BIRZEIT UNIVERSITY

Faculty of Engineering and Technology

Department of Electrical and Computer Engineering

Home Gym

A Virtual Vision-Based Trainer.

Prepared by:

Shahed Jamhour	1180654
Yasmeena Assi	1180899
Mohammad Abu Zeinah	1181965

Supervised by:

Dr. Yazan Abu Farha

Section:18

A Graduation Project submitted to the Department of Electrical and Computer Engineering in partial fulfillment of the requirements for the degree of B.Sc. in Computer Engineering.

BIRZEIT

February 2023

Abstract

This project aims to design a system that analyzes human movement and physical exercises to serve as an alternative to going to gyms. Our system will encourage people who do not prefer to go to gym to start working out in their homes, which will have a positive impact on their health. The designed system provides feedback to users to help them improve their exercise by analyzing data captured through a camera using machine learning algorithms. The video captured by the camera is processed using computer vision technology to detect the human body and extract the human skeleton information. The extracted data is then prepared and normalized to ensure consistency between users and exercises. The normalized data is classified to accurately identify the exercise being performed to finally provide feedback to assist users in improving their exercise technique.

المستخلص

يهدف هذا المشروع إلى تصميم نظام يحلل حركة الإنسان وتمارينه الرياضية كبديل للذهاب إلى صالات الألعاب الرياضية. سيشجع نظامنا الأشخاص الذين لا يفضلون الذهاب إلى الصالات الرياضية على ممارسة التمارين في منازلهم، والتي ستكون لها تأثير إيجابي على صحتهم. يوفر النظام تغذية راجعة للمستخدمين لمساعدتهم على تحسين تمارينهم من خلال تحليل البيانات التي تم التقاطها من خلال الكاميرا باستخدام خوارزميات التعلم الآلي. يتم معالجة الفيديو الذي يتم التقاطه بواسطة الكاميرا لاستخراج معلومات الهيكل العظمي ومن ثم معالجة البيانات المستخرجة لضمان الاتساق بين المستخدمين والتمارين. تصنف البيانات المطابقة لتحديد التمرين المنفذ لتقديم تغذية راجعة لمساعدة المستخدمين على تحسين تقنية تمارينهم وشكل الحركة الرياضية المطبقة.

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

CNN Convolutional Neural Networks. 9

DTW Dynamic Time Warping. 23

GCN Graph Convolutional Neural Networks. 8

HPE Human pose estimation. 7, 8, 12

KNN K-Nearest Neighbor. 3, 24

Chapter 1

Introduction

It has been shown in hundreds of studies that working out extends your life and improves your health. Indeed, it is essential for every sphere of life as it helps muscles to develop, energy levels to increase, brain functions to improve and it even helps to prevent and treat mental illnesses like depression [9].

Joining a gym has become a popular choice for people looking to improve their physical fitness. However, many individuals struggle to maintain their gym memberships in the long term, as they may find it time-consuming, expensive, crowded, intimidating, confusing, and monotonous. Additionally, some people may lack the motivation to work out on a regular basis. Also, the COVID-19 pandemic has heightened this trend, with gym attendance declining as people seek new methods to remain fit.[9]

As a result, the name "Home GYM" was chosen as a designation for this project, which denotes a fitness facility that can be positioned anywhere within a residence or alternative location of one's choice. The Home Gym project aims to address the barriers that prevent people from engaging in regular physical exercise, such as laziness or lack of access to gym facilities. By providing a convenient and effective way to exercise from the comfort of one's own home, the project hopes to encourage more people to prioritize their health and well-being.

In this project we will design a system that uses machine learning algorithms to analyze and recognize human movement during exercise, allowing it to provide personalized workouts and guidance that are tailored to the user's needs and goals. One of the key features of the Home Gym project is its use of video input and giving feedback.

Through the use of a user's camera, the system will be able to capture and process video data to develop a model that is capable of distinguishing and classifying different exercises. This will allow the system to provide clear instructions and feedback to users as they work out, helping them to perform the exercises correctly and safely. The designed system combines computer vision techniques, machine learning algorithms, and data processing to precisely evaluate the user's posture and offer useful feedback. The system uses a camera to allow users to record their exercise routine. The video is then processed through human pose estimation [1], which is a computer vision task to identify and classify the joints in the human body to extracts the skeleton information from the video.

We have opted for skeleton-based action recognition because it focuses specifically on the movements and postures of human skeletons to recognize actions. A skeleton sequence captures only action information and is immune to contextual distractions such as background variations and lighting changes. The extracted skeleton data is then processed and normalized to ensure that the data is consistent across different users and exercises. Once the data is normalized, it is classified using a K-Nearest Neighbor classifier.[7] Finally, the system generates feedback for the user based on the classification results.

1.1 Motivation

Nowadays, you don't have to hit the gym to get a good workout. Exercising at home is just as effective. People started to prefer to work out at home for a variety of reasons, for example, working out at home eliminates the need to travel to a gym or fitness center which is especially appealing to people with busy schedules. Moreover, here in Palestine, gym operating hours are typically separated for men and women, which may not be suitable for everyone as not everyone can go to the gym at the time they wish. Another reason is privacy, as many people prefer the privacy of working out at home, rather than being in a public space with other people. Working out at home is also budget-friendly, as there is no membership fee. Online fitness resources, such as workout videos or fitness apps, can often be accessed for a lower cost than in-person fitness classes or personal training sessions.The COVID-19 pandemic has accelerated

the trend of digital fitness and online workouts, and many people have found that these apps provide a convenient and effective way to stay active [1].That is where the idea of our project came from, to help people do the exercises they want, the way they want, whenever and wherever they want!

1.2 Problem Statement

In this project, we will develop an application that helps users to work out at home. Users will have an easier time using our system as they can open their device camera to record themselves performing an exercise. The system shall analyze the input stream, to check if the users are doing the exercises correctly and precisely. Then it shall provide them with feedback after each exercise, as well as during the exercise. The system will give the user feedback on whether the exercise was done correctly, and we will ask them to redo it if it was done incorrectly. So, in our project, we are going to start with simple exercises like push up, jumping jacks and squats.

1.3 Contribution

This project makes a significant contribution in the field of exercise and physical health by overcoming the obstacles that hinder consistent physical activity. It provides an efficient solution through the application of machine learning algorithms that tailor workouts and advice to the individuals. Through the use of a user's camera, the system will be able to capture and process video data to develop a model that is capable of distinguishing and classifying different exercises. This capability of the system will enable delivery of feedback to users while they engage in physical activity. The provision of immediate feedback has the potential to facilitate prompt correction of the exercise performance. The system can use overlays or annotations t to highlight when and what was wrong in the performance of the exercise, and what needs to be improved. This not only provides a clear visual representation of the feedback, but it also allows for a more engaging and interactive experience for the users.

1.4 Report structure

This report is structured into five chapters, each presenting the underlying principles and relevant scientific considerations. Chapter 2 offers a systematic literature review, affording a thorough understanding of the challenges that the project endeavors to ameliorate while chapter 3 is focused on discussing the system design and implementation. In chapter 4, the focus shifts to the discussion of the results. The fifth and final chapter is devoted to discussing the conclusion of the study as well as outlining potential areas for future work.

Chapter 2

Background and Related Work

2.1 Related work

It is not surprising that the market has become saturated with apps that use artificial intelligence to help people become healthier. for example, INGGEZ [11] is a Palestinian startup that allows people to enjoy a healthy lifestyle by having access to flexible and cost-effective fitness options with the help of personal trainers. INGEZZ uses the camera, artificial intelligence (AI), and human pose estimation to identify the skeleton of the body to help to achieve the correct posture during doing the exercises. INGGEZ offers a personalized training experience to its users, wherein a personal trainer monitors their progress and provides guidance on how to perform exercises correctly. This is achieved by having the user mimic the trainer's movements in order to learn the correct technique. The app corrects the user, step by step explaining how to stand up, bend over and what you should feel at a certain moment. It gives the user feedback while they are doing the exercise and tells them what exactly they did wrong and how to do it correctly. Also, it is somehow comfortable to use, and the users have fun while they use it, since there is music, and because they interact with it using signs. INGGEZ has many interesting features like real time coaching, measuring users progress by setting a goal and picking a specific program and even competing with friends in virtual fitness workouts and challenges. INGGEZ has many interesting features like real time Coaching, measuring users progress by setting a goal and picking a specific

program and even competing with friends in virtual fitness workouts and challenges [8].

Zenia [15] is a mobile app powered by artificial intelligence, which uses human pose estimation to provide users with tips on how to achieve a proper posture while they work out. This software detects the pose using the camera and estimates how accurate it is and if it is correct, the predicted pose is displayed in green. Red will replace green if the pose is incorrect as shown in the figure below.

In order for the app to function, it must access the device camera. However, no one will see the user during the classes. The app values privacy and supports a safe and secure experience, no personal data is stored, and the app has even an offline mode.

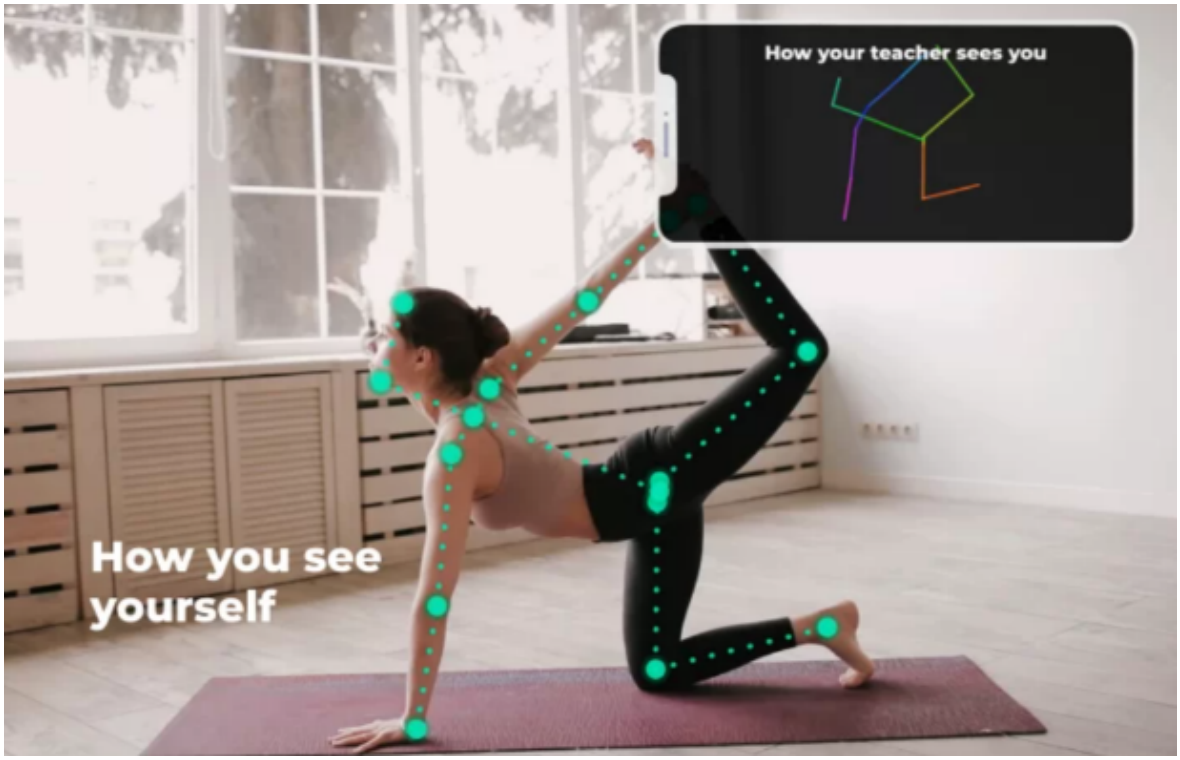


Figure 2.1: Zenia.

2.2 Background

This project will use the user's device camera to detect the user's body parts or joint position using the technique of human pose estimation which is a computer vision task that visualizes a person's orientation in a graphical format. Human pose estimation[9] seeks to produce a skeleton-like representation of the human body, which can then be

processed further to be used in specific tasks.

In previous Human pose estimation studies, graphical models were used to estimate human poses. These models consist of joints and rigid parts and generally follow two steps, first extracting handcrafted features from raw data, then learning classifiers based on these features [9]. The overall process of a body pose estimation system begins with data being captured and uploaded to the system for processing. When detecting motion, we need to analyze a sequence of images, not just a still image, since we need to extract how key points change over time and during the movement pattern. Once the image is uploaded, the human pose estimation system detects and tracks the key points for analysis.[14]. Existing studies have explored various modalities for feature representation, such as RGB frames, optical flows, audio waves, and human skeletons.[6]

2.2.1 Graph Convolutional Networks (GCNs)

Graph Convolutional Neural Networks (GCNs) are a type of neural network that can operate on graph-structured data. In recent years, GCNs have shown great promise in various computer vision tasks, including human pose estimation. GCNs can be used to model the dependencies between different body joints and predict their positions more accurately. In Graph Convolutional Neural Networks , every human joint is viewed as a node at every timestep. In the spatial and temporal dimensions, neighboring nodes are connected through edges. After constructing a graph, layers of convolution are applied to it to discover patterns of action in space and time. As a result of their good performance on standard benchmarks for skeleton-based action recognition, GCNs have become a standard approach for skeleton-based sequence processing[13].

Despite the encouraging results, GCN-based methods have some limitations:

1. Robustness: Although GCN directly handles joint coordinates, the distribution shift of coordinates, which frequently occurs when using a different pose estimator to acquire the coordinates, has a significant impact on recognition ability[13].

2. Interoperability: Many previous works demonstrate that representations from different modalities, such as RGB, optical flows, and skeletons, are complemen-

tary to one another. Thus, an effective combination such modalities can often boost performance in action recognition. Due to GCN's irregular graph of skeletons, it can be difficult to integrate with other modalities, especially in the early stages, since they are usually represented on regular grids.

3. Scalability: GCN considers every human joint to be a node, and its complexity rises linearly with the number of people, restricting its application to scenarios involving several people, such as group activity identification.

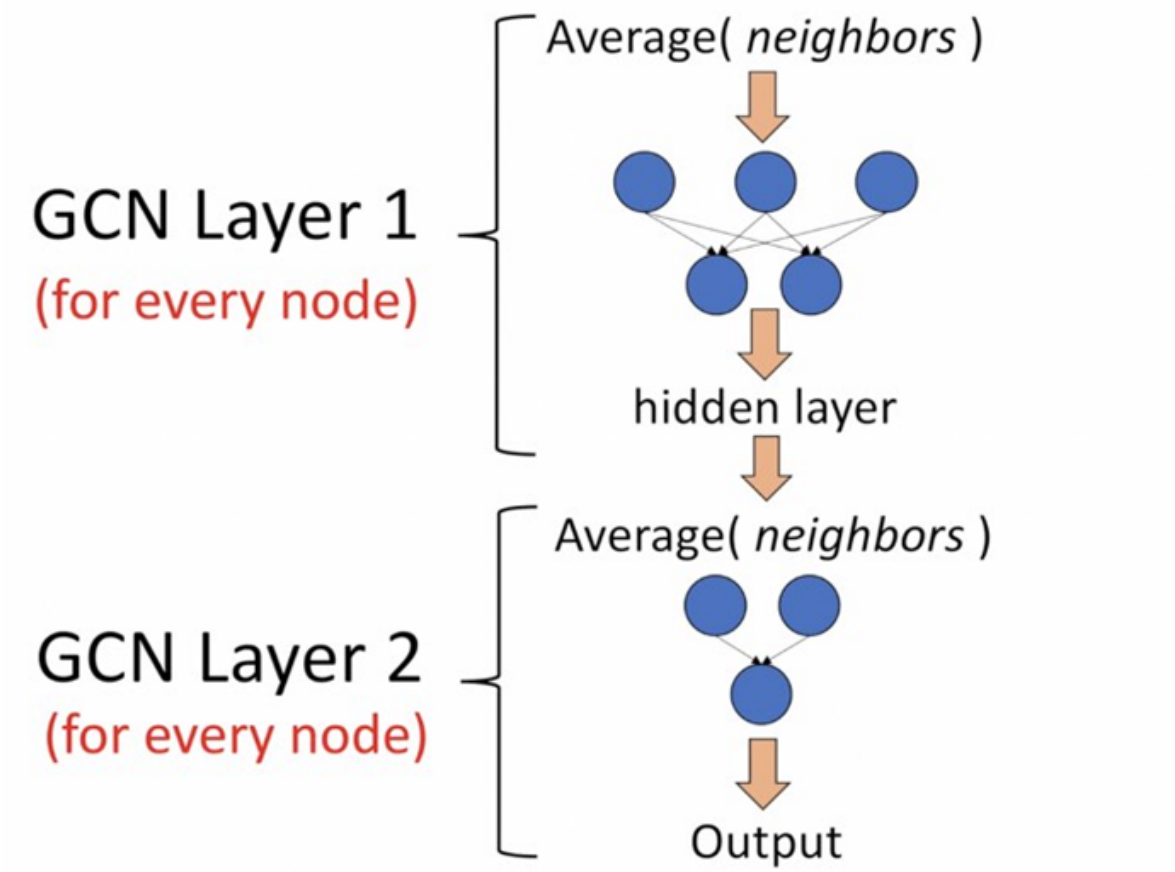


Figure 2.2: 2-layer GCN.

2.2.2 Convolutional neural network (CNN)

Convolutional Neural Networks are a type of artificial neural network that are primarily used for image recognition and classification. They are specifically designed to handle data with a grid-like topology, such as images and videos. Here is a brief of how CNN works: A filter is applied or a feature detector to the input to get the feature

maps. filters help in getting different features that exist in an image input, for example, edges, vertical and horizontal lines. Then, pooling is applied to the feature maps for invariance to translation, when the input is changed the output will not change, this is in which concept the pooling is related, then we flatten inputs to a deep neural network to output the class of the object, neural networks act as a black box, we pass input and results are returned from the model[5]. The CNN architecture is shown in the figure below.

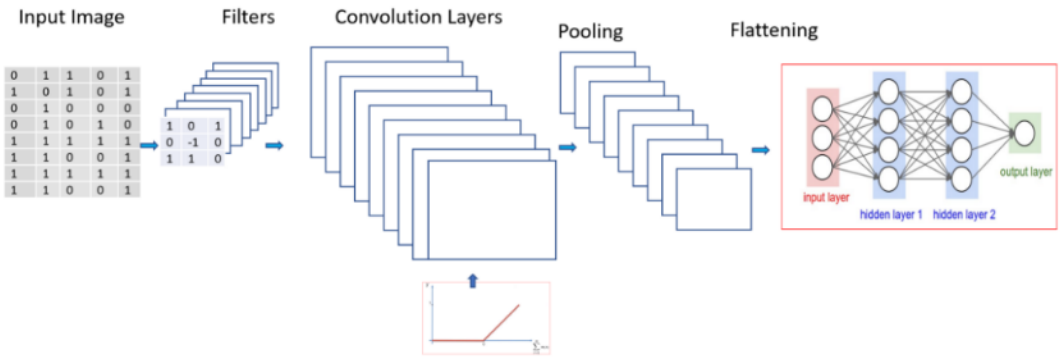


Figure 2.3: CNN Architecture.

2.2.3 exercise classification

Exercise classification is an action recognition task for identifying different types of physical activities performed by people.Various modalities can be used for this task, including RGB video, depth video, skeleton data, and wearable sensors. Deep learning techniques have been widely used for exercise classification in recent years, especially convolutional neural networks (CNNs).One line of research in CNN-based exercise classification employs 2D-CNN techniques that first model the skeletal sequence as a pseudo picture based on manually specified transformation. The heatmaps are then aggregated along the temporal dimension into a 2D input with color encodings or learned modules. However, these approaches suffer from information loss during the aggregation process, which leads to inferior recognition performance. Other research[3, 2] has taken a different approach and directly converted the coordinates in a skeleton sequence to a pseudo picture with transformations, producing a 2D input with the shape K T, where K is the number of joints and T is the temporal length. Such input cannot make use

of convolution networks' locality, making these approaches less competitive than GCN on prominent benchmarks. The figure below shows 2d and 3d convolution. Some previous works have used 3D-CNNs for skeleton-based action recognition[2]. PoseConv3D is another previous work that is 3D-CNN-based technique for skeleton-based action recognition which takes 3D heatmap volumes as input, with these light-weighted 3D-ConvNets and compact 3D heatmap volumes as input PoseConv3D outperforms GCN in multiple contexts in terms of accuracy while improving robustness, scalability and interoperability.

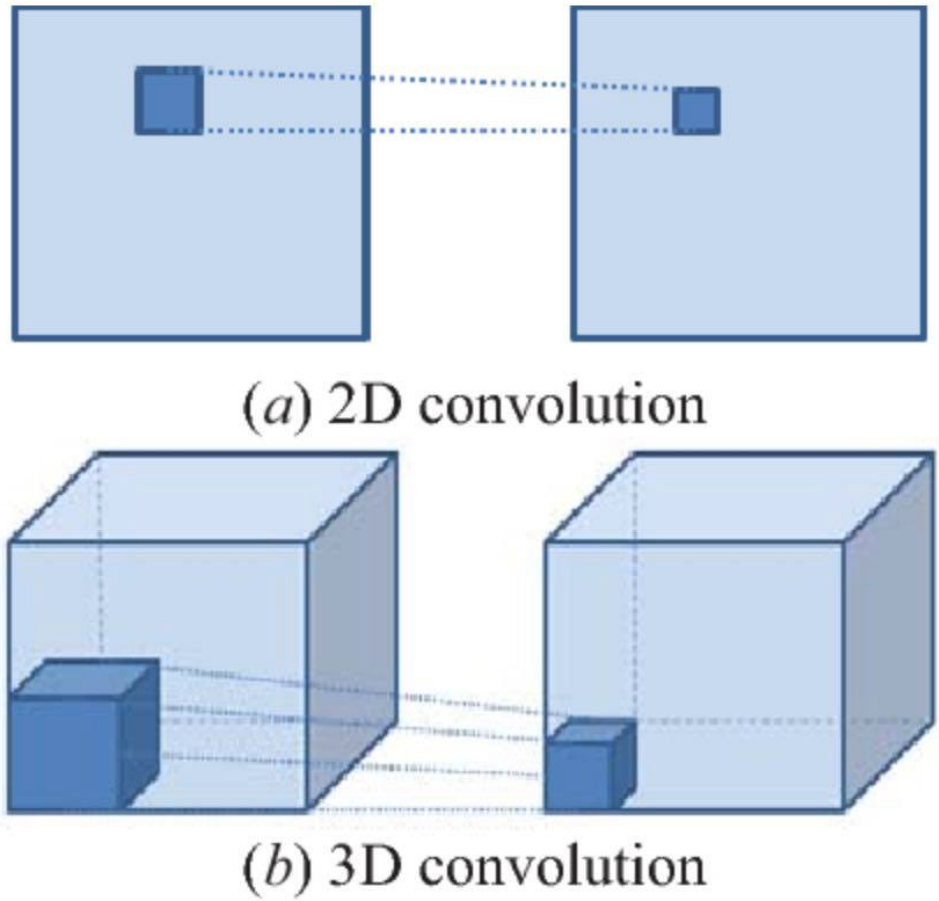


Figure 2.4: 2d convolution vs 3d convolution.

2.3 Open Pose

Open pose is a human pose estimation model. Human pose estimation is a computer vision technique used to detect and locate the positions of a person's body joints and parts in an image or video. This involves identifying the key points of the body, such as the head, torso, arms, and legs, and estimating their positions and orientations in the image as shown in the figure below [7].

Figure 2.5: Human pose estimation.

2.3.1 Open Pose Background

There are two types of body pose estimation: top-down approach, and a bottom-up approach[12]. In the top-down approach, first, we have to identify the people. Then we need to point out the key points for each one of them. Meanwhile in the bottom-up approach, we first identify all the key points for all people and then every two nodes make a pair (The point at the neck and the point at the shoulder). Then we connect each point to the points adjacent to it. After completing this, we have something similar to a graph. After that, we convert the graph into matrices and do some arithmetic operations on it so that we can distinguish the lines that represent bones from others and we erase the lines not necessary. Then collect them to form the shape of the skeleton of the body. The Open Pose paper shows us that the second approaches are

better than the first[12] . The adoption of the speed of implementation of the first approach on the number of people in the picture is the biggest obstacle that the second approach solved. Open Pose is considered a bottom-up approach where the network detects the body parts and key features of the image, then maps those features to form pair [1].

2.3.2 OpenPose Architecture

Open pose uses Convolutional Neural Networks (CNNs) as its primary architecture. CNNs are a type of neural network that are particularly effective at processing and analyzing image data because of their ability to learn and recognize patterns and features within images. By using a CNN, Open Pose can extract patterns and representations from the input data, allowing it to process and analyze visual data effectively.[9]

The specific CNN used in Open pose is VGG-19 [4], which is a deep CNN that has been trained on a large dataset of images and has proven to be effective at image classification tasks. By using VGG-19 as part of its architecture, Open pose is able to benefit from the capabilities of this pre-trained model and apply it to the task of analyzing visual data[9].

VGG-19 output enters two branches of convolutional networks (as shown in Figure 2.7). A set of confidence maps of different and multiple body parts' locations is predicted on the first branch. While a set of Part Affinity Fields (PAFs) are predicted on the second (bottom) branch, which indicates the degree of association between parts. Confidence maps and Part Affinity Fields are shown in Figure 2.6.

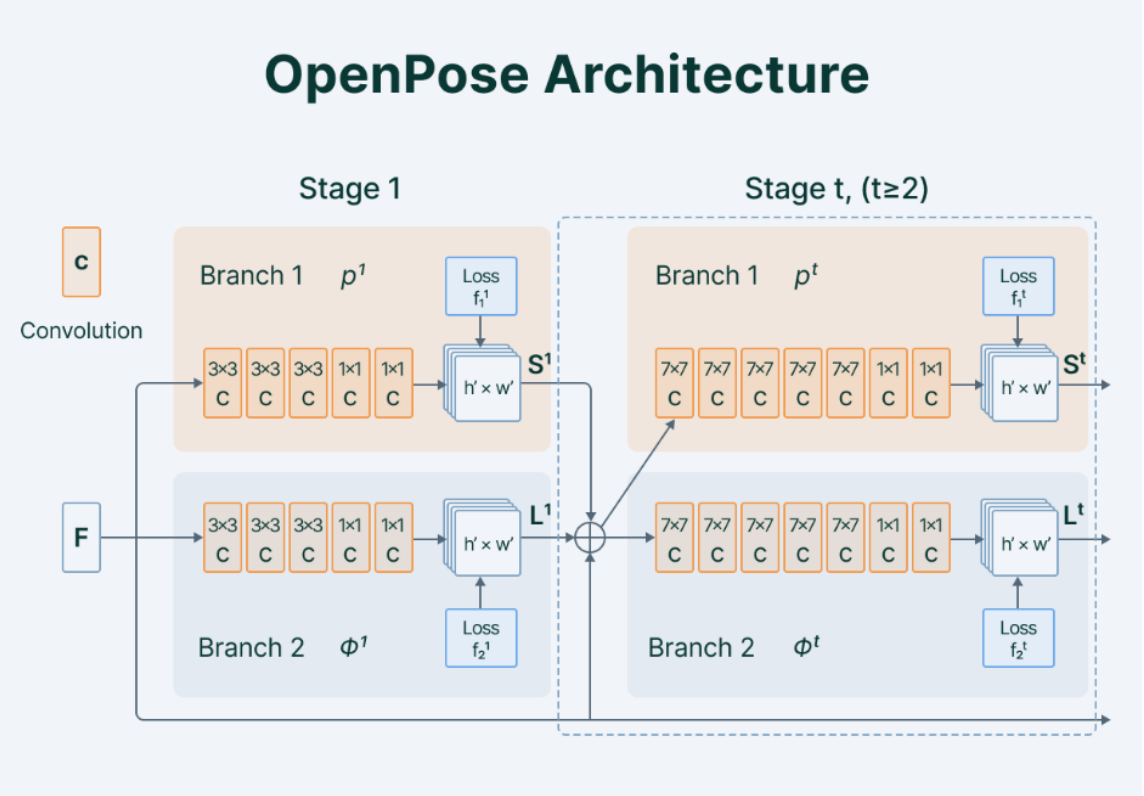


Figure 2.6: Architecture of the two-branch multi-stage CNN.

The Confidence Maps and Part Affinity Fields are processed by a greedy algorithm to obtain the poses for each person in the image. Confidence Maps represent the belief that a given body part can be located in any given pixel while Part Affinity Field (PAF) encodes an unstructured pairwise relationship between body parts with high degrees of freedom[17]. In Stage 1 (left half of the above figure), the network generates an initial set of detection confidence maps (S) and part affinity fields(L).

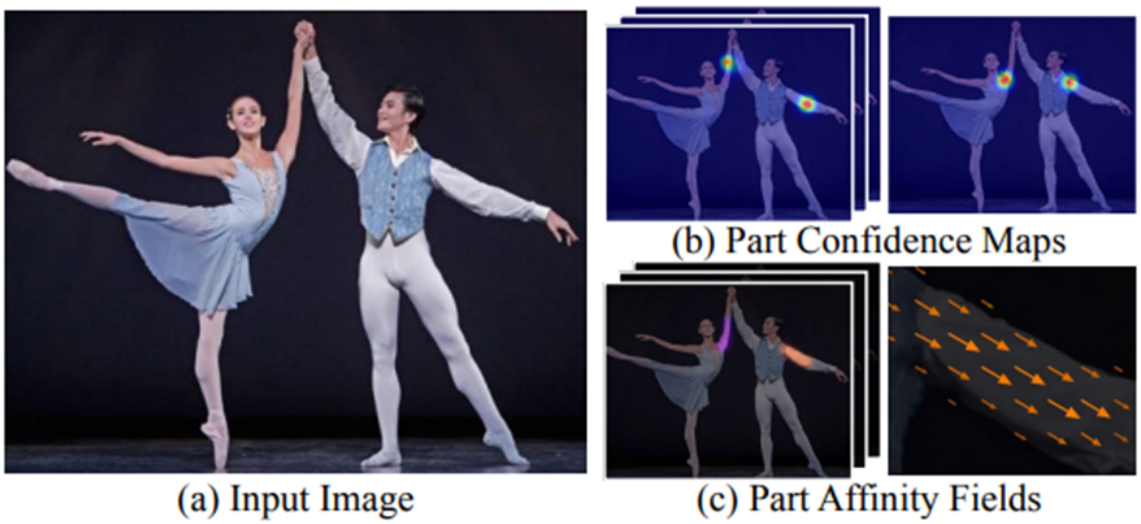


Figure 2.7: Overall Pipeline of the Open Pose architecture.

In each subsequent stage, the predictions from both branches in the previous stage are combined with the original image features F. (Represented by the (+ sign) as in Figure 2.6) then, using these predictions, more refined predictions are produced.

2.3.3 Open pose Main Functionality

There are three modules within Open Pose: body and foot detection, hand detection, and face detection. 25-keypoint body and foot key points are estimated, including 6-foot key points as shown in Figure 2.8 below. The input can be an image, webcam, video, IP camera. . . etc. While the output is an image with key points displayed and saved as png, avi, Jpg..etc. Key Points can be saved as JSON, XML, or YML. We chose to store the output as JSON files.

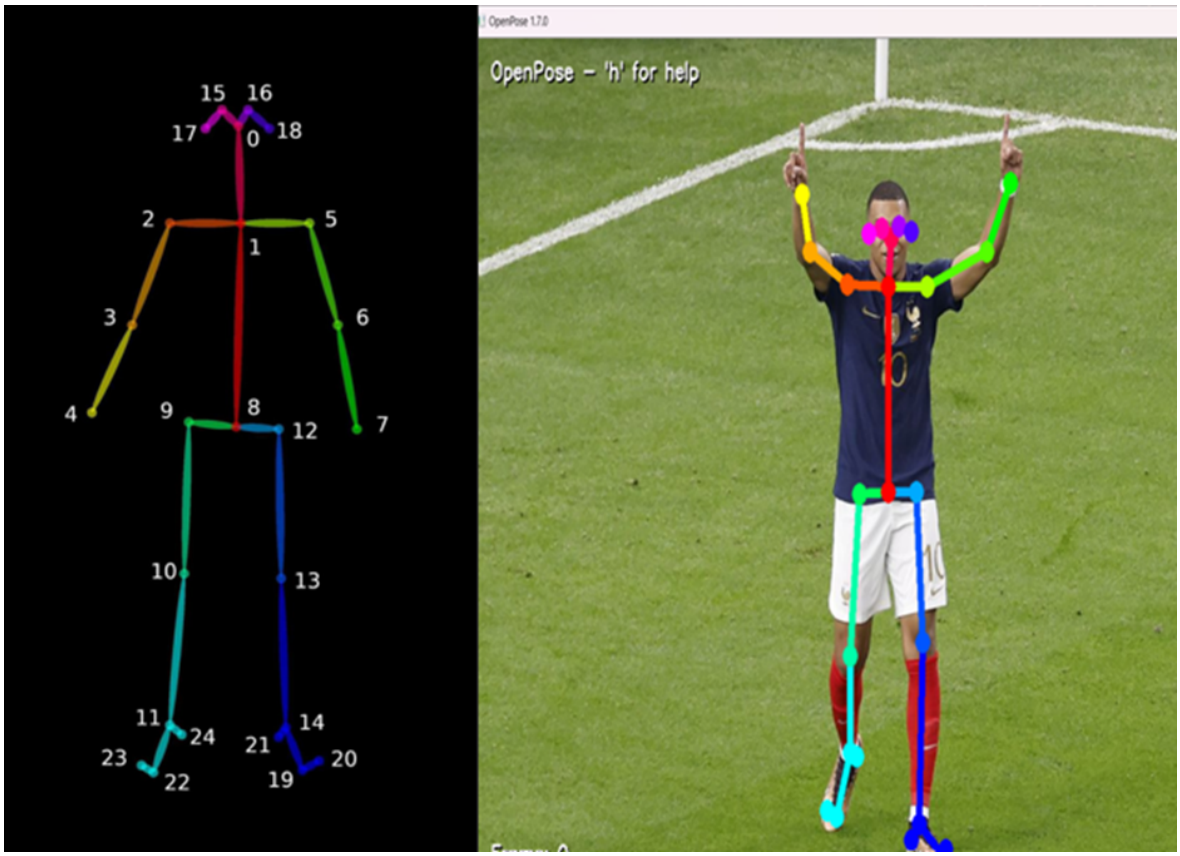


Figure 2.8: Open pose example.

Chapter 3

System Design and Implementation

3.1 System Design:

Initially, we assumed that the input we received was intended for an individual engaged in physical exercise, then the output is a class label that describes and gives feedback about the video since the input is a recorded videos as shown in Figure 3.1. Also, many person's poses will be determined, using 25 body points, for each frame. Each body point can be estimated using the open pose and using a pre-trained convolutional neural network, as we get the time series, Dynamic Time Warping (DTW) algorithm will be employed then to compare the time series. In this algorithm we fill up a matrix, and each cell is considered as a function that relates to the other neighboring cells. In the below figure it can be noticed that the system takes the input then converts it to time series, then compares it with another time series from the labeled videos using Dynamic Time Warping and the label of the closest-matching known time-series will be used to classify the unknown video.

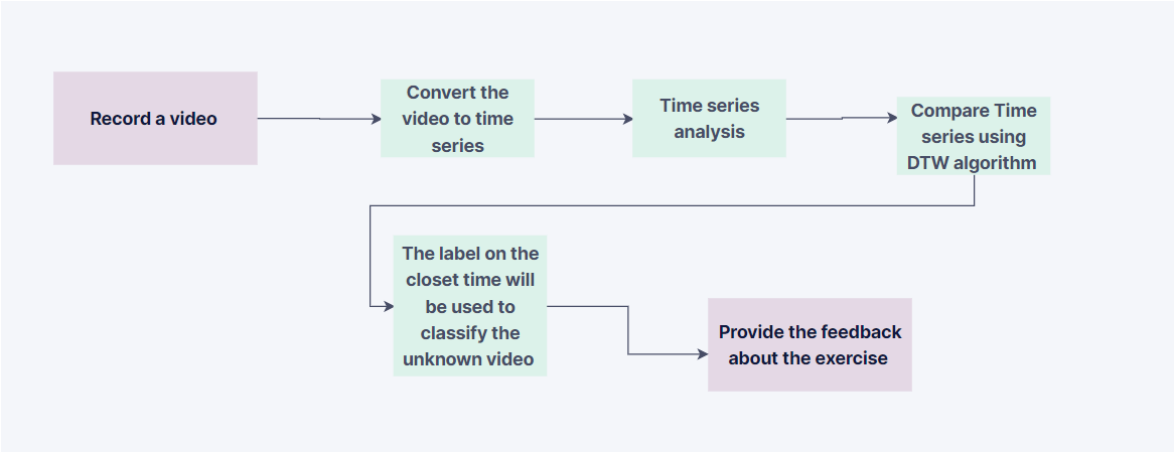


Figure 3.1: Block Diagram for the system design

3.2 Exercice Classification :

Classification is known as the process of identifying and assigning individual quantities to groups or sets. It is used to predict the labels of test data points after training sample data [6]. In this project, we will classify exercises to identify what exercise is being performed. The system takes the videos as input data then transforms it into a sequence of skeletons which will be then processed so the system can classify the exercises. Skeletons simplify the estimation process by reducing the number of parameters required to represent a pose, making it computationally efficient and easier to interpret the results. Additionally, skeletons are invariant to variations in body shape, clothing, and other factors, making them a robust representation for human pose estimation. The figure below shows the steps used in classifying an exercise:

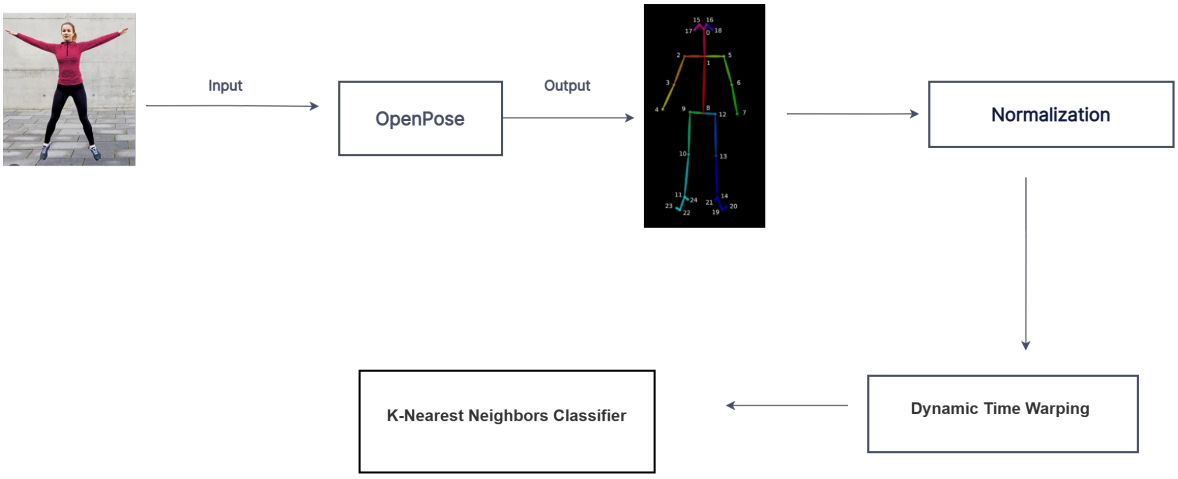


Figure 3.2: Classification Pipeline

3.2.1 Data normalization :

The normalization part was an important part of our project since we faced many problems and we got many errors and inaccurate results which were solved by the normalization. The object deformation has a huge impact on the accuracy of the detection at all types of video monitoring analysis. For example the skeleton, when it is close to the camera, the skeleton will take a large pixel area, but if the skeleton is away from the camera position it will take a smaller area of the pixels. Clearly, the person's position caused interference in the debriefing of the characteristic information and for the analysis based on the skeleton. Therefore, the accuracy of the video monitoring analysis will be affected .

Relying on the output of open pose directly is sub optimal. For example, the coordinates of the output skeleton depend on the person location with respect to the image (to the left or to the right). However, if the person perform the same exercise, the system should always predict the same label no matter where it is performed. In other word, the system should be shift invariant. Furthermore, the system should be scale invariant. For instance if both a man and a child perform the same exercise, their skeleton representations generated by open pose would be different, nonetheless, the system should predict the same label. the result must be the same so that we can from knowing what exercise he is doing and how we will give him feedback, to solve this problem we did the coordinates normalization which give the same coordinates for

the skeleton what ever the position of it or the scale of it since we took a body point number 1 (neck point, because it comes in the middle of the body almost crosswise), as an origin (0,0) point, we did this by subtracting it from all the other points. For Example: point 5 - point 1 (origin) And so on for the rest of the points.

Thus, we have canceled the effect of shifting to the right or left. In order to eliminate the effect of zoom in or out (the scaling of the skeleton), we divided all the points by the length of the bone that connects between the neck point (point 1) and the mid-hip point (point 8), which we obtained its length by means of the Euclidean distance law between two points.

$$dist((x, y), (a, b)) = \sqrt{(x - a)^2 + (y - b)^2} \tag{1}$$

The two points which represents the spine because it is the bone that mediates the body as the Figure 3.3 below shows, and we did this because the ratio between the length of the bones, whether it is the person close to the camera or far away are fixed proportions, we applied the same previous steps exactly to the x and y coordinates thus, we have achieved data normalization, so that we can use the data again with the following steps of the project whenever the body moves the same result will be taken.

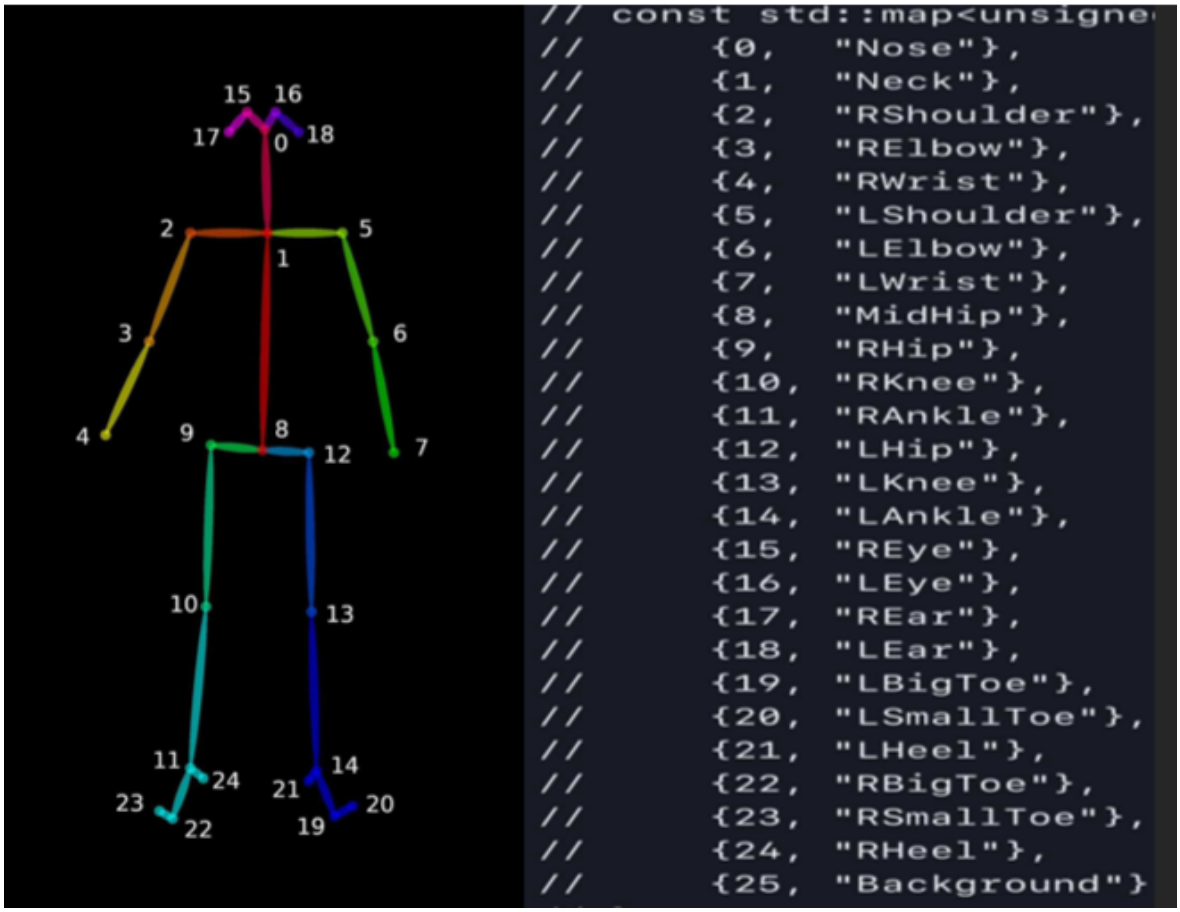


Figure 3.3: The 25- body points with their names

3.2.2 Dynamic time warping (DTW):

Dynamic warping algorithm is used to measure the similarity or the distance between two time series with different lengths. In the general cases when we compare two series together we match them or compare them by one-to-one matches but when there is different lengths, we cannot match like this, from here the idea of using DTW algorithm came, to use the one-to-many and many-to-one matches, which minimized the distance between the two series[16].

The measured distance between the two-time series indicates the similarity between them and used to classify them. Usually, the Euclidean distance measurements used between two time series which equal the sum of the squared distances from each nth point in one time series to the nth point in another time series, but if one of the time series is shifted slightly at time axis while its identical from the other time series, the Euclidean distance may consider that the two-time series are different. From this idea

the DTW was introduced to solve this issue and to give intuitive distance measurements between two time series[10].

Here is an example explaining how DTW works and its effects on the time series with different lengths.

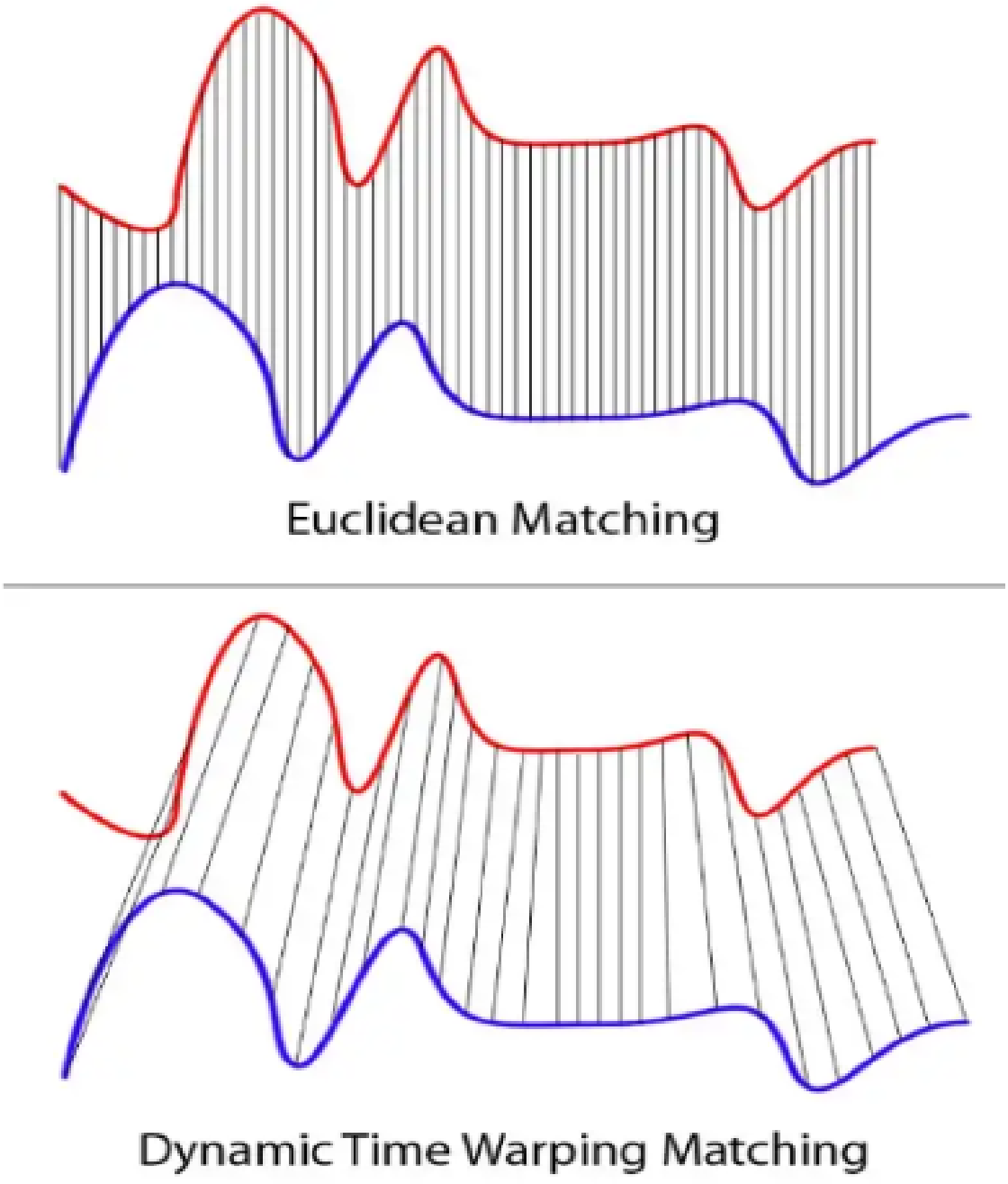


Figure 3.4: Dynamic Time Warping.

As shown in the figure above, the blue pattern is longer than the red one, since if

we applied one-to-one matching, the mapping will not be perfect, some point in the blue curve will be left out so this happens when we use one-to-one matching. On the other hand, as we see the Dynamic Time Warping solved the problem of the left-out points, by developing one-to-many match or many-to-one match by this way there will not be a left-out point for both the red and the blue curves[16]. The implementation of the algorithm:

```
int  DTWDistance (s: array [1..n], t: array [1..m]) {
    DTW:= array [ 0..n, 0..m]
    for i:=1 to n
        for j:=1 to m
            DTW [i, j]:= infinity
    DTW[0, 0 ]:=0
    for i:=1 to n
        for j:=1 to m
            cost :=d(s[i],t[j])
            DTW[i,j]:= cost + minimum (DTW[ i-1, j] ,// insertion
                                       DTW[ i, j-1],//deletion
                                       DTW[i-1, j-1])// match
    return DTW [n, m]
}
```

So, we have two signals x and y signal which we will align these signals by constructing a cost matrix which its rows equal the number of points in x+1 and its column equals the number of points in y +1, by this matrix we will know which points are correspond to each other, the first column and the first raw in the matrix will be infinity, the point (0,0) will be zero, then we will compare the points based on the conditions in the DTW algorithm [4]. In our project we used the DTW to get the distance between the sequences of the skeleton, to do the classification to classify each video or skeleton to its most similar one. So, we calculated the minimum distance and the alignment path by using the DTW, the minimum distance is the point at the last column and raw, by this we find the most similar video, which has the minimum distance between the two videos, then the path was calculated by DTW also, by finding the minimal

cost in the matrix or the cheapest path. Because DTW tell us which path is better, we will use the path when we give the feedback, the path tell us what are the images that we should compare with each other, For example, if the pair (5,7) is one of the path pairs, this means that we have to compare frame No. 5 of the first video with frame No. 7 of the second video, and so on for the rest of the pairs in the path. In the DTW we implement a function to compare a skeletons , This function takes the Euclidean distance between each point in the first skeleton and its corresponding point in the second skeleton (for example, we find the distance between point 7 from the first skeleton and point 7 from the second skeleton and so on) and then takes the average of those distances, which represent the distance between the two files. the average of the distance for all files, before we took the average, we tried the max and the min, then we found that the average gives us more sensible results.

3.2.3 Classification Using KNN:

Classification is known as the process of identifying and assigning individual quantities to groups or sets. It is used to predict the labels of test data points after training sample data as in Figure 3.5 [7]. The K-Nearest Neighbor (KNN) is one of the most popular methods to classify the data set. It has been widely used in machine learning and data mining because it is simple but still very useful with distinguished performance. The K-Nearest Neighbor algorithm is a non-parametric method for classification and regression that falls under the category of lazy learning. The neighbors are picked up from a set of objects or objects having same properties or value[7].

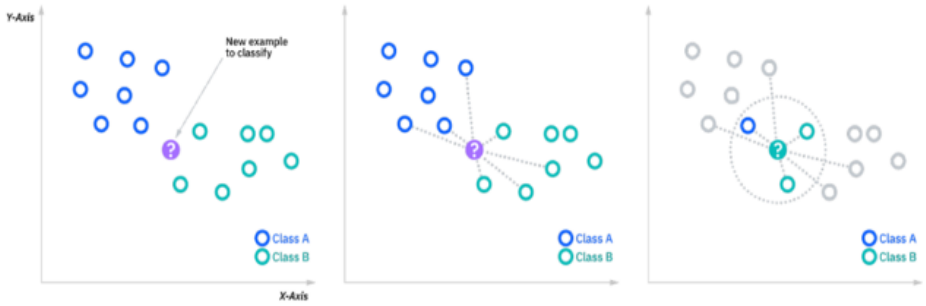


Figure 3.5: Classification using KNN.

The key idea behind KNN classification is that similar data points are more likely to belong to the same class. Therefore, the algorithm calculates the distance between the test point and all the training points and then selects the K points that are closest to the test point [6]. Several different distance measures can be used for KNN classification, including Manhattan distance, Euclidean distance and Minkowski distance. Distance measures are chosen according to the characteristics of the data set and classification task [8]. The distances are determined by DTW using Euclidean distance, and they are then stored in a matrix called the distance matrix. Hence, DTW was employed to measure the separation between each video's most similar video and the series of skeletons. The figure below shows the working steps of KNN algorithm:

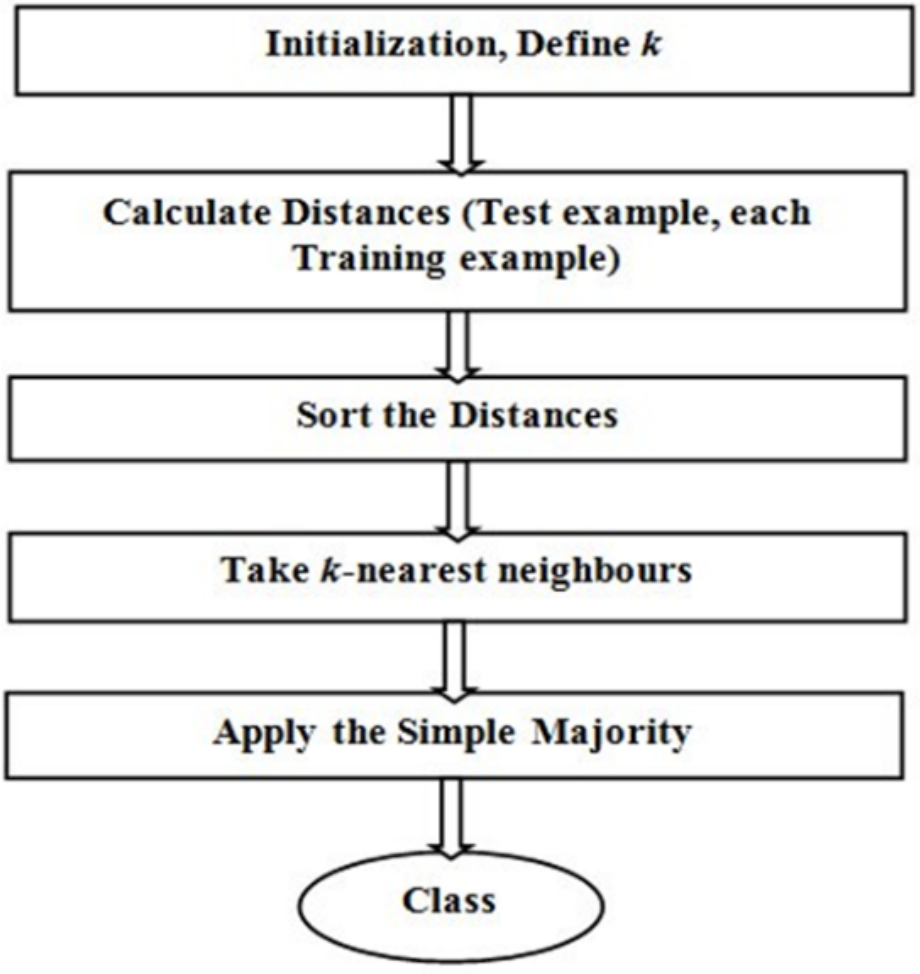


Figure 3.6: KNN Classification Steps.

3.2.4 BaryCenter:

The barycenter (also known as the centroid) is a point in mathematics that denotes the average position of a group of points. It can be viewed as the system's center of gravity or mass. When describing an object's position with other objects in space, the barycenter is a common term used in geometry, physics, and astronomy. In our project, we used barycenter to condense the motions seen on video. Barycenter is used to compute distance with the average video for each exercise instead of computing distance with all videos. Once these spots have been located, we can use the DTW algorithm to determine how far apart they are from the barycenter. The average of all of these distances is then used to get the barycenter. Overall, analyzing and evaluating the execution of workouts can be done by using the barycenter to compile the movements seen in a film. We can feedback to help the person's technique by pointing out these patterns or movements. Also, barycenter was a good technique to make our system faster than before.

3.3 Feedback

The system is designed to provide feedback to users on their exercise performance . After classifying the exercise, the system compares the Barycenter training video with a video of the user performing the same exercise. The system finds the alignment path using the DTW and compares the frames of the two videos and identify to differences between them. It then calculates the mean and standard deviation of these differences according to the following equation:

$$Upperoutlierlimit = Mean + (3 * standarddeviation) \tag{2}$$

Then identifies any outliers that fall outside of three standard deviations from the mean. The system uses these outliers to generate feedback for the user by highlighting areas of the video where their exercise performance needs improvement. The feedback is provided visually, by highlighting specific parts of the video. Overall, the feedback system provides an automated and objective way for users to improve their exercise performance and technique, which can help prevent injuries and optimize their workout routine. The following figure shows a person , trying to do the jumping jacks incorrectly

by not moving his legs in the right way, so the system highlighted where he is doing the exercise wrong with red dots.

Figure 3.7: Feedback.

Chapter 4

Experiment and Results

4.1 Data set

One of the challenges that we faced is finding the data needed for this project, the data is very important in our project because we will use it in training and testing the machine learning, so we had to solve the problem of finding data, at first we tried to obtain ready-made data from the Internet, but as expected, unfortunately, we did not get anything useful from the Internet, so we decided to create the necessary data for us by ourselves by doing these exercises and sending it to the algorithm to be trained on, but we realized that we need a large amount of data that is not we can configure them by ourselves, so the solution was to take the exercise videos from YouTube by searching for someone who does these exercises in the YouTube videos, and then we collect them and make the algorithm train on them, and in this way we have obtained the necessary data for us in the project. We collected 15 different videos for training, 5 for each exercise (Jumping Jacks , Squats and Jogging) . We also collected 10 Videos for testing. Since we couldn't find enough videos and the data set wasn't large, we used simple classification model(KNN).

4.2 Evaluation metrics

In our analysis, we will incorporate commonly used metrics including Accuracy, precision and recall. Accuracy refers to the overall number of times the model was correct, while precision measures how well the model can predict a particular category. On the other hand, recall indicates the number of times the model was successful in detecting a specific category. The calculation of Accuracy, Precision, and Recall in the confusion matrix is demonstrated in Figure 4.1.

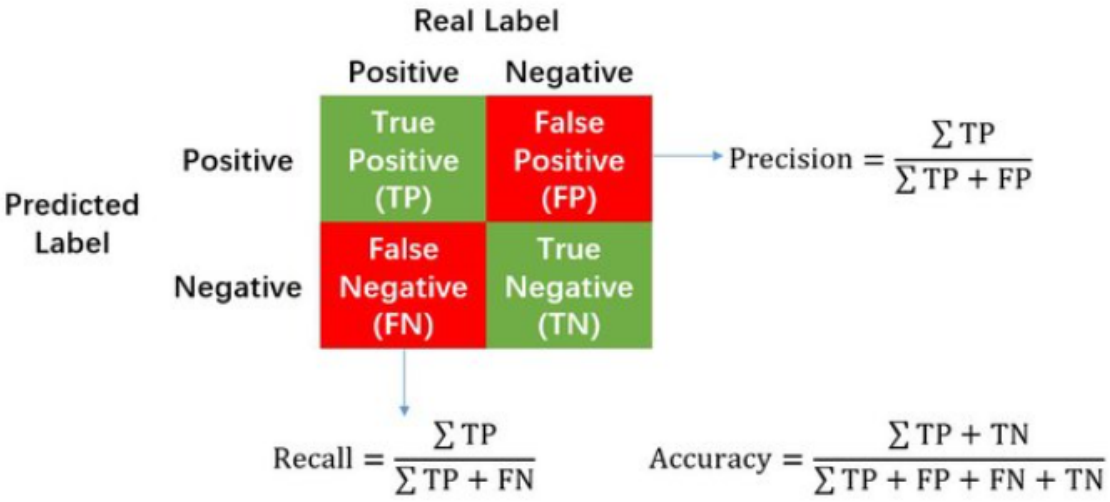
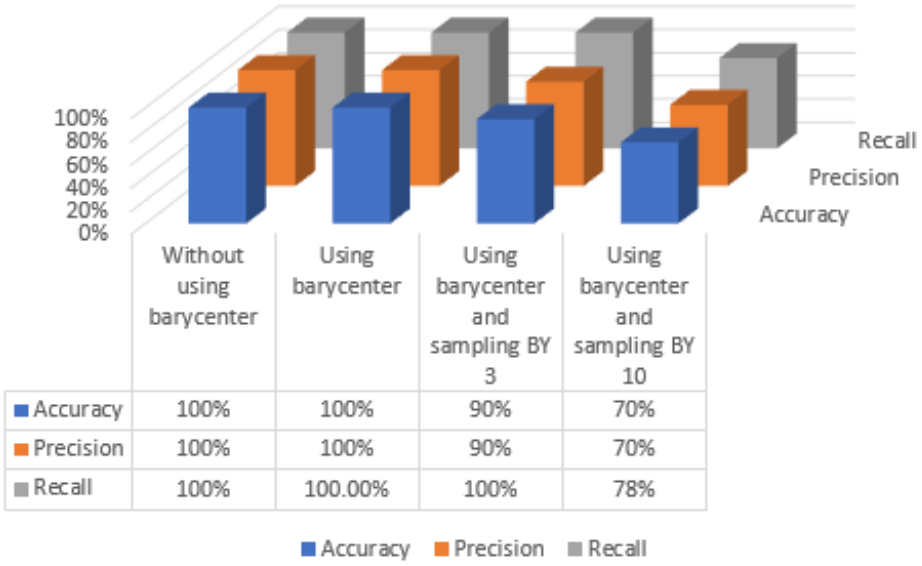


Figure 4.1: Accuracy,Recall and Precision equations .

Figure 4.2 below shows the accuracy, recall and precision results. It can be noticed that the results are best using barycenter only, as the accuracy , precision and recall are all 100 %. When using barycenter and sampling by 3 , the accuracy and precision decreased to 90%, while the recall was still 100%. Finally, when the using the barycenter and sampling by 10, , the accuracy and precision decreased to 70% while the recall became 78%. It can be noticed that the more we sample, the more the metrics results decrease.



	Without using barycenter	Using barycenter	Using barycenter and sampling BY 3	Using barycenter and sampling BY 10
Accuracy	100%	100%	90%	70%
Precision	100%	100%	90%	70%
Recall	100%	100.00%	100%	78%

Figure 4.2: Accuracy,Recall and Precision results .

4.3 Experiment results

To ensure that all processes are operating effectively and any issues related to normalization, specifically shifts and scaling, have been resolved, we conducted experiments on four images. This involved testing the original image without any modifications, an enlarged version of the original image, a shifted version of the original image, and an image of the same individual executing a different movement, as depicted in the accompanying images. We computed the distance between these images and notice that the normalization eliminate any variations caused by scaling or shifting.

Figure 4.3: Original image.

Figure 4.4: zoomed out image.

Figure 4.5: Shifted image.

Figure 4.6: Different movement.

The results of the coordinates normalization were as follows:

Table 4.1: The results of the coordinates normalization

Modifications	Distance
Zoomed-out	0.2711
Shifted	0.360
Different Movement	7.869

Through the previous results, we can notice that there is no significant difference between the original image and the zoomed and shifted images in their both distances , while there is a noticeable difference between the original image and the different image, which means that the data normalization results are acceptable and even excellent.

In statistics and machine learning, sampling is a typical strategy for estimating features of a population or dataset through the examination of a smaller subset, or sample, of that population or dataset. Instead of analyzing the full dataset, which might be computationally expensive or even impossible in some situations, the notion is that by analyzing a sample, we can make inferences about the wider population.

Instead than processing every video at once, sampling can be used in the context of video processing to evaluate a portion of a huge collection of videos. This enables us to accomplish objectives more quickly and effectively, which can be especially valuable in circumstances where processing resources are constrained . The results of the running time after we used the sampling and barycenter are in the below table.

Table 4.2: Running Time Values

The technology	Runing Time
Without using barycenter	60 minutes
Using barycenter	15 minutes
Using barycenter and sampling	1 minute

After completing the normalization stage, DTW was tested on a couple of videos, and the videos where the same exercises are performed were classified together.

Figure 4.7: Alignment results obtained using DTW.

Chapter 5

Conclusion And Future Work

5.1 Conclusion

In conclusion, we proposed a system that will help users to work out at home and perform the exercises correctly. The system asks the users to open their cameras as input and analyze this input stream, to ensure that they are doing the right exercise. It will detect the user's body parts or joint position using the technique of Human pose estimation (Open Pose) which uses Convolutional neural networks (CNN) as its main architecture. Then the system will use the DTW classification and give feedback for the user .

5.2 Future Work

This project has various aspects to be completed in the future, such as increasing the number of exercises supported by the system so that it gradually becomes a comprehensive system and the gym can be completely replaced by it, and we can also make the system support more than one user at the same time in parallel where it can measure the movement of each of them separately and give them feedback separately, and in the future we can change the machine learning algorithm used in the event that another suitable and better algorithm appears, which will increase the speed and accuracy of

the system. In the near future, we look forward to completing this project after graduation and applying the above-mentioned improvements and others if possible.

References

[1] Nilesh Barla. "A Comprehensive Guide to Human Pose Estimation". In: (2022). URL: https://www.v7labs.com/blog/human-pose-estimation-guide?fbclid=IwAR2n391j8iYr1g8p5zRkqBhdyg7mJ0ikfJB0XQoPKhGvC2lV1y4Ipcp42Qg.

[2] Jefersson A Dos Santos Carlos Caetano Jessica Sena Franc¸ois Bremond and William Robson Schwartz. "Skelemotion: A new representation of skeleton joint sequences based on motion information for 3d action recognition". In: (2019). URL: https://arxiv.org/abs/1907.13025.

[3] Peng Cui Dingyuan Zhu Ziwei Zhang. "Robust graph convolutional networks against adversarial attacks". In: (2019). URL: https://dblp.org/rec/conf/kdd/ZhuZ0019.html.

[4] TSimonyan et. "Visual Geometry Group 19 Layer CNN". In: (Sept. 2017). URL: https://paperswithcode.com/method/vgg-19.

[5] Amirreza Shaban Hamid Reza Vaezi Joze. "Mmtm: Multimodal transfer module for cnn fusion". In: (2019). URL: https://arxiv.org/abs/1911.08670.

[6] Essoukri Ben Amara Neili Boualia. "Deep Full-Body HPE for Activity Recognition from RGB Frames Only". In: (2021). URL: https://doi.org/10.3390/%20informatics8010002.

[7] Sarah Jane Delany Padraig Cunningham. "k-Nearest neighbour classifiers". In: (2007). URL: https://www.researchgate.net/publication/228686398_k-Nearest_neighbour_classifiers#fullTextFileContent.

[8] Sarah Jane Delany Padraig Cunningham. "k-Nearest neighbour classifiers". In: (2007). URL: https://www.researchgate.net/publication/228686398_k-Nearest_neighbour_classifiers#fullTextFileContent.

[9] SkillsYouNeed. “The Importance of exercise”. In: (2020). URL: https://www.skillsyouneed.com/ps/exercise.html.

[10] Philip Chan Stan Salvador. “FastDTW: Toward Accurate Dynamic Time Warping in Linear Time and Space ”. In: (2019). URL: https://cs.fit.edu/~pkc/papers/tdm04.pdf.

[11] Inggez startup. “Inggez”. In: (2018). URL: https://fit.inggez.com/.

[12] Shih-En Wei Tomas Simon and Yaser Sheikh. “OpenPose: Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields”. In: (May 2019). URL: https://arxiv.org/abs/1812.08008.

[13] Philippe Weinzaepfel Vasileios Choutas. “Potion: Pose motion representation for action recognition”. In: (2018). URL: https://ieeexplore.ieee.org/document/8578832.

[14] Liubov Zatolokina. “ Human Pose Estimation Technology Capabilities and Use Cases”. In: (2022). URL: https://dev.to/liubovzatolokina2022/human-pose-estimation-technology-capabilities-and-use-cases-in-2022-8i6.

[15] Zenia. “Zenia App”. In: (2018). URL: https://www.zenia.app/.

[16] Jeremy Zhang. “Dynamic Time Warping”. In: (Feb. 2020). URL: https://towardsdatascien
com/dynamic-time-warping-3933f25fcdd.

[17] Yaser Sheikh Zhe Cao. “Realtime Multi-person 2D Pose Estimation Using Part Affinity Fields”. In: (2017). URL: https://ieeexplore.ieee.org/document/8099626.