FITNESS TRAINER USING COMPUTER VISION AND MACHINE LEARNING

Avishek Thakur
Department of Computer Science & Engineering, School of Engineering& Technology, *Sharda University, Greater Noida, Uttar Pradesh, India*

Sumeet Dhiman
Department of Computer Science & Engineering, School of Engineering& Technology , *Sharda University, Greater Noida, Uttar Pradesh, India*

Amitesh Kumar
Department of Computer Science & Engineering, School of Engineering& Technology, *Sharda University, Greater Noida, Uttar Pradesh, India*

Sandeep Kumar
Department of Computer Science & Engineering, School of Engineering & Technology, Sharda University
Greater Noida, Uttar Pradesh,India

Abstract—**Physical activity and exercise can have instant and everlasting health benefits. But when it comes to working out, form is the most dominant thing. Poor form places undue intensity on muscles, tendons, and ligaments, leading to strains and sprains. Good form lessen overcompensation and the likelihood of injury.This is one of the many factors why people work under the guidance of a trainer. Whether one wants to develop an individualized program, or simply feel that they would benefit from the additional accountability, a personal trainer can be a great resource. Online coaching and virtual training have emerged as a staplein the fitness industry. Because of technology improvements, an appropriate application may provide continual reminders and the much-needed motivation to focus more on exercise and diet. AI has extended itsroots in practically every functional area of business over the years. Pose estimation is one of the most prominent AI solutions; it is used to calculate the position and orientation of the human body given a picture of a person. Our objective is to build an automated fitness coach system that can do everything a physical personal trainer does. The program collects motion data from users through a camera and then uses human posture estimation, repetition counting, and formevaluation via voice-based real-time feedback.**

Keywords—Artificial Intelligence, fitness, form evaluation, pose estimation.

I. INTRODUCTION

Exercise and physical activity may be beneficial to one's health both now and in the future. Adults should strive for 150 minutes of varied physical exercise each week. Stretching, aerobic, muscle-strengthening, and bone-staining practises are just a few that assist the body in numerous ways. Physically active people are less likely to experience depression or cognitive decline as they age. Exercise lowers the chance of getting a number of illnesses, including cancer, diabetes, and heart disease. Most gyms include a choice of exercise machines as well as trainers who may advise us on the activity and proper form. However, the lack of the aforementioned equipment and trainers may be a significant barrier to us exercising at home.

Our objective is to build an AI-based trainer that will improve the effectiveness of your at-home workouts. The project's objective is to create an AI system that will calculate the quantity and quality of repetitions using CPU- based posture estimation to help you workout. This initiative attempts to make exercising more convenient and pleasant. The user interface will not be distracting. We will look at an overview of the contributions of different families, their algorithms, advantages, disadvantages, and efficiency in relation to other technologies now in use, applications, and prospective future work.

Due to backdrop clutter, scenario changes, item appearance changes, etc., visual tracking of non-rigid objects in complicated scenarios is a challenging challenge. Yilmaz provides an analysis on the subject. However, tracking humans differs from tracking other non-rigid objects. One aspect of the human body is flexible, allowing for changes in posture at a glance. However, the look of regional body components, such the torso, hand, and arm, has a hard quality.Therefore, modeling the global non-rigid appearance, with consideration of local rigid body parts, is areasonable solution for human tracking.

World Health Organization (WHO) has listed 10 threats to global health, which are: (i) Air pollution and climate change, (ii) Non-communicable diseases (NCD), (iii) Threat of a global influenza pandemic, (iv) Fragile and vulnerable settings, such as regions affected by drought and conflict, (v) Antimicrobial resistance, (vi) Ebola and high-threat pathogens, (vii) Weak primary care, (viii) Vaccine hesitancy, (ix) Dengue and (x) HIV. Regular cigarette use, physical inactivity, hazardous alcohol use, and poor diets are the main risk factors for NCD. With an estimated 38 million deaths per year globally from NCDs, which mostly affect poor nations, they are the worst diseases. Obesity contributed to two-thirds of the primary causes of non- communicable illness mortality cases globally. People with excessive weight have a higher % chance of developingheart disease, diabetes, and other ailments. A person's excessive weight may be brought down to a healthy level byengaging in the right number of physical activities. Regular exercise can help maintain good health and ward off chronicconditions. School students are usually encouraged to work on their fitness to promote healthy weight loss behaviors. Teaching physical education at the school level may still not be enough to provide the required fitness-related information, especially on the type of exercises that studentsneed to perform. One of the methods to introduce and teach fitness activities to the students is by using a fitness instructor or trainer. A good fitness trainer program also can contribute to the effectiveness of physical education in the educational institution.

Every pupil needs a customised trainer, which takes timeand money to provide. Artificial intelligence (AI) technology may thus be used to speed up the customizing process by determining the best exercise regimen for a certain student's demands or preferences. This study presented an objective discussion of the usage of AI technology to select an appropriate virtual fitness trainer based on user-submitted criteria.The majority of gyms include a selection of workout machines as well as trainers that may instruct us on the activity and the proper form.But the unavailability of the equipment and trainers can be an important reason that can stop us from doing exercise at home. We aim to build an AI-based trainer that would help you exercise more efficiently in your own homes. The project focuses on creating an AI algorithm to help you exercise, by determining the quality and quantity of repetitions which is done by using pose estimation running on the CPU.

Despite the market having a large number of health and fitness applications, their diversity is still quite constrained. This is why AI Fitness Trainer may be utilized toincorporate physical education into the curriculum.According to a study, establishing customized training plansthat cater the physical exercises to the demands of the individual user can improve a virtual fitness application. A thorough virtual trainer design, in particular, can speed up the process of creating a specialized fitness workout programme. To compute and customize the AI Fitness Trainer in this instance, we employed an AI algorithm.

II. Literature Review

This AI-based workout assistant and fitness guide is intended to help those who don't have access to a gym but are nonetheless motivated to exercise at home to keep their bodies in good shape. to assist them in performing the workouts properly and guard against both immediate and long-term injury. Along with a personalised daily workout calorie count, this also offers a personalised health guide and food plan.Chen, H. K.[9] proposed The programme also lists essential health insurance and policies offered by the Indian government to the general public and uses API and Web services to determine eligibility. The goal was to leverage photographs from the multiperson solution and an efficient single-shot approach to give a bottom-up approach for the activity of estimating the pose of the user and real-time segmentationof the user. They therefore suggested using a CNN, or convolutional neural network, by training it to detect and classify the key points and subsequently give accurate results by analysing the relative displacements and consequently by clustering or identifying the group of various key points and analysing the pose instances. The model obtained a COCO[6] accuracy of the points of 0.665 and 0.687 using multiple level inference and single-scale inference.The training data set consists of 60000 photographs, 25000 frames in which the user really completes the activity, and a few images of the same stance with different critical points. The MediaPipe posture estimation tool employs a 33 key points approach, whereby it finds the key points, uses them appropriately, and then guesses the pose by looking at the data set. Utilizing the blaze pose tool, which uses a machine learning approach to pose detection, it tracks the pose from a real-time camera frame or RGB video.

An AI-based application called an Intelligent Virtual Fitness Trainer (IVFIT) was designed and developed as part of a study that aimed to give school kids a personalised physical education experience. This programme was created to (1) show users five training routines for 30 minutes, teach them three different dance styles for 22 minutes, and show them five fundamental self-defense techniques. These training scenarios—in which the fitness trainer's advice isdetermined using a computation of artificial intelligence algorithms—were given to a physical education class of 23 students. After each activity, the percentage of prediction error was measured to assess how well the app provided recommendations.Mokmin, N. A. M.[2] implemented To find out how students felt about the intended VFT application, a teacher observation test was later carried out. Overall, students felt that the VFT application was successful in encouraging greater levels of physical activity participation without teacher supervision.A tailored virtual trainer can help assign an appropriate virtual trainer for particular users and be utilised for physical education activities, it can be inferred from thestudy. To fulfil the goals of this study, a mobile application was created and posted to the app store. To encourage pupils' learning of various motions, there were five major trainers employed. The percentage of the 23 participants' prediction errors was examined using descriptive analysis (12 female and 11 male). The majority of participants (70%)said that they thought of themselves as being extremelybusy in their daily lives. The remaining participants (30%) said they exercised little to never each day. A 24.5 was the average BMI score.

The objective is to build an automated fitness coach system that can do everything a physical personal trainer does. Using a webcam to capture user motion data, the software then uses human posture estimation with the help of repetition counting and form evaluation through voice-basedreal-time feedback. In order to accurately suggest and keep track of a specific set of workouts to a person, trAIner incorporates artificial intelligence. In the beginning, we will give the enrolled user a questionnaire to assist us gather the necessary data on their medical history. The workouts that are not recommended will be filtered out based on the user's medical history, including injuries and medical issues. Based on the aforementioned elements and the amount of hours and days the person is willing to put in each week, a routine is created. Singh, V., Patade[1] developed the pose detection model, which keeps track of both the form of a specific workout and the number of repetitions being performed, is used to monitor the workouts. A voice-based system provides immediate exercise feedback to let the user know whether or not he is doing the task correctly.trAIner utilises Mediapipe Pose for pose detection, categorization, and repeat counting.BlazePose , a real-time body posture tracker, is used by Mediapipe Pose. For one individual, BlazePose creates 33 body keypoints. The compact, intelligent, and AI-based personal trainer TrAIner is clever. The lightweight pose estimate model will speed up and improve the posture identification process, producing results more quickly. The application will operate in the browser, making it simple to use.

Stacey, D., Hopkins[10] proposed this study which explores methods for providing fitness coaches with evidence-based information. Where do fitness coaches acquire their evidence-based knowledge, specifically? What approaches to conveying evidence-based information are most useful for fitness trainers? What obstacles and enablersexist for fitness trainers using research-based knowledge in their practise? We outline a thorough evaluation of research on knowledge translation initiatives aimed at personal trainers. Fitness instructors were described as those who help the general public with workout programme planning and monitoring.

This study examines novel tracking strategies for kids in films with indoor and outdoor backdrops. It focuses on tracking a whole child-thing while also tracking the physical components of that object in order to produce a successful system. Aljuaid, H.[8] developed this effort which offers a technique for labelling the head, upper, and lower portions of the body before selecting a particular region within each section and tracking it in line with the labelling strategy using a Gaussian mixture model (GMM) algorithm. Three diverse uses of the technology are shown: seated motion, crawling, and child-object walking. During system testing, walking object tracking performedthe best, scoring 91.932% for body-part tracking and 96.235% for whole-object tracking. For body-part tracking and whole-object tracking, crawling object tracking obtained 90.832% and 96.231%, respectively.Crawling object tracking achieved 90.832% for body-part trackingand 96.231% for whole-object tracking. Finally, seated- moving-object tracking achieved 89.7% for body-part tracking and 93.4% for whole-object tracking.

It is a challenging task to track non-rigid objects with considerable form fluctuation in complicated scenarios. Since the human body has good local rigid characteristics, human tracking is a specific instance of this issue.Ioffe, S.[4] developed the innovative human tracking technique we suggest in this research investigates the local stiff characteristics while retaining the overall structure extremely effectively. There are three steps in this process. First, using the patches clustering approach, stiff, structured components are recovered to represent the human body. Then, using a structured constraint approach, the rigid portions are tracked. Finally, global similarity is used to determine the object's ideal approximated state.

A programme was developed to keep an eye on things in a setting that simulates a babysitter's vision with the main goal of developing a reliable and useful moving child-object identification system. Mohamad, D.[8] proposed this study to examines novel tracking strategies for kids in films with indoor and outdoor backdrops. It focuses on tracking a whole child-thing while also tracking the physical components of that object in order to produce a successful system. The method described in this work involves marking the head, upper, and lower regions of the body, identifying a specific area within the three sections, and then following the labelled section usinga Gaussian mixture model (GMM) algorithm .

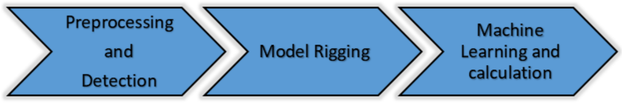
Correia, F[7] proposed the use of virtual reality for both personal and professional goals has expanded thanks to technological improvements. This technology's main objective is to transport users tovirtual worlds. When avatars are used in place of the user's actual body, the user's degrees of presence and embodiment are augmented. It is possible to animate the avatars using inverse kinematics to simulate the user's movements based on data collected from sensors positioned all over their body. These sensors can be found in varying numbers, allowing for different levels of body tracking accuracy.

The utility of tracking human body parts to pick up on nuances in social interactions is discussed in this research. Although many other body-part tracking algorithms have been put forth, we concentrate on particle filtering-based tracking using previous models since it has various benefits for social interaction research. Ukita, N [5] implemented The benefits, drawbacks, and potential qualities of the body-parts tracking utilising previous models are presented with real findings as a first step for collecting subtle information from movies of social interaction activities. Applying this method to the MMDB dataset and attempting to build baby postural prior models in the hopes of extending its use to early detection of ASD patients Look for alternative methods for monitoring bodily component.

Artificial intelligence-powered virtual trainers called AI personal trainers help you reach your fitness objectives. After gathering some information about your body measurements, current level of fitness, fitness goals, and more, the computerised personal trainer may offer you individualised training and diet plans. Artificial intelligence-powered virtual trainers called AI personal trainers help youreach your fitness objectives. After learning a few details about your body, your present level of fitness, your fitness goals, and other things, the computerised personal trainer may give you a customised training and nutrition plan. MediaPipe is a framework for creating cross-platform, multimodal (for example, video, audio, and any time series data) and applied ML pipelines.Ammar Anuar[12] proposed a perception pipeline that may be created using MediaPipe as a network of modular elements, such as media processing functions and inference models (such as TensorFlow and TFLite). Our 3D human position reconstruction demo app was quite easy to design using MediaPipe, enabling quicker neural network inference on the device and synchronising our result presentation withthe video capture stream..

III. Methodology

Start our pycharm project and you can see that we have named it pose estimation project. Have a folder with post videos so if I open this up you can see that we have a lot of different types of videos. Total of nine videos some of these videos are actually slow motion so when we are running it. It will look like it's slow but it's not actually slow it's actually slow motion. I took these videos from pixels.comso we are going to test these videos out and see how well does our pose estimation work.

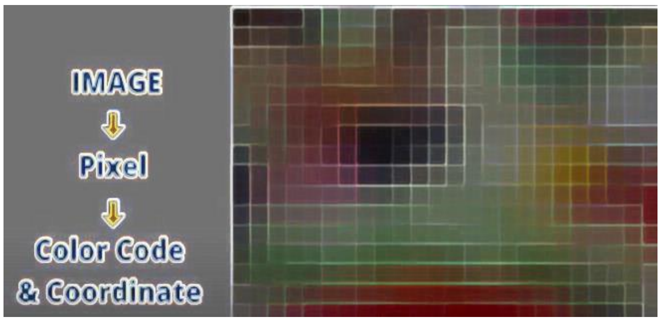


(Fig 1. Roadmap for Dataflow)

We will select all of this or let's create a new file. We will write here pose as the motion minimum so this is the bare minimum code that is required to run it and later on we are going to see how we can create a module out of this so that we don't have to write the code again and again for a lot of different projects.

Start by importing cv2 and then importing media pipe as mp. But now you can see that we get an error this is becausewe did not include these packages in our project. Include those so go to file settings and go to our project interpreter and open that up and here write opencv dash python. Install and then media pipe. Now both the packages have been installed and go back and you can see the error is now gone so open cv is the library that we will be using for image processing and media pipe is the framework. That will allowus to get our post estimation. So now the first thing we will do we will read our video. So we will write cap is equals to cv2 dot video capture and simply give in our video number so here we will write pose videos and we will write video number one dot mp4. Later we can change it if we want and then write while true. Write success and image is equals to cap dot read so that will give image. Write cv2 dot in show and write image and then write cv2 dot weight key and Write one so that we get a one millisecond delay. Run this and see if it works and there we have our video went quite fast so the frame rate is actually quite high at this point.

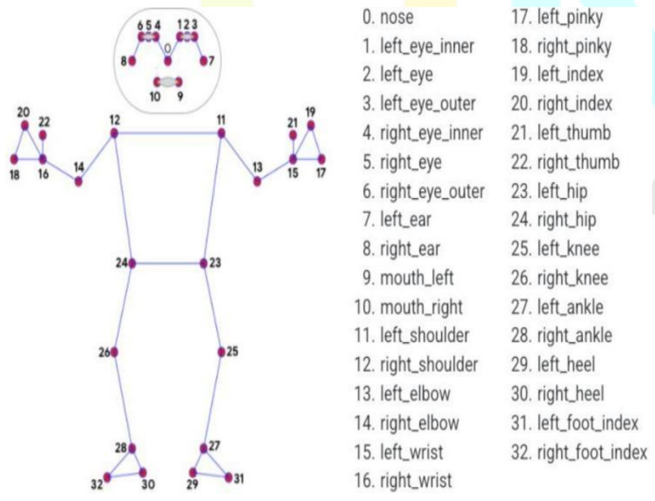
What we can do is we can check the frame rate by writing here time. Current time is equals to what we need to import time as well so write here import time and then write time dot time then write that our fps is equals to 1 divided by our current time minus our previous time. Our previous time is equals to current time and need to define the previous time up here at the top so write here previous time is equals to zero. Then simply put our text so write put cv2 dot put text and write in our image and then in the text itself. So write string fps but convert it into an integer. Before we do thatwe have our origin, let's say 70 and 50. Let's say the cv2 dot font plane and let's three and then for the color we can put two five five zero and two five zero three so this should giveus our frame rate. Let's run it we have it so you can see it's a hundred something frames per second. So that's quite a lot ifyou want to reduce it we can put here say 10. So now it'slike 60 frames per second but when using our model it will automatically slow it down so we don't need to worry about that. Next step would be to create our model so our object sothat we can detect purpose so here we are going to write mp pose is equals to mp.solution.



Solutions dot pose so going to use and then going to create our object we are going to say pose is equals to mp pose. Then give in our parameters, so if we go to the parameters you can see that we have the static image mode. This is basically that when you are detecting and when you are tracking. If you put it as true then it will always detect basedon the model it will always try to find the new detections butwhen you put it as false. It will try to detect and when the confidence is high it will keep tracking so there will be a tracking confidence and then there will be a detection confidence so whenever it detects if the confidence is more than 0.5 it will say okay now we have detected now i will goto tracking now the tracking will check if the tracking confidence is more than 0.5 it will keep tracking whenever itgoes below 0.5. Then it will come back to detection so this way we do not use the heavy model again and again. For detection instead we use detection then tracking them. Whenever it's lost we use the detection again so this is what it does. Then we have the upper body only so you candecide if you want to detect only the upper part. So it will have you can see here we have 33 poses landmarks or only 25 so it's up to you which one do you want to use. Here we also have a feature to smooth which is by default true. We will keep it as true so we can define all of these parameters or we can skip them it's up to us. For the initial purposes we are going to skip all of these for the simplicity. We are going to convert our image so we will say image rgb is equals to cv2 dot cvt color so this image is in bgr but this library or this framework uses rgb so to make it compatible. We have to convert it so write image and then write cv2 dot what color underscore bgr to rgb. This is our conversion andonce we have done the conversion we are simply going to send this image to our model.

Write here results equals to pose dot process. Write image rgb so that's pretty much so that will give us our detection our pose but it will not draw anything. For now we can just run it to see if everything is working and now you can see that the frame rate has decreased so this is good to see that our model is actually working. You can see that it's almost uh real time so that is really good. Now what we can do is we can draw our landmarks or whatever we have detected we can draw it. But before we do that we can print the results as well. Write here results you can see we get nothing so it's just a class but we don't see any information. How we see the information, we simply write results dot pose landmarks land marks. If we run that now you will see that we are getting actual landmarks so each landmark will have the x y and z value. Then it will also have a visibility value so how visible is it so this is the information that we get so we can put all of this in a list later on. So that it's easier to access so then we will check if it is detected or not.We will say that if this is present if this is true then we are going to write here that mp draw dot draw landmarks draw underscore landmarks. We are going to define our parameters but you can see we don't have anything. So we are going to declare that we will write here mp draw is equals to mp solutions solutions draw utilities. Use that and then we can write here within our landmarks we can send in our image then we can write results dot pose landmark so this is the same thing that we are printing out. We will write here land marks and now you can see we are getting all the points. You can see that it is in red now what we can do is we can also add the connections or the lines between these. So here we can write mp dot not mp pos dot pose connections. That will fill up the connections, we have the green lines which are the connections and n we have the points as well detecting the all at real time. Now we are getting this information but how can we know which is for which so here it just by says landmark.

What we need to do is to organize a little bit so that it is in a list and we can simply use these values in our project. For example if I want landmark number five and landmark number three so if we go to the media pipe website you can see here that these are the landmarks. That they have given so for example if I want the right ear so I can just say I want the element number five element number and eight element number so give me that element. If I want the nose I can say give me the element number zero so based on this it willgive me location of nose. This will become very easy for us to actually uh create our new projects so gesture recognition. A lots of different applications will become very easy so how can we extract the information within this object. Now we can write for id and landmark in enumerate we will writethis result. That’s going to loop through this and we wantthe count as well that's why we have written enumerate so it will give us the loop count over here so 0 1 2 3 and so on. Then going to get the shape. Write height width and channelis equals to img dot shape and the reason we need this shape is because it let us actually show you the output. Write print and write dot x or let's just print limit so you can see what is happening. Print the id number as well so you know what exactly we are extracting here. Run this so there you can seewe have the id number and this is the information of the landmarks. You can see here from 0 to 32 we will have all these 33 landmarks, now we know that we have the id and we have the landmark but you can see the landmarks theyare actually in decimal places so this is basically a ratio of the image.

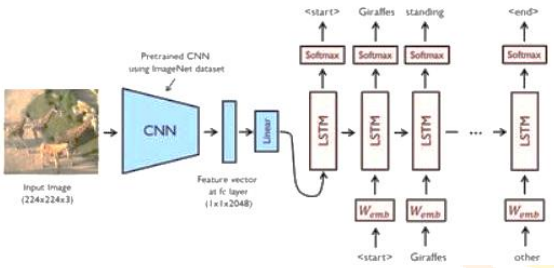


What we can do is to get the actual pixel value. Say that l m dot x this will give us this x value multiplied by the width ofthe image. That will give us the x of so you call the point or the landmark and it will give us the exact pixel value. The same thing we can do for lm dots y and multiply it with the height ad put them in our cx and cy variables. This way it will be easy for us to use the indentation. We need to convert this into integer because we have to make sure it's not a float or a double. Because talking about pixel values sothen what we can do is to confirm that this is happening and we are getting the correct values. Simply print the circle on

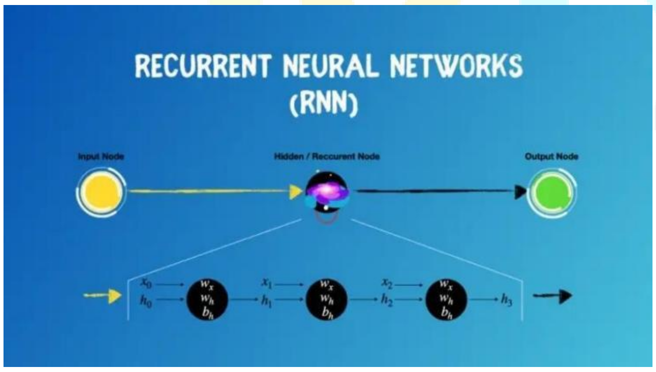
top of this point so we can write here cv2 dot circle and in the circle we will write image. Write cx and cy and write thevalue of the color and then we will write cv2 dot filled so this will overlay on the previous points. If we are detectingit properly we have those blue dots and these are the ones that we put ourselves. So this means that we are getting the correct information at the correct pixel values. It is working well we can reduce this to five and here its much well.

Try another video, we have been using the same video we have like nine videos or test case so let's try a random one. This is giving us very good results on many test cases.What we can do is we can convert this into a module so thatwe can use these values very easily. First thing we will do,we will create our module let's call it pose module and wewill copy all of the bare minimum code. Paste it here so thatlooks good now the first thing to make it a module.

If underscore name is equals to underscore main, then we are going to write main. So what this does is that if we are running this by itself then it will run the main function. If we are just calling another function it will not run the whole part. We will write main and within the main we are going to write everything. The dummy code in the main so whenever you want to see what a module is capable of the testing script. Create a class so in that class what we should be able to do is we should be able to create objects. We should be able to have methods that will allow us to detect the pose and find all these points. Pose detector and insideof our first method which is the. Initialize the parametersthat are required so whatever parameters we are expecting we will write mode. If I go to pause estimation and we are going to parameters and write so the first one is mode so we will keep it as false. So that we get fast detections plus tracking and then we will write here upper body so upper body is equals to false. Then we have the smoothness is equals to true and then we have the detection confidence is equals to 0.5 and then we have the tracking Confidence is equals to 0.5. So these will be our initial parameters andthen we can write self dot mode is equals to mode, now if you are not familiar with object oriented programming then this basically means that whenever we create a new object itwill have its own variables. So this whenever you write self dot, it is the variable of that object. So whenever we are using a variable within our class we will write self dotobjects self-mode self-upper body. The specific variable of our class and our object is basically the one that the user hasprovided so it will set this instance of that object to false. p body is equals to up body and then we will write smoothness is equals to smooth and then we will write detection confidence. Write track confidence declare these so again this will be part of that object so we need to write self so here we will write self.draw self.impose and self.poseand we will write here self.impose so this is good and what we can do is earlier we were not using any of these parameters but now we have to so here we are going to sendin all the parameters so we will write here self dot mode so self dot body self dot smooth self-thought detectionconfidence and self. Initialization is done and now what we can do is we can create a method to find the pose. So we canwrite find pose and self so whenever we are creating a new method we have to write cell first. We have to give in our image and then we will also have a flag called draw. We will put it as true so this basically what it will do is it will ask the user do you want to draw or not. Do you want to display it on the image or not, we want display so we are going to put this inside and this will draw results. Write the self dot so here we will write self dot pose process and self dot mp draw. Then we need to do is put that flag, we willsay that if landmarks are present or let's put the draw inside. So if the landmarks are detected and we set draw then tonew draw so that we should have a working class. We can create an object from it and then we can run. Try detector is equals to our pose detector and we will give in the default parameters. What is happening here is that this should be inside the while loop, so here we are going to write detector dots find pose. Then we have to give in our image input so that should be return the image detection. We will write return image and this should draw on our video. So let's run the pose module and now it running. Let's try it on the first image. This will track all 25 points in the image. Next is we can do the main part, which is to find the points. We are going to define the get position and we will write inside that we want an image and then again we want the draw flag so by default.



Let's put it as true and we are getting an error for results because we need to write self dot results and we need to push this in for loop. Now we need to check first as well if the results are available so if the results are available thenwe will use this for loop and then we need to put it in a list. We are going to write a list lm list which is for our landmarks and we will append our values. So here it depends on you what kind of values do you want to append. So we are going to append only the x and y values and the idif you want to append the z and the other one the visibility. Write lm list dot append and in the id. Then write the cx andthe cy and add the option that if draw then do this. lm list is equals to detector dot find position is it find position or get position. Keep it final position so it's similar to the previous one find position find position and then we will give in our image. Keep it as true for drawing so we know it is detecting and we can print out the list as well. We can print out lm list and now we have the list and if we go till the end you will see we have a total of 32 points so we have 33 points because we are starting from zero so now.



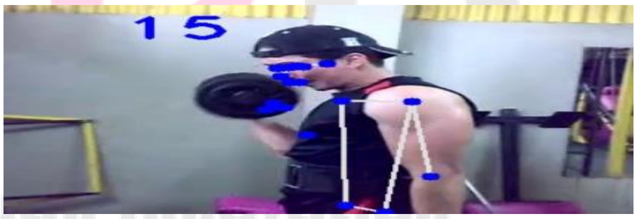
We can easily say for example we could draw number 14 so you have to put this at draw false and write lm 14 at one and we can see what exactly we are tracking. We can change the color of it so that it is a little bit different. For that we can put it as red and there you go can see we are tracking this elbow. Whenever it finishes the video it will give you an error you can loop through that there's a function for that as well but we're not going to do that so this is the basic idea ofhow you can convert this into a module. You can use this in any project that you want and you can easily find the position so how exactly can you do that. Create a new file and we will call this our awesome pose project. Copy the parts from the main so as i said this is the testing code so wewill copy all of that and we will paste it in our awesome project and now we have to import so we will write here import cv2 and then we have to import time and then we have to import our module so our module needs to be in the same folder if it's not in the same folder it will not work so you have to write import pose module as pm because it's a smaller name so we can write here pm dot post detector so this will create the detector and everything else will run the same way so if we run this now our projects.

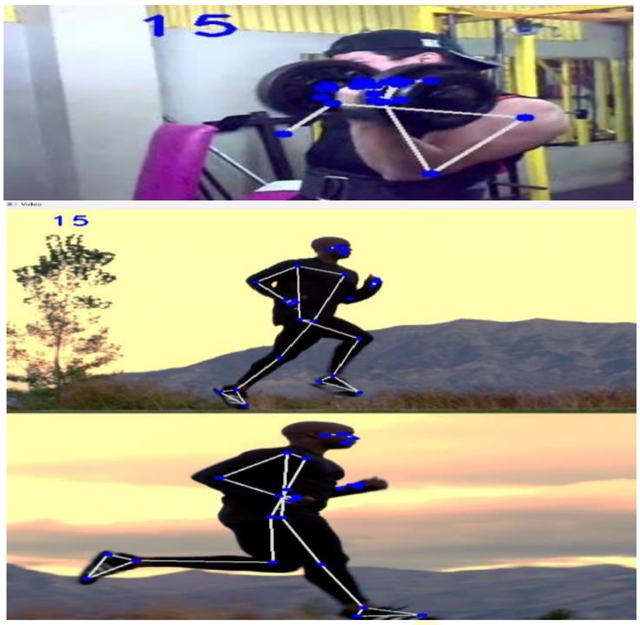
It runs exactly the same way so now you can use this in many different applications. One thing we can do is we can test some other videos. so let's say you go again, we are detecting the right elbow and it's tracking really well. Even when it's a little bit hidden it's still tracking it, that's really efficient. Let's try number three what happened to number three, there is an error list index out of frame. This is a problem because we are not checking uh if the list if actually the list is filled. We need to check if lm list is not equals to zero, actually the length of it is not equal to zero. Cause if the length of this is not equals to zero then what wecan do is change that at the main module. Let's run the project again and this time it works. The first frame was not able to detect that's why it was giving this error. So now youcan see it's detecting that elbow. You can detect the right elbow very nicely and with excellent frame rate. Best part is it's running on cpu it's not actually using gpu. There are a few instances where it gets it wrong but overall you can see it's really good so this is basically how you can detect posein an image or in a video. You can do the same thing on a webcam. Testing it again and again is just little hard. You can now use this pose module and all these files will be available, minimum code to detect the posture.

IV. RESULT

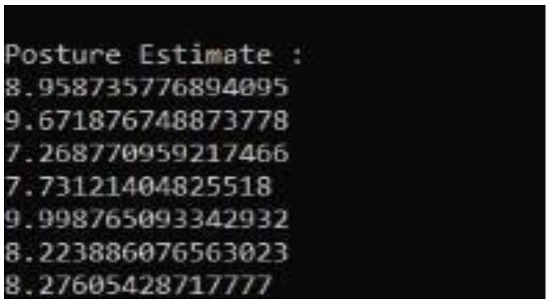
Using RNN the data points at the joins are identified in the frame and its locations pixel data was tracked to next frame.

Successfully tracking 14 points in the model





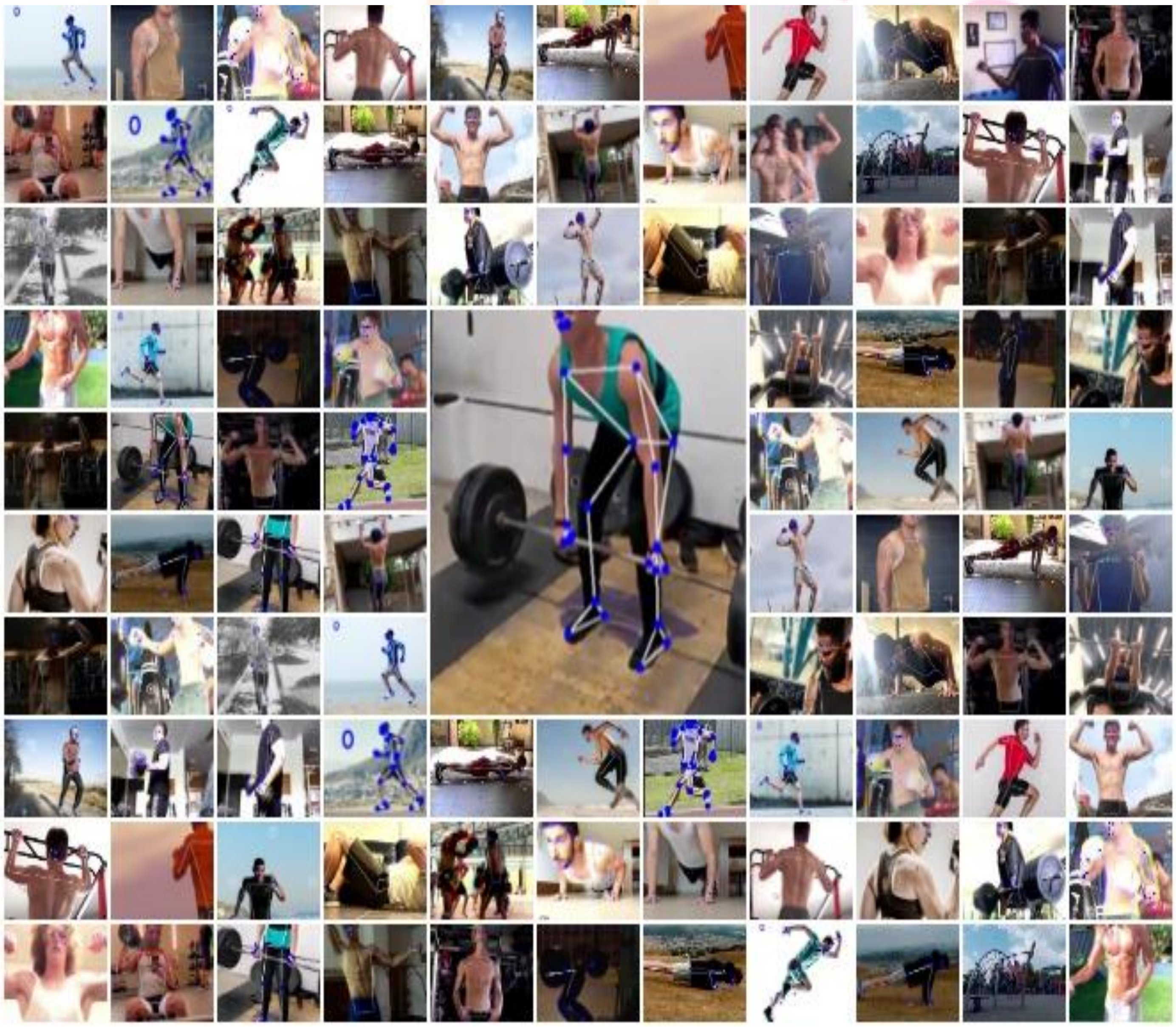
Posture estimate prediction in correlation with the perfect running posture when compared with input.



```
Posture Estimate :
8.958735776894095
9.671876748873778
7.268770959217466
7.73121404825518
9.998765093342932
8.223886076563023
8.27605428717777
```

Confusion Matrix

n=165	Predicted: NO	Predicted: YES
Actual: NO	50	10
Actual: YES	5	100



V. CONCLUSION

We have successful estimated the perpetual motion in a running athlete with the success rate from 74% to 98%* depending upon the source. This will help the user in creating a healthy body and mind. Increasing the overall mental and physical health and providing greater standard ofliving.

VI. REFERENCES

[1] Singh, V., Patade, A., Pawar, G., & Hadsul, D. (2022, April). trAIner- An AI Fitness Coach Solution. In 2022 IEEE 7th International conference for Convergence in Technology (I2CT) (pp. 1-4). IEEE.

[2] Mokmin, N. A. M. (2020). The effectiveness of a personalized virtual fitness trainer in teaching physical education by applying the artificialintelligent algorithm. Sciences, 8(5), 258-264.

[3] Aljuaid, H., & Mohamad, D. (2013, December). Object tracking simulates babysitter vision robot using GMM. In 2013 international conference on soft computing and pattern recognition (SoCPaR) (pp. 60-65). IEEE.

[4] Ioffe, S., & Forsyth, D. (2001, July). Human tracking with mixtures of trees. In Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001 (Vol. 1, pp. 690-695). IEEE.

[5] Ukita, N., & Nakazawa, A. (2013). Human Body-parts Tracking for Fine-grained Behavior Classification. In Proceedings of the IEEE International Conference on Computer Vision Workshops (pp. 777-778).

[6] Ha, T. Y., & Lee, H. (2019). Presenting Direction for the Implementation of Personal Movement Trainer through Artificial Intelligence based Behavior Recognition. Journal of the Korea Convergence Society, 10(6), 235-242.

[7] Correia, F., Gonçalves, G., Monteiro, P., Coelho, H., Melo, M., & Bessa, M. (2019, November). Evaluation of different body tracking configurations in the sense of presence and embodiment. In 2019 International Conference on Graphics and Interaction (ICGI) (pp. 101-106). IEEE.

[8] Aljuaid, H., & Mohamad, D. (2013, December). Child's Body Part Tracking Simulates Babysitter Vision Robot. In 2013 International Conference on IT Convergence and Security (ICITCS) (pp. 1-4). IEEE.

[9] Chen, H. K., Chen, F. H., & Lin, S. F. (2021). An AI-based exercise prescription recommendation system. Applied Sciences, 11(6), 2661.

[10] Stacey, D., Hopkins, M., Adamo, K. B., Shorr, R., & Prud'homme, D.(2010). Knowledge translation to fitness trainers: A systematic review. Implementation Science, 5(1), 1-9.

[11] Gregory, R. W., Henfridsson, O., Kaganer, E., & Kyriakou, H. (2021). The role of artificial intelligence and data network effects for creating user value. *Academy of management review*, *46*(3), 534-551.

[12] Ammar Anuar, Khairul Muzzammil Saipullah, Nurul Atiqa h Ismail, Yewguan Soo “OpenCV Based RealTime Video Processing Using Android Smartphone” , IJCTEE, Volume 1, Issue 3