є важливою. У порівнянні з іншими частинами людського тіла голова рідко закривається і є більш сприятливою для отримання. Кореляція його риси та постави висока. Тому алгоритм локалізації голови простий з незначним обчисленням. Під час поз у людей, коли вони стоять, сидять, стають на коліна та сидять, схрестивши ноги, голова не перекривається

іншими частинами тіла. Локалізація голови на основі даних скелета є точною. Використання зображень скелетів від Kinect SDK може, таким чином, отримати точну інформацію про голову. Навпаки, за умови, що голова людей не розташована на найвищому місці людей або інші частини тіла закупаються голову, дані про скелети можуть бути неточними або неправильними. Наш метод використовує дані зображення глибини та скелета інтегровано для розміщення голови. Загальний процес такий:

За координатою голови із зображення скелета оцінюють, чи розташована голова в зоні тіла людей із зображення глибини:

– якщо координата голови розташована не в зоні тіла людей із зображення на глибині, постава не є стоячи, сидячи, стоячи на колінах і сидячи схрещеними ногами. Тоді розпізнавання позовів людей здійснюється завдяки вдачі до BPNN;

– якщо координата голови розташована в зоні тіла людей із зображення глибини, то координата голови є достовірною. Тоді голову розміщують відповідно до позиційного відношення координат між головою та шиєю людей:

а) якщо координата голови нижча за всі координати інших частин тіла, або нахил лінії з'єднання головної координати та координати шиї більший за  $-1$  (або менше  $1$ ). Потім розпізнавання постави обробляється BPNN;

б) якщо координата голови вище всіх координат інших частин тіла, а нахил лінії зв'язку координат шиї та голови людей менше  $-1$  (або більше  $1$ ), то підтверджується, що постава є одним з стоячи, сидячи, стоячи на колінах і сидячи схрещеними ногами. Позу судять про співвідношення між висотами контуру та головою людей та просторовим відношенням між тими характеристичними точками, які є тривимірними.

Нам потрібна точна залежність між двома даними, а саме, вдаючись до зображення глибини, висоти голови контуру людини та різниці у висоті між

головою та шийним вузлом, а також видно зображення кістки. Роздільну здатність зображення глибини та скелета ми встановлюємо як  $512 \times 424$ .

Розпізнавання згинальних і лежачих пози за допомогою BPNN, яка посиляється на мережу прямої передачі, що є багатошаровою, встановленою на основі алгоритму розповсюдження помилок (або ВР коротше). Що стосується нейронної мережі, то вона використовує різницю між фактичним та бажаним виходами для корекції прав на з'єднання мережевих шарів та корекції шару за шаром ззаду вперед. BPNN має великі переваги у вирішенні нелінійних задач або нелінійних структурних задач: він може використовувати вхідні та вихідні змінні нейронної мережі для навчальної мережі для досягнення нелінійних цілей калібрування. Один зразок має  $m$  вхідних вузлів,  $n$  вихідних вузлів і прихованих вузлів в одному або декількох прихованих шарах. Багато прихованих шарів вимагає значної кількості тренувань. Згідно теорії Колмогорова, тришарова мережа ВР може підходити до будь-якої мережі за розумною структурою, а також належним вагою. Як і для будь-якої безперервної функції, її можна наблизити мережею ВР, яка має три шари [23, 24]. Ці три шари стосуються вхідного шару, прихованого шару, а також вихідного шару. В результаті була обрана нами мережа ВР, яка наділена трьома шарами, що мають відносно просту структуру.

Початок вичислення центру тяжкості відбувається від самого лівого пікселя, а рух – за годинниковою стрілкою до кінця пікселя, як показано на рис. 3.4.

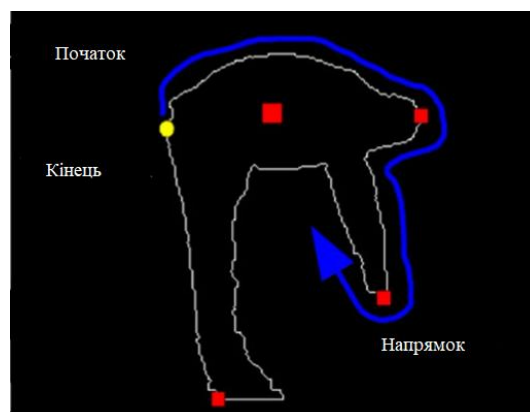


Рисунок 3.4 – Розрахування центру тяжкості

### 3.3 Навчання BPNN

У BPNN легко зрозуміти кількість відносно нейронів вхідного шару, а також нейронів вихідного шару, але нейрони прихованого шару впливають на продуктивність BPNN. Вісім типів постуральних зразків вводяться в BPNN в цьому дослідженні, і кожен зразок має вісім ознак ознак. BPNN виводить два оцінені результати постави, тобто є вісім нейронів вхідного шару, а також два нейрони вихідного шару, але точне число щодо нейронів прихованого шару потрібно визначити. Для навчання прийнято вісім типів постуральних зразків, як показано на рис. 3.5.

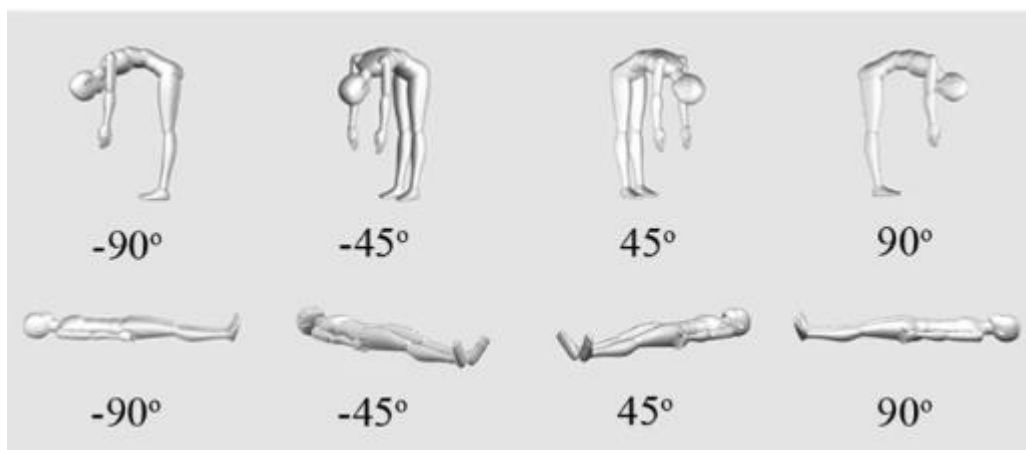


Рисунок 3.5 – Вісім типів постуральних зразків

Характеристики векторів згинання значно відрізняються від лежачих. Тому в цьому дослідженні BPNN було застосовано простим способом із введенням декількох типів зразків до BPNN, і кожен тип зразків був розташований приблизно з однаковою кількістю. Вважається, що така домовленість не призведе до проблеми перенавчання. Хоча відібрані зразки були репрезентативними та майже не мали шуму зразків, у цьому дослідженні була прийнята методика раннього зупинення [25], щоб уникнути проблеми перенапруги. У цьому дослідженні точність навчальної мети з нейронної мережі визначається як 0,001, а щодо розміру кроку

навчання – 0,01. На рисунку 3.6 показано, що щодо оптимальної ефективності тренувань це реалізується в епоху 67.

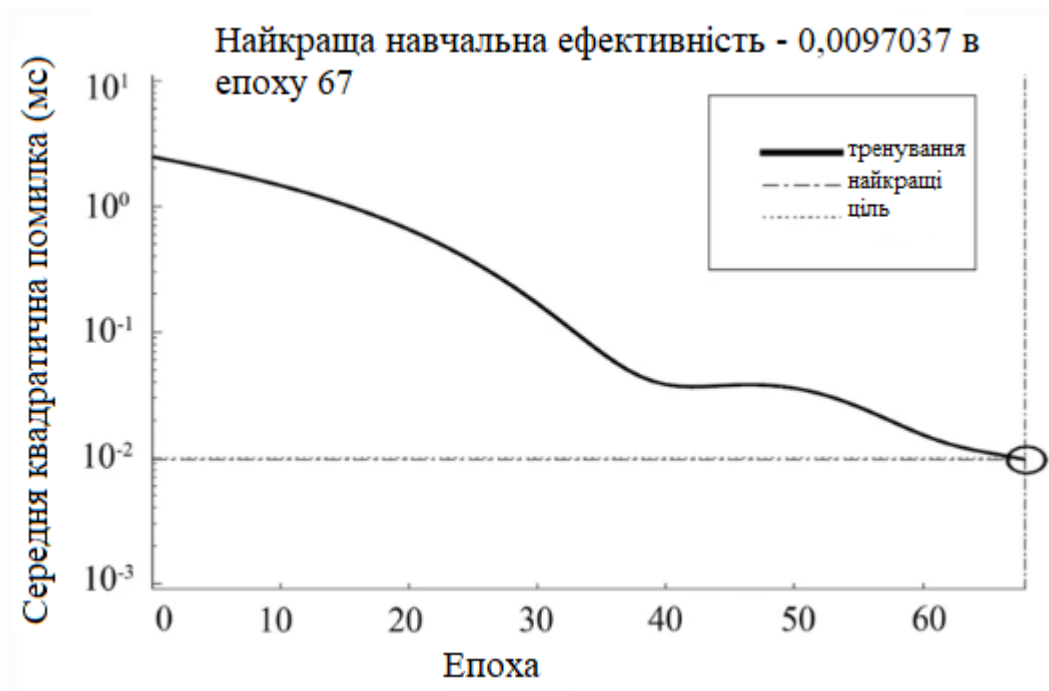


Рисунок 3.6 – Крива тренувань BPNN

## 4 ВИЗНАЧЕННЯ ТІЛА ТА ПОЗИ ЛЮДИНИ З ВИКОРИСТАННЯМ АЛГОРИТМУ OPENPOSE

OpenPose - система, яка може відстежувати рух тіла, включаючи руки і обличчя в режимі реального часу. Вона використовує комп'ютерний зір і машинне навчання для обробки відео-кадрів і може відстежувати одночасно кілька людей.

Ключові точки - набір з 18 або 25 точок, які відповідають частинам людського тіла, наприклад: 0 - ніс, 1 - підборіддя, 2 - праве плече і тд. Саме ці точки допомагають комп'ютеру розпізнавати людей на фото та відео, та отримувати уявлення про їх позицію. Авжеж, такий спосіб розпізнавання не може відрізнити одну людину від іншою, оскільки ключових точок занадто мало для розпізнавання обличчя, але побудова скелетонів з цих точок представляє собою перспективний напрямок, який може бути використаний в багатьох галузях. Наприклад: спортивні тренування, промислові підприємства, аг-додатки та інше.

OpenPose не є чимось новим для світу, оскільки вже доступні 2D бібліотеки оцінки тіла, такі як Mask R-CNN або Alpha-Pose. Вони, в свою чергу, вимагають від своїх користувачів реалізувати більшу частину конвеєра, власний зчитувач кадрів (наприклад, відео, зображень або потокового відео), дисплей для візуалізації результатів, генерація вихідних файлів з результатами і т.д. OpenPose подолав всі ці проблеми і пропонує “рішення із коробки”. Він може працювати на різних платформах, в тому числі Ubuntu, Windows, Mac OSX та вбудовані системи (наприклад, Nvidia Tegra TX2). Він також забезпечує підтримку різних апаратних засобів, таких як графічні процесори CUDA, графічні процесори OpenCL і пристрої CPU-only.

Дана бібліотека використовує в якості системи для глибинного навчання - кодову базу Caffe. У кодовій базі Caffe жорстко кодуються додатки, навчання та нейронні мережі. Їх важко налаштувати. Крім того, відсутні основні функціональні можливості, такі як підтримка вбудованих платформ і паралельне навчання GPU. Всі ці обмеження роблять процес розгортки OpenPose на різних системах занадто

важким та інколи неможливим. Саме тому почали з'являтися реалізації OpenPose які використовують інші системи глибокого навчання, наприклад: TensorFlow, PyTorch. [26].

Перед тим як почати використовувати алгоритм OpenPose, визначимо різницю між парами та частинами. [27]. Частина тіла є елементом тіла, як шия, ліве плече або праве стегно. Пара є парою частин, зв'язок між частинами. Можна сказати, кінцівка, але зв'язок між носом і лівим оком, безумовно, не кінцівка. Також ми маємо справу зі зв'язками між вухами і плечами, які не існують у реальному житті.

Загалом, для визначення ключових точок людини, OpenPose використовує алгоритм, що містить декілька кроків [28].

Крок 1. Конвертація зображення (рис. 4.1).



Рисунок 4.1 – Крок конвертації зображення

Модель приймає на вхід зображення розміром  $w \times h$ . Кожен піксель складається з трьох каналів, кожен канал - значення з діапазону  $[0, 255]$ . На цьому етапі приводимо піксель з діапазону  $[0, 255]$  до діапазону  $[-1, 1]$ .

Крок 2. Обробка нейронною мережею. Головною частиною алгоритму є конволюційна нейронна мережа (рис. 4.2). Остання операція нейронної мережі повертає тензор, що складається з 57 матриць. Ця остання операція є лише конкатенацією двох різних тензорів: теплових карт і PAF, плюс одна матриця для фону.

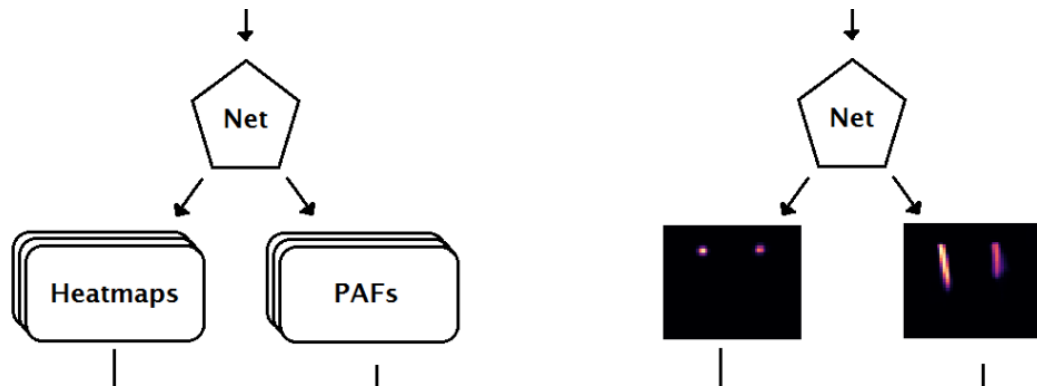


Рисунок 4.2 – Крок обробка нейронною мережею

Теплова карта є матрицею, яка використовується для впевненості у тому, що певний піксель містить певну частину тіла (рис. 4.3), 18 теплових карт, пов'язаних з кожною частиною тіла. Ми виберемо розташування частин тіла з цих 18 матриць.

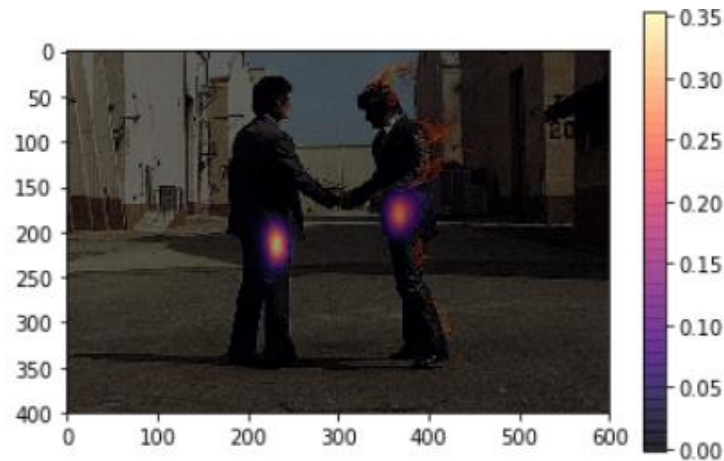


Рисунок 4.3 – Приклад теплової карти

PAFs (Part Affinity Fields) - матриці, які дають інформацію про положення і орієнтацію пар. Тобто, завдяки цим даним ми дізнаємося, де знаходиться решта тіла по відношенню до даної частини тіла. Вони приходять у пари: для кожної частини ми маємо PAF у напрямку "x" і PAF у напрямку "y". Є 38 PAF, пов'язаних з кожною з пар (рис. 4.4).

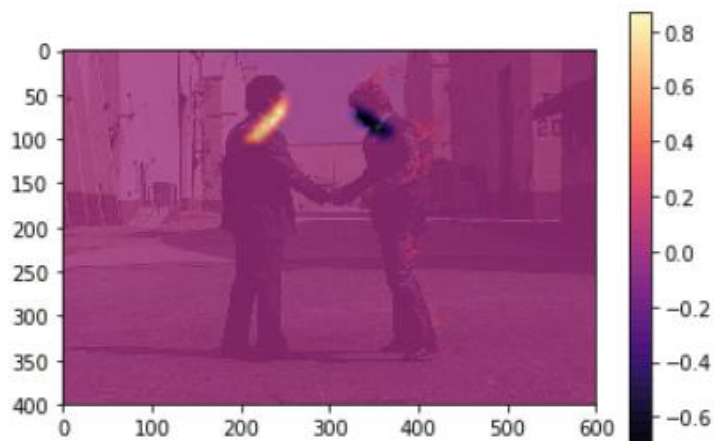


Рисунок 4.4 – Приклад PAFs

Крок 3. Використання алгоритму “Non Maximum Suppression”. Зараз ми маємо справу з деякими околами ключових точок. Наступним кроком нам необхідно перейти від областей точок, що вказують на частину тіла до однієї точки з цієї області, які будуть локальними максимумами цієї області. Цей процес здійснюється за допомогою алгоритму NMS (рис. 4.5).

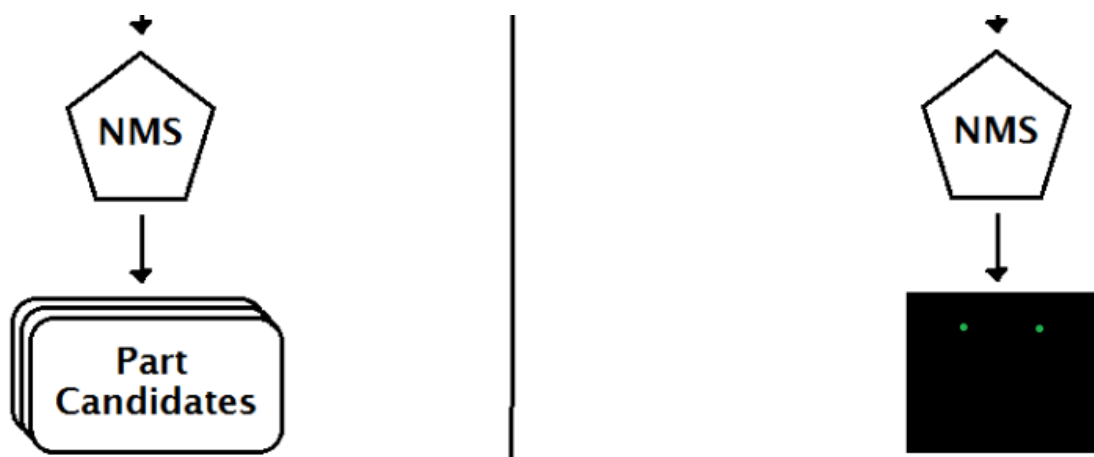


Рисунок 4.5 – Крок алгоритму “NMS”

Дуже часто під час розпізнавання об'єктів на виході роботи алгоритма ми можемо отримати декілька об'єктів, що насправді вказують на один й той самий об'єкт NMS (рис. 4.6).

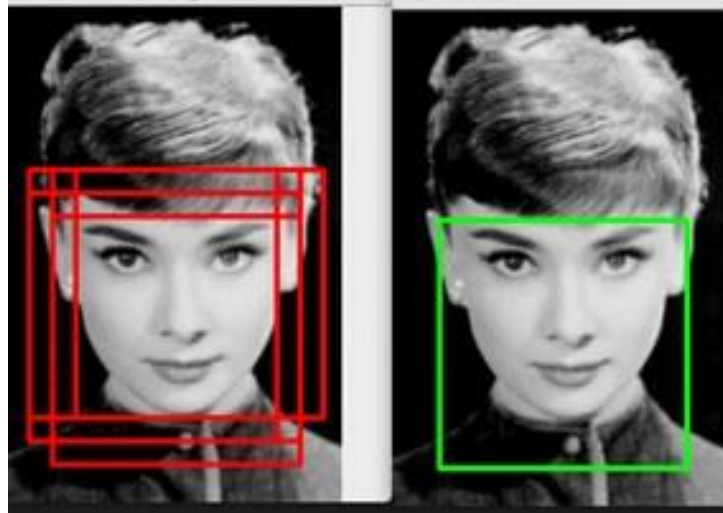


Рисунок 4.6 – застосування non-max suppression

Це характерно для алгоритмів, що проходять по зображенню вікном (конволюційні неронні мережі, гістограми орієнтованих градієнтів, тощо). Постає задача “видалення” об'єктів, що насправді містять один й той самий об'єкт. Загалом, роботу алгоритму можна описати наступним чином:

- серед групи розпізнаних об'єктів, що мають спільну область, обирається той, що з найбільшою вірогідністю правильного розпізнавання;
- об'єкти, вірогідність яких не перевищує якусь задану границю, ігноруються;
- далі проходимося по всіх об'єктах, що мають з обраним спільну область зображення. Якщо ця спільна область не перевищує якийсь заданий процент, то це означає, що це різні об'єкти і ми залишаємо обидва. В іншому випадку, скоріш за все ми маємо справу з одним й тим самим об'єктом. Тоді ми “ігноруємо” об'єкт з не максимальною вірогідністю правильного розпізнавання. Тому алгоритм отримав назву non-max suppression.

Крок 4. Побудова дводольного графу (рис. 4.7). Тепер, коли ми знайшли кандидатів для кожної з частин тіла, нам потрібно з'єднати їх, щоб утворити пари.

Припустимо, що для даного зображення ми знайшли набір кандидатів на шию і набір кандидатів для правого стегна. Для кожної шиї можлива асоціація, або з'єднання кандидата, з кожним з правих стегон. Необхідно серед всіх можливих з'єднань знайти правильні. Знаходження найкращої відповідності між вершинами дводольного графа є добре відомою проблемою в теорії графів, відомою як задача про призначення. Щоб вирішити цю проблему, кожне ребро на графіку повинно мати вагу.

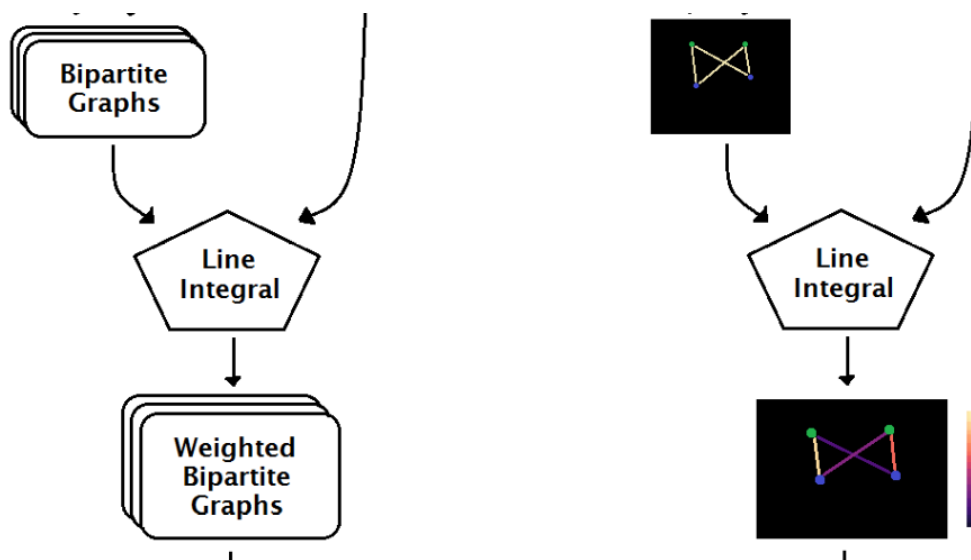


Рисунок 4.7 – Дводольний граф

Тепер, коли ми знайшли кандидатів для кожної з частин тіла, нам потрібно з'єднати їх, щоб утворити пари. Припустимо, що для даного зображення ми знайшли набір кандидатів на шию і набір кандидатів для правого стегна. Для кожної шиї можлива асоціація, або з'єднання кандидата, з кожним з правих стегон. Необхідно серед всіх можливих з'єднань знайти правильні. Знаходження найкращої відповідності між вершинами дводольного графа є добре відомою проблемою в теорії графів, відомою як задача про призначення. Щоб вирішити цю проблему, кожне ребро на графіку повинно мати вагу.

Ми обчислимо лінійний інтеграл вздовж сегмента, що з'єднує кожну пару кандидатів, над відповідними РАФ ( $x$  та  $y$ ) для цієї пари. Інтеграл лінії вимірює вплив даного поля (у нашому випадку, полях афінності частин) вздовж заданої кривої (у нашому випадку, можливих зв'язків між кандидатами). Інтеграл лінії надасть кожному з'єднанню рахунок, який буде збережений у зваженому дводольному графіку і дозволить вирішити задачу призначення.

Крок 5. Призначення. Зважений двосторонній графік показує всі можливі зв'язки між кандидатами з двох частин і містить оцінку для кожного з'єднання. Тепер задача полягає в тому, щоб знайти зв'язки, які максимізують загальний бал, тобто вирішують задачу призначення (рис. 4.8).

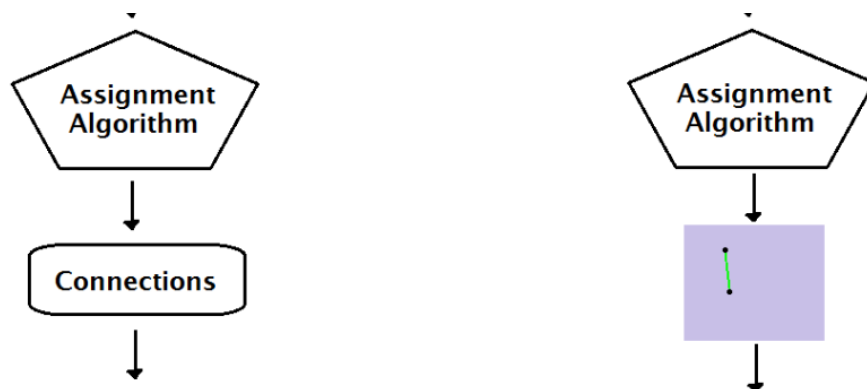


Рисунок 4.8 – Крок призначення

Крок 6. З'єднання і отримання результату. Останнім кроком є перетворення цих виявлених з'єднань в кінцеві скелети. Почнемо з наївного припущення: спочатку кожне з'єднання належить іншій людині. Таким чином, ми маємо таку ж кількість людей, як і виявлені нами з'єднання. Нехай людина буде представлена набором з'єднань, тоді люди будуть представлені у вигляді  $\{N_1, N_2, \dots, N_k\}$ , де  $N_k$  - людина. Кожна з цих множин - тобто кожна людина - містить пару. Давайте

опишемо частину як кортеж індексу, координату в напрямку «x» і координату в напрямку «у». Тут відбувається злиття: якщо люди H1 та H2 поділяють індекс частини з однаковими координатами, вони розділяють одну і ту ж частину. Отже, H1 та H2 є одними і тими ж людьми. Тому ми об'єднуємо обидва набори в H1 і видаляємо H2 (рис. 4.9).



Рисунок 4.9 – Крок з'єднання

В результаті, ми отримали групу людей, представлених у вигляді наборів точок.

## 5. ПРОГРАМНА РЕАЛІЗАЦІЯ. ЕКСПЕРИМЕНТАЛЬНІ РЕЗУЛЬТАТИ

Практична цінність методів розпізнавання тіла та пози людини з різних ракурсів залежить від розміщення в приміщенні стаціонарних камер або камери мобільного телефону, то доцільно створити програмне забезпечення Pose Recognition, за допомогою якого можливо отримати доступ до них та можливість керувати та отримувати інформацію. Таке програмне забезпечення стане складовою частиною більш складної системи, що містить серверну частину, адміністративну панель, API для мобільного додатку клієнтів камери та полегшення використання бібліотеки OpenPose.

### 5.1 Опис програмного забезпечення

#### 5.1.1 Серверна частина та адміністративна панель програми Pose Recognition

Серверна частина програми Pose Recognition складається з двох частин: сайту з API і сайту з функціоналом авторизації і адміністративної панеллю. Сайти повинні бути розгорнуті на IIS сервері з операційною системою, який підтримує ASP .Net Core версії 2.0. Мова імплементації C#.

Для управління інформацією про зали, камери та тренажери використовується адміністративна панель, де необхідно авторизуватися (рис.5.1).

Після авторизації користувач потрапляє на сторінку «Клуби» (рис. 5.2), де є можливість додати новий клуб та інформацію про нього або отредагувати дані.

Войти

Username  
Логин

Password  
Пароль

Запомнить данные

Войти Отменить

Рисунок 5.1 – Вікно авторизації

Система має можливість додовання камери на відповідній сторінці, де редактуються два поля у Камери»: Назва (обов'язкове) та Опис (опціональне) (рис.5.3).

Фитнес-клубы

Добавить новый клуб

Название	Адрес	Телефон	Дополнительная информация
Фитнес-клуб, Харьков, Театральная	г. Харьков, переулок Театральный 1/3	+380577616404, +380937616404	Фитнес-клуб с удивительным расположением в центре города.
Харьков, Салтовка, м. Героев Труда	г. Харьков, ул. Ак. Павлова, 1440, ТЦ «Космос», 5 этаж	+380577616407, +380937616407	Пн - Пт : 07.00 - 23.00, Сб - Вс : 09.00 - 23.00
Харьков, Бавария	г. Харьков, ул. Свиноренко, 9	+380577614419, +380937614419	Пн - Пт : 07.00 - 23.00, Сб - Вс : 09.00 - 23.00
Харьков, м. Дворец Спорта	г. Харьков, пр. Московский 257, НИИ «Кондиционер», 2 этаж	+380577172520, +380937614490	Пн - Пт : 07.00 - 23.00, Сб - Вс : 09.00 - 23.00
Харьков, пос. Жуковского	г. Харьков, ул. Чкалова 1 (кафе молодость)	+380577616420, +380937616420	В настоящее время нет грядущих событий.
Харьков, Холодная Гора	г. Харьков, Полтавский шлях, 184	+380577616412, +380937616412	Множество групповых тренировок, помимо спортзала.
Харьков, Салтовка	НИИ ПТИмаш пр. Юбилейный 56 (пр.50 лет ВЛКСМ 56)	+380577616405, +380937616406	
Харьков, Серповая	г. Харьков, ул. Серповая, 4а	+380937617772, +380577617772	Есть бассейн.

Рисунок 5.2 – Сторінка «Клуби»

Заповніть ці поля інформацією і натисніть «Додати». Далі ви будете перенаправлені на сторінку з списком камер. Детальніше про цю камеру можна дізнатися перейшовши за посиланням із загального списку або з сторінки залу, до якого дана камера прив'язана (рис.5.4.).

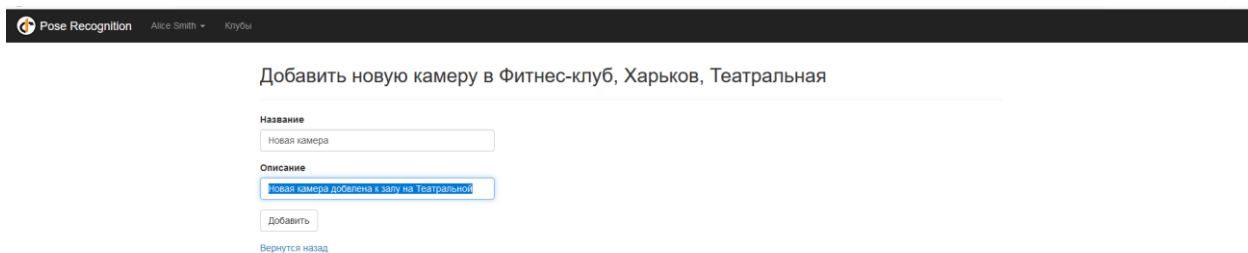


Рисунок 5.2 – Сторінка «Додати камеру»

На сторінці з детальною інформацією про камеру крім полів, які може редагувати користувач, тобто назви і опису, присутній поле з назвою тренажерного залу, до якого прив'язана камера, а так само ClientId і ClientSecret - два поля, які необхідні для авторизації камери клієнта і відправки повідомлень користувачеві. Для перегляду детальної інформації про клуб, натисніть «Детальніше». Ви будете перенаправлені на сторінку клубу, де вказана вся інформація про клуб, а так само список камер, які є в цьому клубі, з списком прив'язаним до камер тренажерів або залів. Щоб побачити список тренажерів для кожної камери, необхідно натиснути кнопку «Подивитися тренажери». Якщо хочете редагувати камеру, натисніть «Редагувати». Якщо хочете додати новий тренажер до камери -натисніть «Додати тренажер» (рис.5.4).

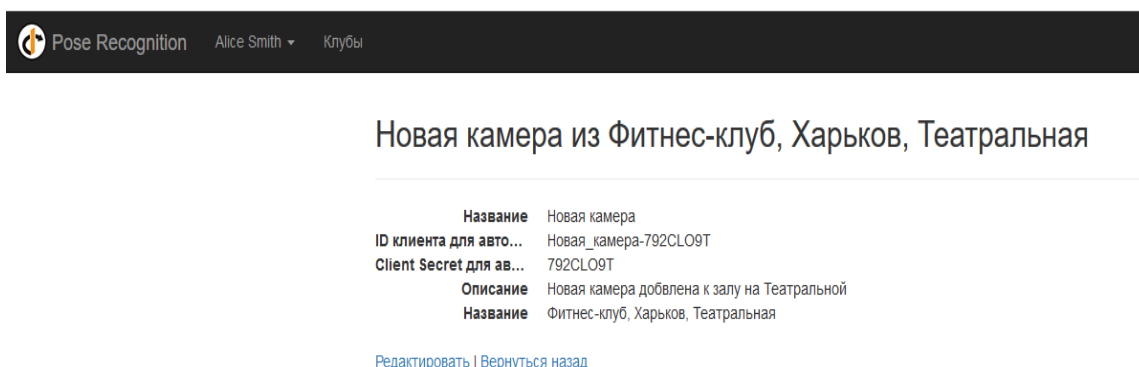


Рисунок 5.3 – Сторінка «Інформація про камеру»

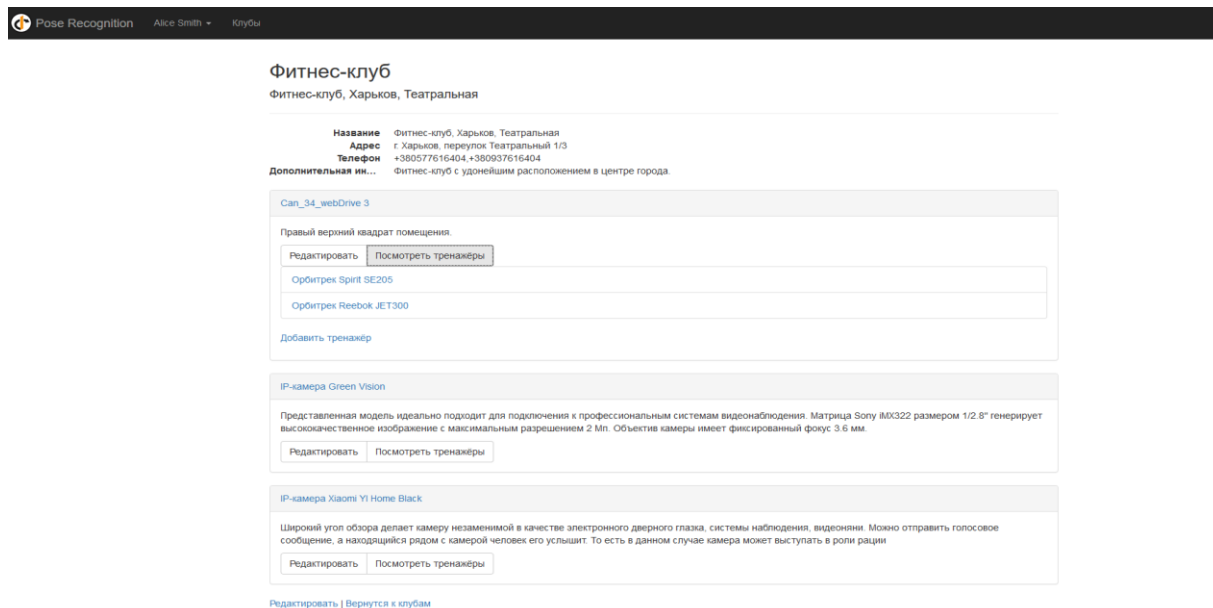


Рисунок 5.4 – Сторінка «Детальніше»

Коли ви натиснете «Додати тренажер», ви будете перенаправлені на сторінку з формою додавання тренажера. Заповніть поля «Назва» та «Код моделі тренажера», а так само виберіть камеру, до якої буде прив'язаний тренажер. У списку будуть доступні всі камери, які в даний момент зареєстровані в залі. Після того, як інформація заповнена, натисніть кнопку «Додати». Ви будете перенаправлені на сторінку з усіма тренажерами (рис.5.5).

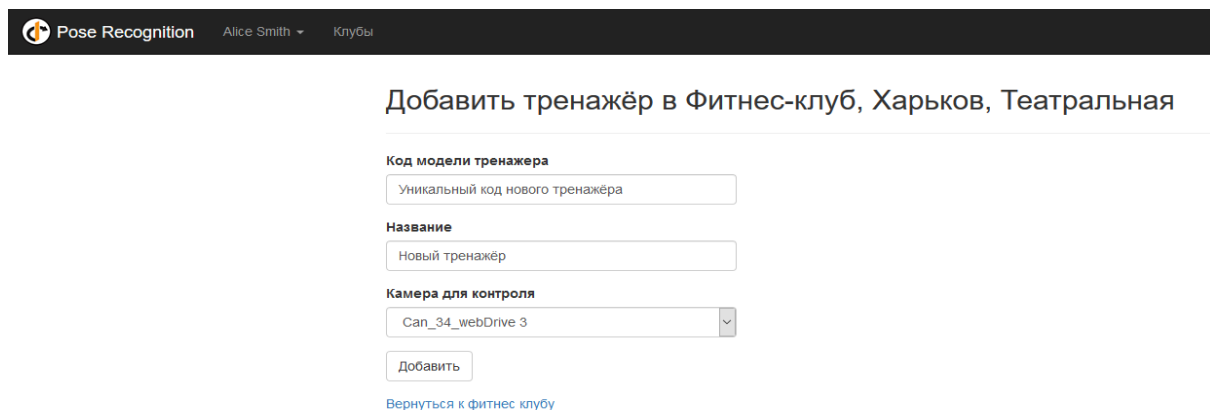


Рисунок 5.5 – Сторінка «Додати тренажер»

### 5.1.2 API для мобільного додатку та клієнтів камери

Крок 1. Авторизація.

Посилання запити: <http://pose-recognition-api.gear.host/connect/token>

Headers: Content-Type: application/x-www-form-urlencoded.

Авторизація користувача. Для цього необхідно передати наступні поля (рис. 5.6).

```
Grant_type: password
Password: <пароль користувача>
UserName: <логін користувача>
Cleint_id: ro.client
Client_secret: secret
Scope: api1
```

Рисунок 5.6 – Авторизація користувача

Авторизація клієнта камери. Для цього необхідно передати наступні поля (рис. 5.7).

```
Grant_type: client_credentials
Cleint_id: <ClientId з сторінки детальної інформації камери в адмінці>
Client_secret: <ClientSecret з сторінки детальної інформації камери>
Scope: api1
```

Рисунок 5.7 – Авторизація клієнта камери

Крок 2. Повідомлення про неправильне виконання вправи. Для того, щоб пристрій з камерою могло відправити запит на сервер з метою повідомити користувача про неправильне виконання, необхідно авторизуватися за допомогою client\_credentials типу авторизації. ClientId і ClientSecret необхідно скопіювати з сторінки з детальною інформацією про камеру в адміністративній панелі (рис. 5.8)..

## Новая камера из Фитнес-клуб, Харьков, Театральная

Название	Новая камера
ID клиента для авто...	Новая_камера-792CLO9T
Client Secret для ав...	792CLO9T
Описание	Новая камера добвлена к залу на Театральной
Название	Фитнес-клуб, Харьков, Театральная

Рисунок 5.8 – Авторизація камери

Після того, як клієнт камери був авторизований, необхідно додати заголовок з токеном авторизації і відправити запит:

Url: `http://poserecognitionrealapi.gear.host/api/exercises`

Request Type: `HttpPost`

Body: `{ "ApparatusId": <номер тренажера>, "Comment": "<текст повідомлення>" }`.

Після цього дані про помилку прив'яжуться до поточного сеансу виконання вправи користувача і будуть показані користувачеві, коли він запросить інформацію про свої помилки, або при наступному сеансі на теекущем тренажері. У випадку з наявністю сертифіката для iOS пристрою буде відправлятися `push-notification`, щоб користувачеві зміг отримати оперативний фідбек.

Крок 3. API тренувань включає:

– автоматично. Якщо у користувача були зауваження під час попереднього виконання, то вони будуть так само додані у відповіді. Необхідно додати заголовок авторизації до запиту.

Url: `http://localhost:5001/api/apparatuses/<id>`

RequestType: `HttpGet`

ResponseType: `json`

Response format:

```
{
  "Name": string,
  "Id": int,
  "Available": boolean,
  "LastExecutionNotes": string
};
```

– початок тренування. Для початку тренування необхідно в тілі запити передати айдішнік тренажера, а так само додати заголовок авторизації до запити.

Url: <http://poserecognitionrealapi.gear.host/api/UserActivities>

RequestType:HttpPost

Request format:

```
{"ApparatusId": <id тренажера, считанный с QR кода на тренажере>}
```

ResponseType: json

Response format:

```
{
  "id": int,
  "apparatusId": int,
  "started": datetime,
  "isFinished": boolean,
  "finished": datetime,
  "userId": string
};
```

– отримання зауважень про неправильність виконання вправи. Необхідно додати заголовок авторизації до запити.

Url: <http://poserecognitionrealapi.gear.host/api/UserActivities/<id тренування отриманий в запиті по початку тренування> / execution-notes>

RequestType: HttpGet

ResponseType: json

Response format:

```
{
  "executionNotes": string
};
```

– закінчення тренування. Необхідно додати заголовок авторизації до запити.

Url: <http://poserecognitionrealapi.gear.host/api/UserActivities/<id тренування отриманий в запиті по початку тренування >>

RequestType: HttpDelete

ResponseType: json

Response format:

```
{
  "id": int,
  "apparatusId": int,
  "isFinished": boolean,
  "finished": datetime
}
```

### 5.1.3 API для використання OpenPose

Для полегшення використання бібліотеки, розробниками було створено спеціальне API, з використання мови програмування Python. Нейронна мережа працює з окремим зображенням. Для того, щоб опрацювати відео-потік необхідно надати на вхід мережі кожен кадр.

TfPoseEstimator. Клас завантажує модель у пам'ять. Щоб імпортувати клас:

```
from tf_pose.estimator import TfPoseEstimator
```

Ініціалізація класу проводиться наступним чином:

```
TfPoseEstimator(get_graph_path(model_path), target_size=(w, h))
```

get\_graph\_path - функція, яка імпортується з tf\_pose.networks. Функція буде обчислювальний граф з файлу, який міститься у змінній model\_path:

- target\_size - ширина і висота зображення у пікселях;
- inference - метод класу TfPoseEstimator, запускає процес розпізнавання скелетонів для зображення. Приймає зображення, повертає багатомірний масив з частинами тіла. Частини тіла для кожної людини містяться в окремому масиві. Елемент масиву - це об'єкт вигляду BodyPart:0-(0.69, 0.41) score=0.72, де BodyPart:0 - тип частини тіла (праве око, лівий лікоть, тощо), (0.69, 0.41) - координати точки, score=0.72;
- draw\_humans(npimg, humans, imgcopy=False) - використовують для малювання скелетонів людей на зображенні. Для кожної людини з масиву humans отримує набір точок, body\_parts та малює їх, після чого розставляє лінії.

Робота з відео за допомогою OpenCV. Щоб обробити кадри відео-потіку використовується клас VideoCapture з бібліотеки openCV:

- VideoCapture(video\_path) - отримує на вхід місцезнаходження відео та повертає об'єкт, який надає доступ до метаданих і кадрів відео-потіку;

– VideoCapture.read - метод, що повертає кадр відео-потoku і переходить до наступного кадру.

Для збереження кадрів у відео-файл звернемося до класу VideoWriter.

– VideoWriter - клас, що отримує на вхід об'єкт IOFileWrapper (дескриптор файлу), його розміри (ширина, висота). Повертає об'єкт, який інкапсулює запис на відео на жорсткий диск;

– VideoWriter.write - метод, що додає кадр до вихідного відео-файлу.

#### 5.1.4 Системні вимоги

Мінімальні вимоги (обробка відео-файлів), що дозволяють ознайомитися з роботою OpenPose (обробка в реальному часі відсутня):

- процесор двоядерний 2.4 ГГц;
- оперативна пам'ять 4 ГБ., DDR3;
- операційна система Windows 10.

#### 5.2 Експериментальні результати за допомогою Kinect

Апаратне середовище щодо дослідження в цій роботі полягає в наступному: комп'ютер із процесором Intel Core (R) процесором I7-9600K та 16G пам'яті, відеокарта Nvidia RTX 2070, Kinect v2 для Windows; Системне середовище: 64-розрядна версія Windows 10 Pro; Середовище IDE: Visual Studio 2019 та Kinect SDK 2.0.

В експериментах беруть участь двадцять дорослих зі здоровими тілами (15 чоловіків і 5 жінок). Серед них 4 чоловіка та 1 жінка є не лише моделями з точки зору навчальних зразків щодо BPNN, але й учасниками всіх експериментів. Усі учасники мають різну висоту та вагу. Відстань учасника та Kinect v2 становить від 2,0 до 4,2 м, а відстань Kinect v2 та землі – 70 або 95 см.

Перед виконанням всієї програми потрібно зібрати навчальні зразки вигинів і лежачих пози. Зразки використовуються для тренування BPNN. Учасників обирають відповідно, які мають різний зріст, а також вагу. Крім того, від них потрібно було поставити дві пози, які згиналися, лежачи орієнтованими на Kinect в чотирьох напрямках ( $\pm 45^\circ$  і  $\pm 90^\circ$ ). У якості отриманих даних тренінгу були взяті особливості векторів кожної пози, а також вісім видів вхідних зразків, як показано на рис. 5.9. Кожного учасника оглядають п'ять разів у кожному напрямку. Таким чином, зібрано 200 проб. Усі зразки надходять до BPNN для тренувань, а набір стосовно результатів після тренувань зберігався пізніше.

Що стосується всіх учасників, то вони класифікуються на три набори зразків. Набір 1 відноситься до набору моделей, які проходять навчальні зразки BPNN, набір 2 відноситься до набору учасників, які не проходять вибірки з точки зору вибірки BPNN, а набір 3 відноситься до всього набору учасників. Визнано шість пози людей, і шість поз:

- стоячи;
- сидячи;
- стоячи на колінах;
- сидячи схрещеними ногами;
- згинаючись;
- лежачи.

Крім того, шість поз кожного учасника зроблені у трьох різних експериментальних середовищах (зі світлом в приміщенні, із природним світлом в приміщенні та з природним світлом зовні). Кожен учасник займає шість пози кілька разів, коли він / вона орієнтується на Kinect v2 в різних напрямках. Крім того, він /

вона потім визнається програмою. Напрямки стояння і сидіння з схрещеними ногами становлять  $0^\circ \sim \pm 180^\circ$ , і вони поставлені відповідно 30 і 24 рази; напрямки сидіння і стоячи на колінах -  $0^\circ \sim \pm 90^\circ$ , і вони поставлені відповідно 28 і 20 разів; Напрямки лежачих та згинальних поз становлять  $\pm 45^\circ \sim \pm 90^\circ$ , і вони поставлені відповідно 24 та 20 разів. На рисунку 5.9 показана сцена постанови людини. У таблиці 5.1 наведено експериментальну статистику успішних часів розпізнавання постанови та точності розпізнавання.

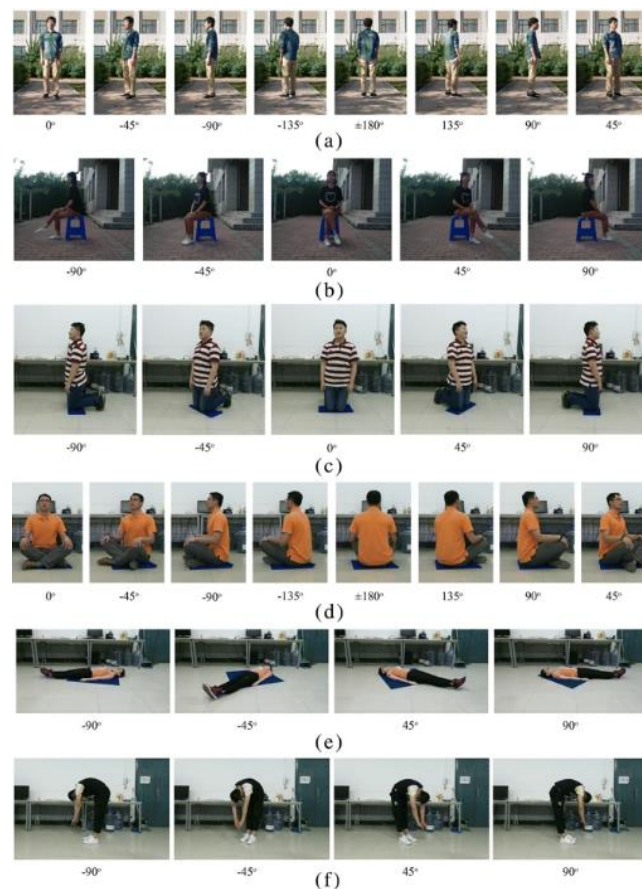


Рисунок 5.1 – Постава людей

Робота програми включає п'ять етапів, етап А – отримання площі та контуру тіла людини; стадія В – локалізація голови; етап С – оцінка співвідношення висоти контуру людини до висоти голови та розрізнення стоячи, сидячи чи стоячи на колінах, сидячи з схрещеними ногами та інших пози; стадія D – судження про сидяче і колінне положення відповідно до відношення між характеристичними

точками, які є тривимірними просторовими; етап Е – це відмінність згинання від лежання на основі BPNN.

Таблиця 5.1 Точність розпізнавання постави людини

	A	B	C	D	E	F	Середня
Набір 1	600/600	544/560	391/40 0	480/480	480/480	400/40 0	2859/292 0
Точність набору 1	100%	97.17%	97.75%	100%	100%	100%	99.14%
Набір 2	780/780	719/728	505/52 0	624/624	621/624	511/52 0	3760/379 6
Точність набору 2	100%	98.76%	97.12%	100%	99.52%	98.27%	99.05%
Набір 3	1380/138 0	1263/128 8	896/92 0	1104/110 4	1101/110 4	911/92 0	6655/671 6
Точність набору 3	100%	98.06%	97.39%	100%	99.73%	99.02%	99.09%

Робота програми включає п'ять етапів, етап А – отримання площі та контуру тіла людини; стадія В – локалізація голови; етап С – оцінка співвідношення висоти контуру людини до висоти голови та розрізнення стоячи, сидячи чи стоячи на колінах, сидячи з схрещеними ногами та інших пози; стадія D – судження про сидяче і колінне положення відповідно до відношення між характеристичними точками, які є тривимірними просторовими; Етап Е – це відмінність згинання від лежання на основі BPNN.

У таблиці 5.2 розглянуто час роботи програми стосовно кожного етапу кожної пози. Час виконання кожного етапу в середньому короткий, тобто час виконання кожної пози коротше або дорівнює 15 мс, що може задовольнити вимогу в реальному часі.

Таблиця 5.2 Час виконання етапів кожної пози (одиниця: мс)

	A	B	C	D	E	F
Етап А	10.218	9.151	8.936	9.178	11.106	9.216
Етап В	0.028	0.032	0.032	0.028	0.022	0.011
Етап С	0.299	0.309	0.382	0.309	0	0
Етап D	0	0.045	0.047	0	0	0
Етап E	0	0	0	0	3.812	3.207
Загалом	10.545	9.537	9.343	9.515	14.94	12.434

Дослідники [15, 26] запропонували методи з метою реалізації розпізнавання десяти пози. Три пози в дослідженні [15] (які стоять, схилиються вперед, сидять на стільці) і три пози в [26] (які стоять, сидять, підхоплення) мають певну схожість із стоянням, сидінням та згинанням, прийнятим у дослідженні сучасності відповідно. Метод, використаний у [15], базувався на характеристиках кута відносно пози, а також на поліноміальному класифікаторі SVM ядра. Крім того, рівень впізнаваності щодо десяти пози становив у середньому 95,87%. У роботі [29] автори використали тривимірні спільні гістограми глибинних зображень, а також приховані моделі Маркова для класифікації пози. Ми порівнюємо ці методи з методами, які використовуються в поточному дослідженні; порівняння можна побачити в таблиці 5.3.

Попереднє дослідження [12] оцінило поставу на колінах за контуром людини та відношення ширини верхньої частини тіла до ширини нижньої частини тіла. Точність становила 99,69%, але експеримент мав передумову, що людина повинна орієнтувати деякий фіксований напрямок ( $\pm 90^\circ$ ,  $\pm 45^\circ$ ) до Kinect. Нижня ширина тіла, більша, ніж ширина верхньої частини тіла, якщо людина стикається з Kinect, може спричинити неправильне судження. У попередньому дослідженні [8] було використано чотири методи розпізнавання пози людини. Найкращим був метод, заснований на BPNN. Однак, хоча його точність розпізнавання становила 100%, метод міг розпізнати лише три пози (стоячи, сидячи та лежачи). Значення його характеристик були вилучені за допомогою даних скелета. Отже, метод не міг

витягти ефективні значення властивостей для згинання та стояння на колінах. Аналогічно позі сидячого в [12], що в [16] вимагало, щоб людина сиділа з фіксованою поставою. Сидячі пози чоловіків і жінок різні. Жінки часто сидять елегантно, тобто сидять із зімкнутими разом ноги і сидять з щиколоткою на коліні. Наш підхід спочатку оцінює постави на колінах і сидінні відповідно до відношення висоти контуру людини до висоти голови, а потім, він розрізняє дві пози відповідно до відношення між характеристичними точками, які є тривимірними просторовими. Такий підхід може судити про більше стоячи на колінах за умови, що люди перебувають у різних напрямках Kinect (від  $-90^\circ$  до  $90^\circ$ ), включаючи людину, що стоїть перед Kinect, і навіть витончене сидяче положення жінки.

Таблиця 5.3 Порівняння точності декількох методів (%)

	A	B	C	D	E	F
Даний метод	100	98.06	97.39	100	99.73	99.02
Стаття [12]	99.38	97.25	99.69	–	99.22	99.53
Стаття [16]	100	100	–	–	–	100
Стаття [15]	95.87	95.87	–	–	95.87	–
Стаття [26]	93.5	91.5	–	–	97.5	–

Даний гібридний підхід має обмеження. З одного боку, вигини та лежачі пози не можуть бути розпізнані в більшій кількості напрямків, коли людина орієнтується на Kinect, оскільки значення ознаки не очевидно і не може отримати кращого значення, коли людина орієнтується на Kinect при 0.

## ВИСНОВКИ

В атестаційній роботі були проаналізовані існуючі методи розпізнавання пози, а також проекти, технічно близькі до даного.

На основі вимог до системи були виділені основні класи та компоненти системи, встановлені зв'язки між ними. Була розглянута реалізація окремих прецедентів.

В ході роботи були отримані наступні результати:

- проведено аналіз програмних аналогів і наукової літератури предметної області;
- сформовані навчальна і тестова вибірки;
- реалізована, навчена і протестована модель нейронної мережі для розпізнавання пози людини у режимі реального часу;
- реалізована серверна частина та адміністративна панель програмної системи Pose Recognition;
- реалізован API для мобільного додатку та клієнтів камери програмної системи Pose Recognition;
- проєдені експериментальні результати за допомогою Kinect.

Продемонстровано, що спроектована система перевершує попередні методи. Середній показник розпізнавання всіх шести пози становить понад 99%. Позиції, що стоять і сидючи з схрещеними ногами, можна розпізнати, коли людина орієнтується на Kinect у будь-якому напрямку (від  $0^\circ$  до  $\pm 180^\circ$ ), а рівень розпізнавання становить 100%; постави на колінах і сидінні можна розпізнати, коли людина орієнтується на Kinect у напрямку від  $0^\circ$  до  $\pm 90^\circ$ , а рівень розпізнавання становить понад 97%; а вигини та положення лежачи визнаються BPNN, а рівень розпізнавання становить понад 99%. Однак вигини та положення лежачи можна розпізнати лише тоді, коли людина орієнтується на Kinect у напрямку від  $\pm 45^\circ$  до  $\pm 90^\circ$ . Частина суглобів тіла затьмарена, тому що ми використовуємо лише один Kinect.

**ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Poselet conditioned pictorial structures / L. Pishchulin [и др.] // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. — 2013. — pp. 588—595
2. Toshev A., Szegedy C. Deeppose: Human pose estimation via deep neural networks // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. – 2014. – pp. 1653 – 1660.
3. Zhe Cao, Tomas Simon, Shih-En Wei Yaser, Sheikh The Robotics Institute, Carnegie Mellon, University: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields // – Режим доступу: <https://arxiv.org/pdf/1611.08050.pdf>
4. L. Pishchulin, E. Insafutdinov, S. Tang, B. Andres, M. Andriluka, P. V. Gehler, and B. Schiele. Deepcut: Joint subset partition and labeling for multi person pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4929–4937, 2016
5. Du Y., Wang W., Wang L. Hierarchical recurrent neural network for skeleton based action recognition. // Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015. – Vol. 7. – No. 12. – P. 1110–1118.
6. H.-S. Fang, S. Xie, Y.-W. Tai, and C. Lu, “RMPE: Regional multi-person pose estimation,” in ICCV, 2017.
7. K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask r-cnn,” in ICCV, 2017.
8. Girshick R. Fast R-CNN. Режим доступу: <https://arxiv.org/pdf/1504.08083>
9. L. Xia, C.C. Chen, J.K. Aggarwal, in Computer Vision and Pattern Recognition Workshops IEEE. Human detection using depth information by Kinect (2011), pp. 15–22
10. J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, A. Blake, in Computer Vision and Pattern Recognition (CVPR) IEEE. Real-time human pose recognition in parts from single depth images (2011), pp. 1297–1304

11. J. Shotton, T. Sharp, A. Kipman, et al., Real-time human pose recognition in parts from single depth images. *Commun. ACM* 56(1), 116–124 (2013)
12. W.J. Wang, J.W. Chang, S.F. Haung, et al., Human posture recognition based on images captured by the Kinect sensor. *Int. J. Adv. Robot. Syst.* 13(2), 1 (2016)
13. S.C. Hsu, J.Y. Huang, W.C. Kao, et al., Human body motion parameters capturing using kinect. *Mach. Vis. Appl.* 26(7–8), 919–932 (2015)
14. T.L. Le, M.Q. Nguyen, T.T.M. Nguyen, in *International Conference on Computing, Management and Telecommunications IEEE*. Human posture recognition using human skeleton provided by Kinect (2013), pp. 340–345
15. P.K. Pisharady, Kinect based body posture detection and recognition system. *Proc. SPIE Int. Soc. Opt. Eng.* 8768(1), 87687F (2013) -87687F-5
16. O. Patsadu, C. Nukoolkit, B. Watanapa, in *International Joint Conference on Computer Science and Software Engineering IEEE*. Human gesture recognition using Kinect camera (2012), pp. 28–32
17. Z. Xiao, M. Fu, Y. Yi, et al., in *International Conference on Intelligent Human-Machine Systems and Cybernetics IEEE*. 3D human postures recognition using Kinect (2012), pp. 344–347
18. L.A. Schwarz, A. Mkhitarian, D. Mateus, et al., Human skeleton tracking from depth data using geodesic distances and optical flow ☆. *Image Vis. Comput.* 30(3), 217–226 (2012)
19. O. Wasenmüller, D. Stricker, in *Asian Conference on Computer Vision*. Comparison of Kinect V1 and V2 depth images in terms of accuracy and precision (Springer, Cham, 2016), pp. 34–45
20. X. Li, Y. Wang, Y. He, G. Zhu, Research on the algorithm of human single joint point repair based on Kinect (a Chinese paper). *Tech. Autom. Appl.* 35(4), 96–98 (2016)
21. H. Li, C. Zhang, W. Quan, C. Han, H. Zhai, T. Liu, An automatic matting algorithms of human figure based on Kinect depth image. *J Chang Univ Sci Technol* 39(6), 81–84 (2016)

22. Білоус Н.В., Власенко В.П, Красов А.І. Вирівнювання яскравості фону зображень текстових знаків з використанням високочастотного фільтра // Вісник Харківського національного університету імені ВН Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління» – 2019 – с.6-16.

23. Nan, Artificial Neural Network Tutorial, 1st Ed (Beijing University of Posts and Telecommunications Press, Beijing, 2006), pp. 47–78

24. H.Y. Shen, Determining the number of BP neural network hidden layer units. J Tianjin Univ Technol 5, 13–15 (2008)

25. J.M. Nasser, D.R. Fairbairn, The application of neural network techniques to structural analysis by implementing an adaptive finite-element mesh generation. AI EDAM 8(3), 177–191 (1994)

26. Single Pose Comparison—a fun application using Human Pose Estimation (Part 2) - [Електроний ресурс]. Режим доступу: <https://becominghuman.ai/single-pose-comparison-a-fun-application-using-human-pose-estimation-part-2-4fd16a8bf0d3>, вільний. - Загол. з екрану.

27. Human pose estimation using OpenPose and tensorflow - [Електроний ресурс]. Режим доступу: <https://arvrjourney.com/human-pose-estimation-using-openpose-with-tensorflow-part-1-7dd4ca5c8027>, вільний. - Загол. з екрану.

28. Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. - [Електроний ресурс]. Режим доступу: <https://arxiv.org/pdf/1611.08050.pdf>, вільний. – Загол. з екрану.

29. S. Thawornwong, D. Enke, The adaptive selection of financial and economic variables for use with artificial neural networks. Aichi Gakuin Daigaku Shigakkai Shi 56(10), 205–232 (2004)

30. L. Xia, C.C. Chen, J.K. Aggarwal, in Computer Vision and Pattern Recognition Workshops. IEEE. View invariant human action recognition using histograms of 3D joints (2012), pp. 20–27