IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

SPORTS TRAINING ACTION DETECTION

Mohammad Samad Alam Ansari[1], Mohd Ahsan Ansari[2], Dr. Pankaj Kumar[3]

[1-2]Student, Department of Computer Science and Engineering, Shri Ramswaroop Memorial College of Engineering and Management, Lucknow, India.

[3]Professor, Department of Computer Science and Engineering, Shri Ramswaroop Memorial College of Engineering and Management, Lucknow, India.

Abstract: This application allows to train a person for various exercises and give live feedback on the screen with a beep sound. And how it does that? the answer is that it detects the coordinates of the body joints using a wonderful machine learning algorithm called MediaPipe. This algorithm is created and maintained by Google; it detects total 33 landmark on human body from head to toe. This application then takes the coordinates and evaluates the vector geometry of the pose through the exercises, giving the user helpful feedback in real-time. This allows the user to identify any potential problems with their pose and to quickly make adjustments to ensure the accuracy of their form. The feedback provided by this application helps the user to improve their technique and to avoid any potential injuries that could be caused by incorrect form. By evaluating the vector geometry of the pose, the application is able to provide valuable insights that can be used to help the user achieve their fitness goals. Pose estimation is a type of method in which spatial locations or parts of key body joints is calculated using the live camera of the device with a good camera quality so that all the body joints are visible clearly. This computer vision technique is very uniquely detecting human body posture in live picture form and shows the keypoints of the body like elbow, hip, knee as output to the device in live time which is a lot better than uploading image or video.

The output created by the algorithm is create by comparing the difference between the user joints angle input and the fixed angle saved in the database.

Keywords **- CNN, MediaPipe, OpenCV, Numpy, Pandas.**

I. INTRODUCTION

Numerous exercises, such dead lifts, squats, and shoulder presses, are beneficial for improving human body fitness, but they can also be quite damaging. If they are carried out incorrectly. The hefty weights employed in these exercises can lead to injuries to the ligaments or muscles. Many people do not maintain the proper posture while practicing these exercises frequently because they lack the training or understanding necessary. Muscle tension and tiredness could result from this. By creating a project that recognizes the user's position while exercising, provides feedback, and makes modifications as needed, we assist people in executing workouts with proper posture by utilizing the most recent pose estimation methodologies.

The first step of the project is to use pose training system which uses computer vision technique for camera configuration. In which a trained model called MediaPipe that determines a person's joints as a list of skeletal key points from the given data which it gets from the images which is in an RGB image. Pose trainer plays an important role in solving problems related to human body poses detection and also for the activity of recognition movement of body detection. MediaPipe is used which utilizes neural network for the detection purposes. The latter part of the project involving calculating of the angles of the body for any exercises.

Controlling PC eyesight is tough because determining human position is complex. To induce bone exposure, controls the confinement of human organs in a photo or film. As a result, detecting a person's posture in an image is challenging since it depends on a variety of factors, such as the image's scale and purpose, the varying levels of illumination, the types of clothes worn, natural characteristics, and human interactions with natural aspects. The application should facilitate the training of pose estimation models using the annotated

datasets. It should provide options for hyperparameter tuning, model selection, and performance evaluation metrics to ensure optimal model performance.

The trained pose estimation models should be integrated into the application to enable real-time pose estimation from live video streams or recorded videos. The inference process should be efficient, accurate, and capable of handling variations in poses, backgrounds, and lighting conditions.

The application should have a user-friendly interface that allows researchers, developers, and practitioners to easily interact with the various functionalities. It should provide intuitive visualization tools for dataset exploration, model training progress monitoring, and pose estimation results analysis.

II. LITERATURE REVIEW

According to Deepak Kumar et al, human posture assessment has been focused heavily in recent years. In health and sports, posture assessment can assist prevent wounds and improving the execution of individual exercises. Recommends, that exercise self-guidance frameworks have the ability to make exercise well-known while also guaranteeing that it is practiced correctly. As a result of the extensive research being conducted on this subject, profound learning approaches appear to be a promising outcome. The use of a combination CNN and LSTM model using MediaPipe data appears to be successful since it correctly arranges all six exercise poses. A basic SVM and CNN also perform substantially beyond our expectations. In comparison to a neural network, SVM is lighter and less unpredictable, and it takes less preparation time [1].

According to M. Arulselvi et al, in this work, various karate and bharathanatyam moves which are original and naive images were captured and action classification was done. The classification is done with the Keras library, which runs on top of Tensorflow using a deep learning algorithm. The proposed work correctly classifies the stances 62% of the time. The goal of this work was to create a multi-view dataset for various stances that included actions from martial arts, sports, and dancing. The work can be used to classify a wide range of positions. The classification of certain positions was poor, and better results will be obtained by fine-tuning parameters or using other machine learning approaches [2].

According to Yi Liu et al, this paper focuses on the state-of-the-art progress of 2-D human pose estimation based on deep learning. Classification, introduction, and performance comparison of the methods are involved. Single CNN methods lead to the low complexity of the network. Multi-stage CNN algorithms improve feature extraction and use the receptive field to represent the interactions between neighboring joints. However, issues such as gradient descent, which are exacerbated by increased network depth, must be addressed. Multi-branch CNN approaches combine the outputs of numerous CNN branches, but at the cost of more complicated calculations and training. RNN approaches describe the interrelationships between body joints with the help of sequential processing utilized in Natural Language Processing, extend the receptive field with the iteration of the recurrent network, and help in overcoming the occlusion problem to a certain degree at the cost of huge calculations. In the absence of target class label information, GAN approaches to capture the high order correlation of the data. Such methods have provided the most up-to-date and effective answer to occlusion. The disadvantage is still the time- consuming calculation [3].

Marina Pismenskova et al proposed a neural network-based approach for human pose recognition. Convolutional neural network-based methods have made a substantial contribution to the problem of processing video data in intelligent surveillance systems. In this paper, a method for recognizing a person's pose based on previously produced high- level data of a human skeleton is proposed, which reduces the error of excessive information. The prepared high-level data is supplied to the network's input. You can limit the amount of information and learning time by using descriptors instead of the entire image as input [4].

According to Yu Kong et al, the availability of massive data and advanced models shifts the research focus on human actions from comprehending the present to thinking about the future. A comprehensive overview of current state-of-the-art techniques for action recognition and prediction from videos was presented. Due to their promise and practical applications in various growing domains focusing on human movements, these approaches have been particularly fascinating in recent decades. Handcrafted feature design, models and algorithms, deep architectures, datasets, and system performance evaluation procedures were all investigated as part of the existing efforts [S].

According to Zhe Cao et al, Real-time multi-person 2D pose estimation is a critical component in enabling machines to visually understand and interpret human interactions. A nonparametric representation of the key point association that encodes both the location and orientation of human limbs was presented. Developed an architecture that learns part detection and association at the same time. It was shown that a greedy parsing method is enough to provide high-quality body position parses while maintaining performance regardless of the number of persons.

It was demonstrated that merging body and foot estimates into a single model improves the accuracy of cache component separately while reducing the inference time when performed sequentially. We created a foot key point dataset with 15K foot key point instances, which we will make available to the public. This work was released as OpenPose, the first real-time system for detecting critical points on the body, foot, hand, and face. Many academic subjects requiring human analysis, such as human re-identification, retargeting, and Human-Computer Interaction, are now frequently employed in the library. Furthermore, OpenPose is now part of the OpenCV library [6).

Abhishek Gupta et al, proposed a Exercise identification system using a traditional RGB camera. The data was obtained from 15 people (ten men and five women) using an HD 1080p Logitech camera and made publicly available. The user was captured and important points were detected using OpenPose. The end-to-end deep learning-based system eliminates the requirement for handcrafted features, allowing the model to be retrained with new data to include new asana. The time-distributed CNN layer was used to find patterns between important points in a single frame, while the LSTM was used to remember the patterns discovered in recent frames. The results make the system even more resilient by lowering the error due to false key point detection by using LSTM for frame memory and polling for denoising [7].

According to Y. Agarwal et al, A system is suggested that classify ten exercise poses and the dataset upholds six classification models of machine learning. The angles retrieved from the Skeleton joints of the TF posture prediction algorithm are used to recognize the exercise pose. All machine learning models achieved an overall accuracy of 94.28 percent. Google Colab and an Ubuntu 18.04.4 LTS terminal were used for data preprocessing and model training. Future plans include expanding the YOGI Dataset to include more exercise positions and incorporating deep learning modules to improve performance. Additionally, an audio guiding system can be incorporated [8].

According to Shruti Kothari, Human pose estimation has been studied extensively over the past years. As compared to other computer vision problems, human pose estimation is different as it has to assemble and localize human body parts on the basis of a previous defined structure of the human body. The use of pose estimation in fitness and sports can help

people avoid injuries and increase their workout performance. According to the author, exercise self-instruction systems have the ability to make exercise more popular while also ensuring that it is practices correctly. Because of the extensive research being done in this sector, deep learning methods appear to be promising. The implementation of hybrid CNN and LSTM model using OpenPose data has proven to be quite effective. A simple CNN and SVM also perform better than our prediction. SVM's performance demonstrates that machine learning methods can be used for solving problems like pose estimation and activity detection. In addition, when compared to a neural network, SVM is lighter and less sophisticated, and it requires less training time [9].

According to Anubhav Singh et al, if the authors consider PC vision assignments convolutional neural networks system are the most demanding design due to their ease and instinctive nature, and their lessened number of a parameter when diverged from the totally related model. CNN has the advantage of accepting the entire picture as an information motion for each body point, as opposed to nearby locators, which are limited to a specific section. On a testing scholarly dataset with a rudimentary model, CNN is used to achieve aggressive results on the subject of human posture estimate. Changing the base learning rate and learning rate approach, experimenting with different types of momentum updates, and fine-tuning the regularization quality are all fun things to do. Furthermore, display gatherings can be used to improve performance. It's possible that 2D pixel coordinates won't provide enough useful information for a true 3D posture calculation [10].

Xuecheng Nic et al, presented the first single-stage model, Single-stage multi-person Pose Machine (SPM), for multi-person pose estimation. This SPM model offers a more compact pipeline and attractive efficiency advantage over existing two-stage-based solutions. The superiority of SPM mainly unifies the person instance and body joint position information and solves the intrinsic limitations of conventional pose representations. In addition, it presented a hierarchical extension of SPR to efficiently factorize long-range displacements into accumulative short-range ones between adjacent articulated joints, without introducing any extra complexity to SPR. With SPR, SPM can be used to estimate the poses of multiple persons in a single-stage feed-forward manner.

IV TECHNOLOGIES INVOLVED

Python

The primary development language for our project is Python. Our project's framework is provided by Python. Guido van Rossum created Python, an interpreted, object-oriented, high-level programming language with dynamic semantics. It was first made available in 1991. The name "Python" is a tribute to the British comedy group Monty Python and is meant to be both simple and entertaining. Because it manages much of the complexity for the user, Python has a reputation for being a beginner-friendly language, replacing Java as the

most popular introductory language and allowing beginners to concentrate on fully understanding programming concepts rather than minute details.

Python is popular for Rapid Application Development and as a scripting or glue language to tie existing components together because of its high-level, built-in data structures, dynamic typing, and dynamic binding. Python is also used for server-side web development, software development, mathematics, and system scripting. Python's easy-to-learn syntax and emphasis on readability result in lower programme maintenance costs.

Additionally, Python's support for modules and packages makes it easier to reuse code and create modular programmes. As an open-source community language, Python is constantly being developed by numerous independent programmers.

CNN

Convolutional Neural Network is referred to as CNN. It is a specific kind of artificial neural network that is especially effective at processing and image recognition tasks. CNNs are frequently employed in computer vision tasks like segmentation, object detection, and image and video recognition. CNNs use several convolutional layers to identify patterns in visual data. These layers use filters to extract features from the input data that are pertinent to the work at hand, such as edges and textures. In order to incorporate nonlinearity into the network, the output of each convolutional layer is then processed through a nonlinear activation function, such as the Rectified Linear Unit (ReLU). The output is often transmitted through one or more fully connected layers after the convolutional layers, which carry out a classification task using the features retrieved from the earlier layers. Typically, a SoftMax function is applied to the output of the final fully connected layer to produce a probability distribution over all feasible output classes.

CNNs have been employed in applications including self-driving cars, facial identification, and medical image analysis because they have been demonstrated to be extremely effective in a variety of image recognition tasks.

Deep Learning

A technique used in artificial intelligence (AI) called deep learning teaches computers to interpret data in a manner modeled after the human brain. Deep learning models can identify intricate patterns in images, text, audio, and other types of data to generate precise analyses and forecasts. Deep learning techniques can be used to automate processes that ordinarily require human intellect, such as text-to-sound transcription or the description of photographs.

Neural networks used in deep learning algorithms are modeled after the human brain. As an illustration, the human brain has millions of linked neurons that cooperate to process and learn new information. Similar to this, deep learning neural networks, also known as artificial neural networks, are constructed inside of computers from numerous layers of synthetic neurons.

Artificial neurons are nodes, which are software components that process data using mathematical operations.

These nodes are used by artificial neural networks, which are deep learning algorithms, to resolve complicated issues. The following are the elements of a deep neural network.

Input layer

A number of nodes provide data into an artificial neural network. The input layer of the system is made up of these nodes. This notes take the inputs from outer world and pass it to the hidden layer of the neural network.

Hidden layer

The input layer of the neural network processes and transmits the data to subsequent layers. These concealed layers employ many degrees of information processing, changing their behavior in response to fresh input. Deep learning networks may examine an issue from numerous perspectives because they have hundreds of hidden layers at their disposal.

Output layer

The nodes that output the data make up the output layer. There are just two nodes in the output layer of deep learning models that provide "yes" or "no" responses. Conversely, those that produce a larger variety of responses have more nodes. Deep learning is a subset of machine learning in the context of this article.

The accuracy of the results in supervised learning only increases with a large and suitably diverse dataset. For instance, because there were more photos of black cats in the training dataset, the algorithm might correctly identify black cats but not white cats. If so, you would need to identify more pictures of white cats and retrain the machine learning models.

Computer Vision

One branch of artificial intelligence called computer vision teaches and equips machines to comprehend the visual environment. Deep learning models and digital photos can be used by computers to precisely detect, categorize, and respond to objects. Work is made easier for humans by computer vision systems' faster completion of boring, repetitive activities. Work is made easier for humans by computer vision systems' faster completion of boring, repetitive activities. Very well-trained computer vision systems will have zero errors. Faster delivery of high-caliber goods and services will be the outcome.

MediaPipe

Mediapipe is a cross-platform/open-source tool for running real-time machine learning models. It was created by Google and is largely intended to make the use of machine learning in streaming video easier. It's an ML solution for high-fidelity body posture tracking that uscs our Blaze Pose research to infer 33 3D landmarks and a background segmentation mask on the full body from RGB video frames. Our solution achieves real- time performance on most modern mobile phones, desktops/laptops, in Python, and even on the web

V. METHODOLOGY USED

Realtime pose estimation, classification, and pose validation has been an arca of research for the past two to three decades. On the basis of the literature review done by us, we realized the basic steps to be followed for building the AI-Based are:

1. Data Collection.
2. Data Pre-processing.
3. Pose Estimation
4. Pose classification.
5. Building the User Interface.

DATA COLLECTION AND DATA PRE-PROCESSING

Data collection is a fundamental step in building an AI-based exercise trainer. It involves gathering a comprehensive dataset of exercise examples or demonstrations. This can be achieved through various means, such as video recordings, motion sensors, or specialized exercise tracking devices. The dataset should cover a wide range of exercises and variations to ensure the robust training of the AI model. Careful consideration should be given to capturing diverse populations, different body types, and various skill levels.

It requires a large number of exercises postures performed from different camera angles. We didn't find any such dataset online and it was not feasible to collect data from YouTube videos and other sites. To overcome the problem of data, we created a custom data collection website to collect user poses.

The data collection portal was made using Tkinter, pymysql, Mysql database was used to store the collected exercise poses.

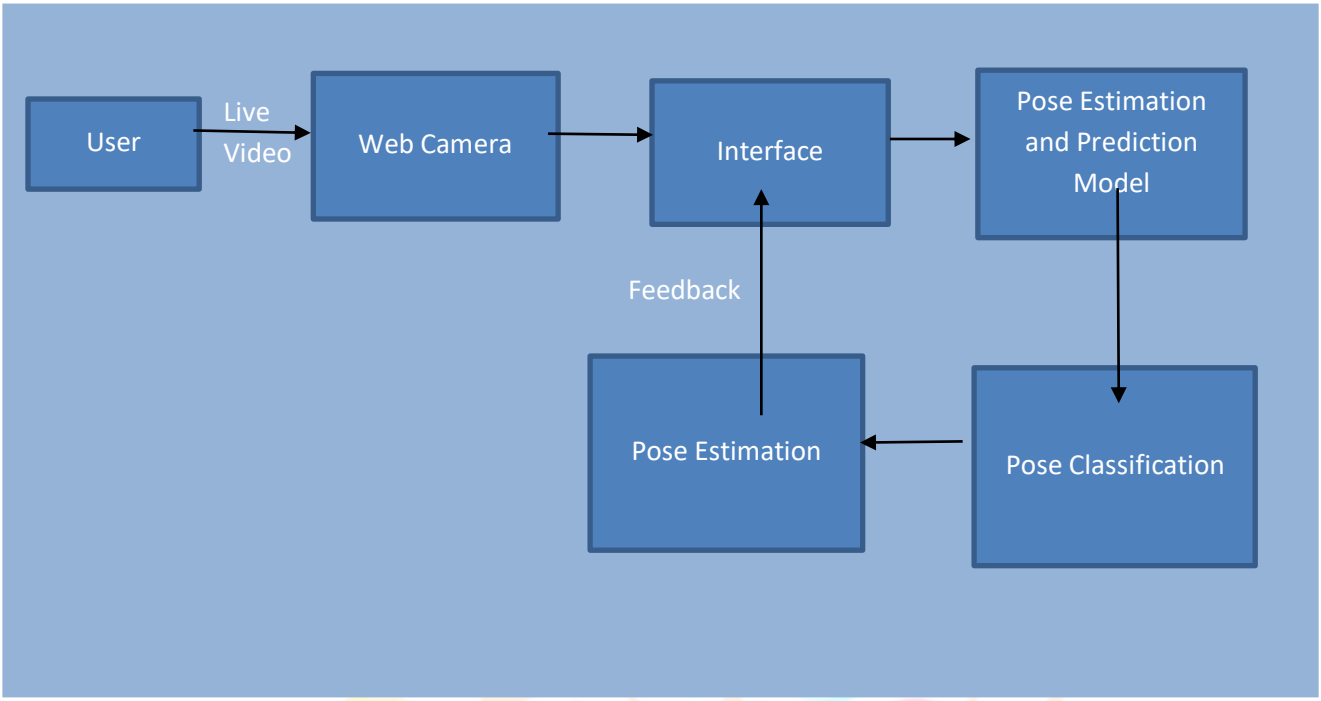


Fig : 3.8.1 Architectural Diagram

The data collection portal has a list of three predefined exercise poses that the users were asked to perform. The user then selects a specific exercise pose that he/she wants to record.

SPORTS TRAINING ACTION DETECTION

Sports Training Action Detection is a computer vision task that infers a person's or object's pose from a photograph or video. Sports Training Action Detection can alternatively be defined as the challenge of determining the position and orientation of a camera in relation to a person or an item. This is usually accomplished by locating, tracking, and identifying a number of key points on a specific object or person. This could be corners or other distinguishing elements on an object. These key points represent significant t joints in humans, such as the elbow and

Once the data is collected, it often needs to undergo pre-processing before being used to train the AI model. Data pre-processing involves several steps to ensure the data is in a suitable format for analysis. This may include resizing or cropping images or videos, normalizing the data, removing noise or outliers, and formatting the data to facilitate further analysis. Proper data pre-processing enhances the quality and reliability of the subsequent steps in the methodology.

POSE CLASSIFICATION

Pose estimation is a critical component of an AI-based exercise trainer. It involves detecting and tracking key body joints or landmarks from the input data, such as images or videos. Computer vision techniques, often employing deep learning models, are used to accurately identify the positions of body joints. This step enables the AI model to understand body movements and postures during exercise, forming the foundation for subsequent analysis.

The process of recognizing, interpreting, and classifying objects and thoughts into predetermined groups, often known as "sub-populations," is known as classification. Machine learning systems use a variety of algorithms to classify future datasets into appropriate and relevant categories with the help of these pre-categorized training datasets.

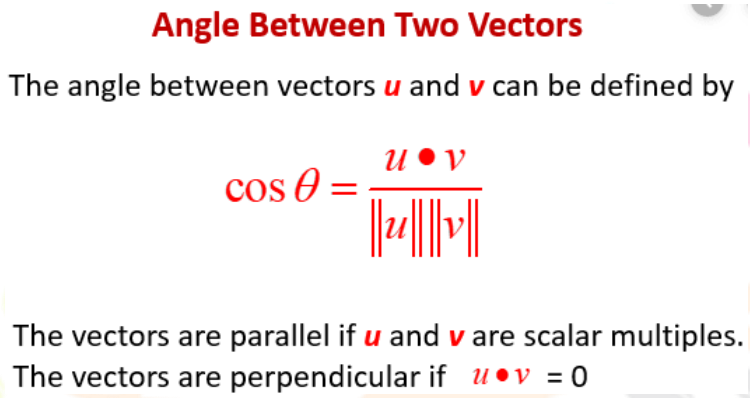
Input training data is used by classification algorithms in machine learning to predict the likelihood or probability that the data that follows will fall into one of the specified categories. To put it another way, categorization is a type of "pattern recognition." In this case, classification techniques used on the training data uncover the same pattern in subsequent data sets.

POSE ESTIMATION

The user is notified about the accuracy of his or her pose. The user receives feedback based on the angle difference so that he or she can improve their posture.

The similarity score given by the feedback generation module informs the user about the accuracy of his or her pose. It employs a similarity algorithm to determine the degree of resemblance between the user's stance and the ideal exercise pose. There are a number of distance similarity algorithms available, each with a varying level of accuracy and validation.

Cosine Similarity with Euclidean Distance: The distance between two vectors can be determined without being contaminated by the fluctuating length of the vectors by using he cosine similarity in the Euclidean distance formula. The cosine similarity equation computes the likeness in orientation between two vectors by utilizing the cosine to calculate the cosine of the angle between them, whereas the



Euclidean distance computes the length a between the two vectors from which the angle is calculated.

BUILDING USER INTERFACE

The user interface is a crucial aspect of an AI-based exercise trainer. It serves as the platform through which users interact with the trainer and receive real-time feedback and guidance during their exercises. The user interface can take the form of a mobile app, web application, or any other platform that accommodates user interaction. It should provide clear instructions, intuitive visualizations, and meaningful feedback based on the AI model's output. A well-designed user interface enhances user engagement and facilitates effective exercise training.

VI. RESULTS

We were successful in developing a application that allows the end user to engage with artificial exercise ai. The application allows the user to choose a position that he or she wishes to accomplish. The user is subsequently given a sore that symbolizes how good of the exercises performed by the user.

The application's basic flow is as follows:

- The user register for in the application by its credentials
- The user selects a exercise from the list of button on the screen.
- The user clicks the button
- The user adjusts their position.
- The user should face the camera and the camera should have a full view of the user’s leg
- The user then performs the selected stance on the screen
- On the screen the reps is increment shows
- Alert for the wrong posture shows with a beep sound

Fig 4.2 Reps Counter

Fig 4.3 Alert Box

Fig 4.4 Login Window

Fig 4.5 Signup Window

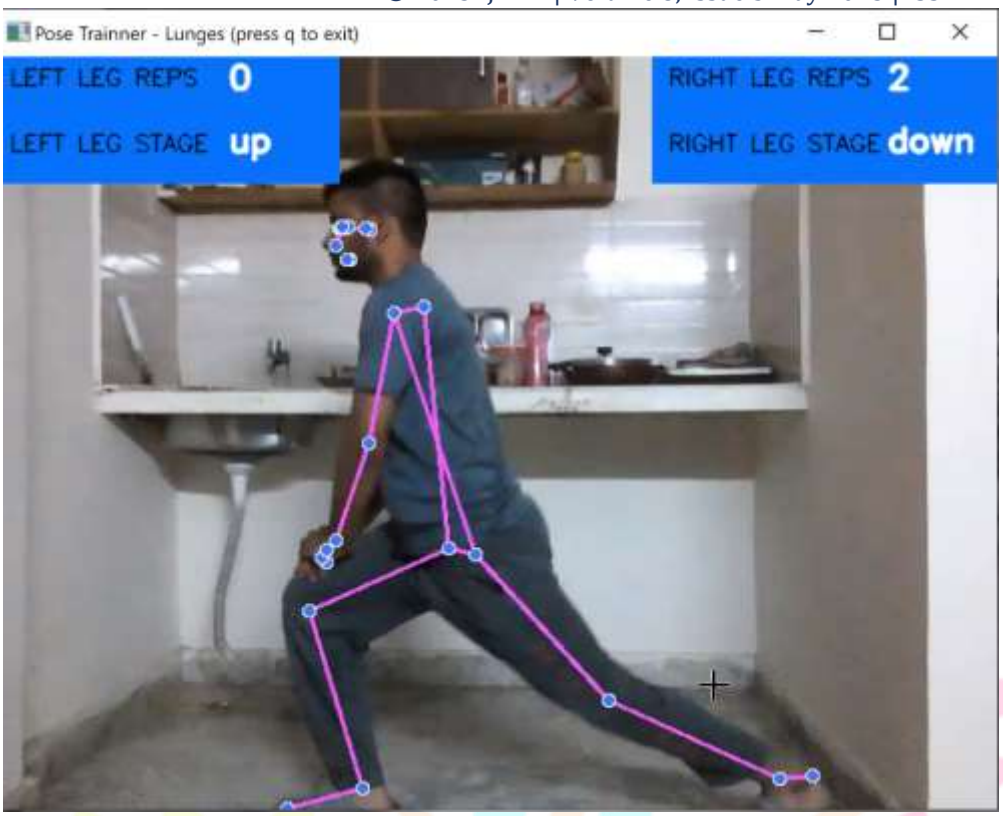


Fig 4.6 Lunges Exercise

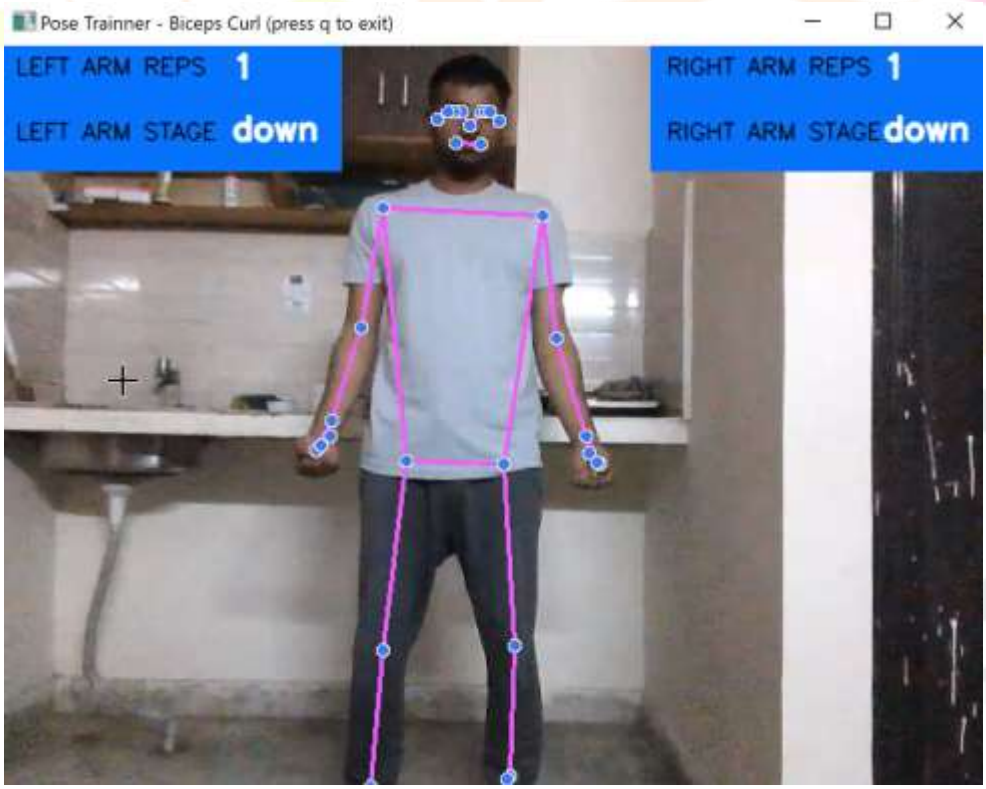


Fig 4.7 Biceps Curl Exercise

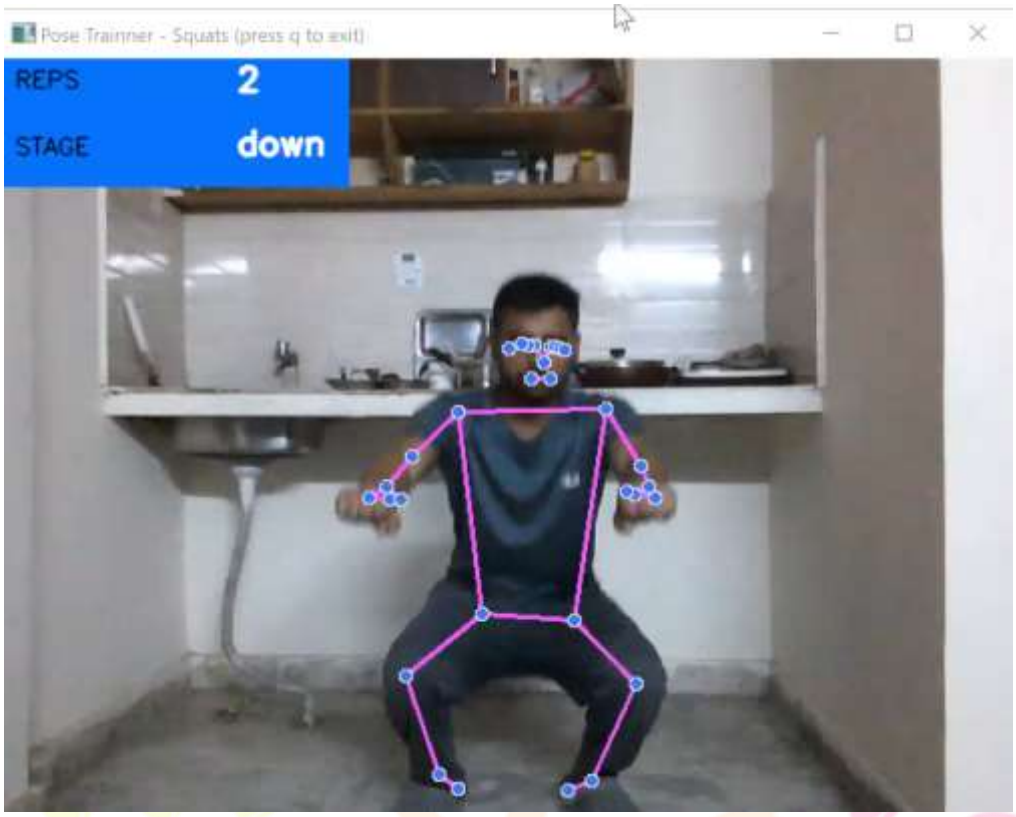


Fig 4.8 Squats Exercise

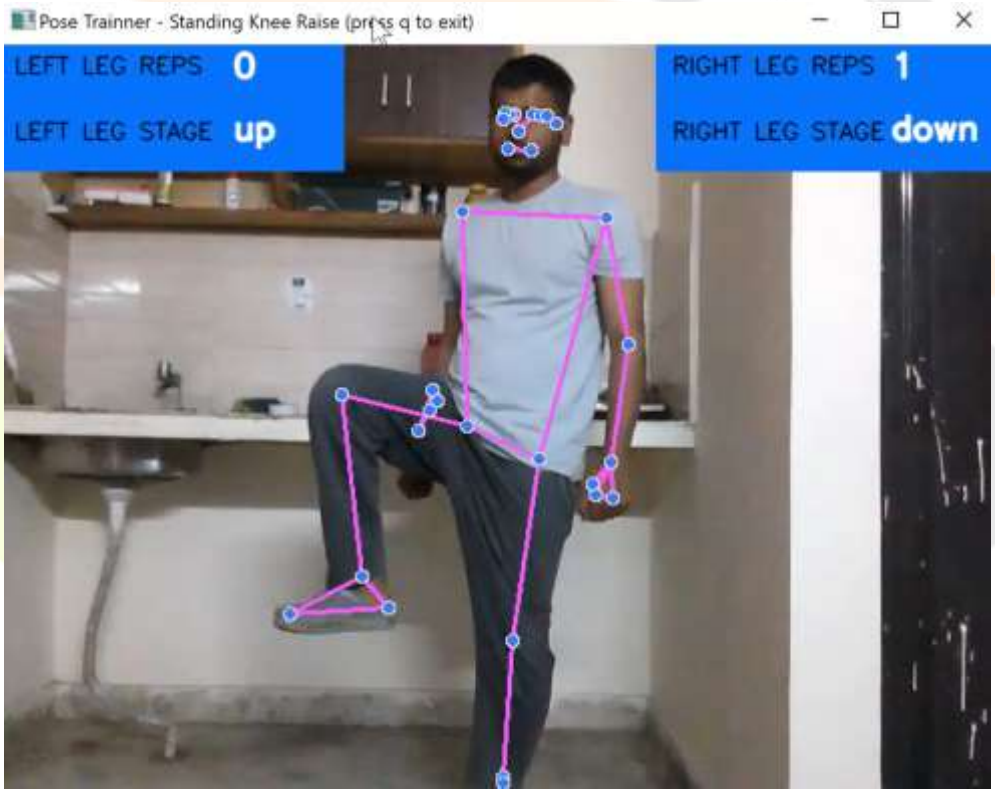


Fig 4.9 Standing Knee Raise Exercise

4.7 Bicep Curl

In Bicep Curl, From the key points of shoulder, elbow, forearms the angle between upper arm and forearm vector is calculated which define the movement of the hands upper arm. If the upper arm does not move up and the angles from elbow is more than 90◦, then it should be almost parallel to forearm for the entire frames of the image. When there is large change in the angle between the vectors it indicates that there is a movement which is happening in the upper arm and lower arm. The forearm should be brought up near to the shoulder so that the bicep is fully contracted there is less than 90◦ between upper arm and forearm. There is minimum threshold is set which define 1 rep for the angle between upper arm and forearm. At the start of the exercise the angle between the upper arm and forearm is near to 180◦ which decreases as bicep is contracted upwards and came closer to shoulder and again increases when the is brought downwards. From the data, the feedback will be generated based on posture of the user regarding the movement of the arm. When the position of both the leg of the user and the position the hand is not correct it gives the alert to the user with message on screen and beep sound. Until the position of user not correct it will not start counting the reps. (Refer Fig.3)

4.9 Standing Knee Raise

In standing knee raise, the hip, knee, and ankle key points are used, the user knee lift slightly above the hip about 90 degree and avoid using the movement of torso for lifting the legs. To calculate the form the posture is used during this exercise, two things are necessary to be identified: the Maximum angle between the hip, knee and ankle and the motion of thigh up and down. To detect the swinging in the lower body between each of the frames of exercise, the changes in difference of the vector is calculated from the motion of up and down the legs. (Refer Fig.4)

4.8 Squat

In squat, as similar to the knee raise exercise the motion of legs should be monitored and calculated. The angle of the joints connected to the knee must be greater the 90 degrees as threshold. Also using torse for lifting the leg indicates incorrect form. The angle is calculated by the angle between hip, knee and ankle, torso movement indicates the range of motion of the body. Lower body part movement indicates incomplete motion of legs and the small angle denote bending of knee which means one rep is correctly done. (Refer Fig.5)

4.6 Lunges

In Lunges exercise, the main key body points are the hip, knee, ankle. It is almost similar to the above exercises but the angle between them are very different. As for each rep of each leg the angle is measured and it must be near to 90 degrees for one leg and the torse of the body must be vertically straight which indicate correct posture of the body for the exercise unless no rep is counted. . Lower body part movement indicates incomplete motion of legs and the small angle denote bending of knee which means one rep is correctly. To detect the swinging in the lower body between each of the frames of exercise, the changes in difference of the vector is calculated from the motion of the leg (Refer Fig.6)

VII. CONCLUSION

Sports Training Action Detection involves the detection, estimation, and classification of various Exercise posture using DL ML, and computer vision, we have also built a user-friendly UI that would help us, to test the data easily without opening the code editor. The user-friendly wants to perform and practicing with Al-Based assistance. We compared different pose estimation techniques to find the best which is light-weight reliable and was able to cover a wide range of poses. We started with posenet. Posenet is a real-time pose detection technique with which you can detect human beings' poses in Image or Video. it has its limitations when it comes to providing high-quality results. In order to obtain high-quality results, using a more advanced pose-detection system is recommended. Such systems can provide more accurate and reliable results, leading to better performance for applications that require pose detection. and reliable but was not able to provide high fps and smooth pose landmarks.

We then explored Open Pose which is the first real-time multi-person system to jointly detect human body, hand, facial, and foot keypoints (in total 135 keypoints) on single images. The major problem with open pose was that it was not light weight to run over web browsers which only supports light weight module.

Media Pipe python allowed to implement pose estimation in python which allows it to run directly in the desktop. MediaPipe offered fast, reliable and lightweight pose estimation solution.

After Pose estimation the major challenge was to find algorithms to classify poses into one of various exercise poses. Instead of using various machine learning models like Naive Baye's, SVM, KNN and other models we started to use the heuristic angle-based approach to classify the pose into various exercise poses. Any user learning and practicing various exercise postures can use the user interface developed.

Nowadays, an AI-Based Exercise Trainer is needed in order to provide an easy and interactive way to perform and practice exercise indoors. Sports Training Action Detection is an Al-based exercise trainer who fills the gap in traditional methods of performing exercise. Using the provided system. the user can choose the posture they wish to perform and then perform the selected exercise pose in front of a webcam. The user's live position/posture is compared to the professional position, and, the difference in angles of the various body joints is calculated. Based on these angle differences, feedback is given to the user so that he can improve his posture.

The technology that we used in this project does not require any additional expensive hardware or tools. The following tasks will be accomplished with the help of AI recognition software and it is very less expensive:

- The model is able to detect various kinds of exercise poses using a media pipe.
- The model is able to classify the exercise postures based on a live video feed and is able to provide feedback accordingly.

REFERENCES

[1] Z. Cao, S. Wei, T. Simon, and Y. Sheikh. They created a Real-time multiple-person in 2d form pose detection. In CVPR, 2017.

[2] A. Newell, K. Yang, and J. Deng. Used hourglass networks technique for human pose detection. In ECCV, 2016

[3] G. Papandreou, J. Tompson, T. Zhu, A. Toshev, C. Bregler, N. Kanazawa, and K. Murphy. For accurate multiple people pose estimation in the wild. In CVPR, 2017.

[4] V. Ramakrishna, S. Wei, T. Kanade, and Y. Sheikh," CNN based Pose Detection algorithm in a Device," in CVPR, 2016.

[5] W. Rawat and Z. Wang. Deep Convolutional Neural Networks for Image Classification: In 2017

[6] T. L. Munea, H. T. Weldegebriel, Y. Z. Jembre, and many others," For Pose Detection in a 2D Model," in CVPR, 2020.

[7] X. Chu, W. Ouyang, and others," For Pose Detection," In CVPR, 2017.

[8] F. Bogo, A. Kanazawa, P. Gehler, C. Lassner, J. Romero, and M. Black. For Automatically calculating the 3d human pose in a single image. In ECCV, 2016.

[9] Steven Chen & R. Yang. Correcting Poses using Pose Detection. In CVPR 2018.