JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

AI FITNESS TRAINER

[1]Kashish Jain, [2]Jignasha Jadav, [3]Manashvini Yadav and [4]Dr Yogita Mane

[1]Student ,[2]Student, [3]Student, [4]HOD
[1]Information Technology,
[1]Universal College of Engineering, Thane-Vasai, India

Abstract : Inactivity is one of the main causes of obesity which has affected many people worldwide. Studies show that fitness is an important goal for a healthy lifestyle and is been used as a measurement for health-related quality of life. A fitness trainer can motivate and teach users to do exercise daily and stay fit and healthy. However, to use a fitness trainer might involve a huge cost and sometimes is not suitable for a certain setting. Exercises are very beneficial for personal health but they can also be ineffective and truly dangerous if performed in an incorrect method by the user. There are lot of mistakes made during a workout when user workout alone without supervision like wrong form which could result fatal for user as they can pull a hamstring or even fall due to it. In our project, we introduce AI Trainer, an application that detects the user's exercise pose and provides personalized, detailed recommendations on how the user can improve their form. Pose Trainer uses the state of the art in pose estimation module known as "BlazePose" tool from "MediaPipe" to detect a user's pose, then evaluates the pose of an exercise to provide useful feedback. We record a dataset of over 1000 keypoints coordinate of parts of body in correct and incorrect form, based on personal training guidelines, we build a machine learning algorithm for evaluation. AI Trainer works on six common exercises and supports any Windows or Linux computer with a GPU and a webcam.

IndexTerms - AI Fitness Trainer, Machine Learning, Pose detection, BlazePose, Health, Workout.

I. INTRODUCTION

The current scenario of pandemics and lockdown made people staying in home for long periods of time, but this cannot be a relevant excuse for being unproductive because it is an excellent idea to utilize the extra time we get for our own health. The main motivation behind this project is to make exercise easier and fun for people and make it more effective for them so that they can exercise more effectively in their own homes.

These days virtual assistant plays a crucial role in our day-to-day life activities and has become an inseparable part of our lives. AI is one such emerging field that we aim to explore through this project of AI-based workout trainer. In our project, we introduce AI Fitness Trainer, a desktop app that detects the users exercise pose, counts the specified exercise repetitions and provides recommendations on how the user can improve their form.

We use BlazePose tool from MediaPipe for pose detection module when user do their work out, and afterwards analyses the form of the pose from the dataset and real-time video and counts the repetitions of the particular exercise.

We started this project during pandemic and when all the gyms were shut down and we were in lockdown. At that time, we understood how important is fitness and how such situations can let us to work out in our homes. Sometimes people cannot afford gym membership and are sometimes shy to work out in gym and use weights. On the other hand, sometimes people can afford gym and trainers but because of tight schedule and inconsistency they are not able to remove time for their body and fitness. Thus, we aim to build an AI-based trainer that would help everyone to exercise more efficiently in their own homes in their own comfort.

The project focuses on creating an AI to help you exercise, by determining the quality and quantity of repetitions which is done by using pose estimation. This project is intended to make exercise more easy and more fun. We are going to see an overview of this project, the algorithms used, its advantages, disadvantages, its efficiency as compared to other existing technologies, applications and possible future work.

II. Proposed System

2.1 Proposed System

There are many fitness apps available in market which allows user to keep track of their health and provide a workout plan to user to achieve their personal fitness goal. These app have many functions that provide user with only data to user it does

not actually provide a space where user can do their work out without going to gym or having a personal trainer. These apps do not overview whether the user is actually doing the exercise and if they are doing correctly.

To overcome the above disadvantage of all existed system we introduce our project AI Fitness trainer. This project is can defined as AI which detects our human body during our workout and gives critical feedback on our exercise form and does rep count so user can put all their focus and energy in the doing our workout rather than keeping the count of our exercise.

The system overcome the important disadvantage of not being able to workout at home and at any time without guidance. The system provides us the opportunity to work out anywhere, anytime with guidance so we can do effective workout. System uses computer vision technology to execute the functions of our system. The system uses state-of-the-art pose detection tool known as "BlazePose" from "MediaPipe" to detect the form of the user's body during the workout. OpenCV is used to mark an exoskeleton on the user body and display reps count on the screen. System uses different videos and photos of specific exercise to create 1000 of data of landmarks keypoints and save as csv file use that dataset to train the ML model of that specific exercise using machine learning classification algorithm named random forest classifier.

2.2. System overview

As shown in Figure 2.1 the proposed system consists of three main parts which gives us and running AI fitness trainer which help user have an efficient and effective workout. On frontend application displays the six exercises, namely "Squats", "Bicep Curls", "Jumping Jacks", "Shoulder Press", "Static Lunges", "Bridge". From above mentioned exercises user can choose which exercise they want to do. After they chose the exercise, they land on to the exercise page where user can see steps to do the said exercise and even have video link present if they want to watch before doing the exercise. When user is ready to start, they need to press start button feed of their webcam will be shown on the screen.

Next part is processing the real time video coming from webcam from the device user is using and to render it in a way so each frame is sent to the program for further analysis of the accuracy of the exercise. System uses a very accurate pose detection module form MediaPipe known as blazepose. The MediaPipe pose estimation tool uses a 33 keypoints approach wherein it detects the key points and accordingly uses and sends the data further for processing. It tracks users' movement from the real-time camera frame by using the blazepose tool that has a Machine Learning approach in pose detection. OpenCV is used to display the 33 keypoints exoskeleton on the with colorful lines and it also shows the rep count of the same exercise. After coordinates of desired keypoints are procured then they are sent to ML module that is trained for the said exercise and finds the accuracy of whether the user is doing it correctly or not if the accuracy is less than 0.9 then the users form is incorrect which further use feedback function to give user constructive feedback so user can correct their form and increase the accuracy above or equal to 0.9 so that it can increase the rep count by 1. When user is finish with the workout, they can press stop button and if want to go to homage they can press back button.

Last part of the system is where all data is stored like csv files which contain over 1000 of landmark point coordinates which have been extracted from the video of the exercise and are manually cleaned to be stored. This data is used by the ML algorithm Random Forest Classifier to create an ML module of that exercise which is deployed when the system needs to check the accuracy of users form and these modules are stored as ". pkl" file extension.

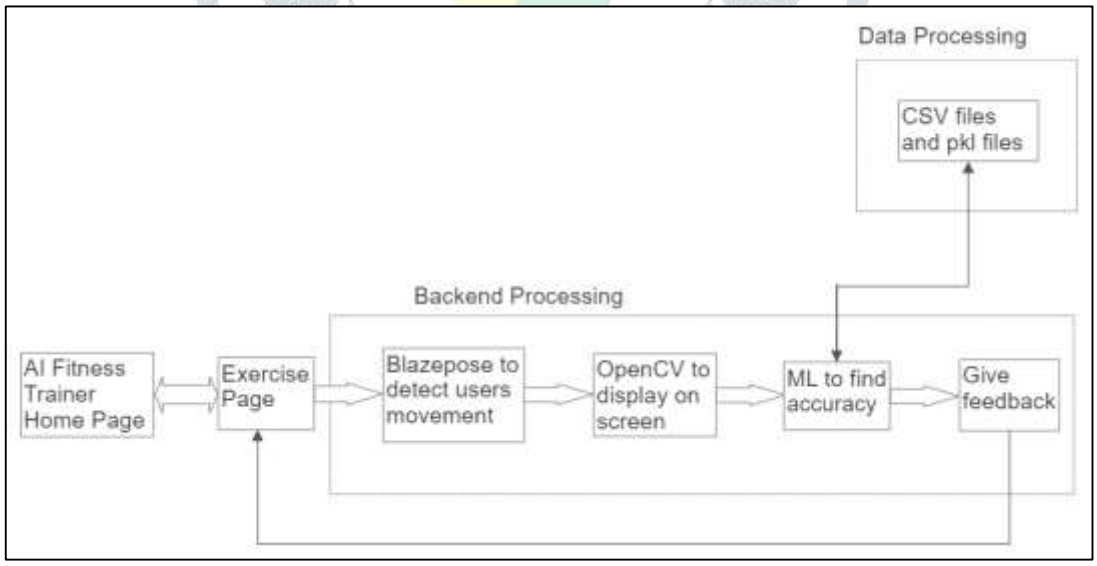


Fig 2.1: System Architecture

III. Implementation

3.1 User Interface Page

To give the user easy access to our AI fitness trainer program module and not have to worry about writing a command to run that program end up with an ugly window pop up in which the program runs, so we made an attractive and engaging User Interface.

Application uses pythons most commonly used standard library Tkinter for making the UI for desktop app. When python combined with the Tkinter provides a fast and easy way to create an GUI for a desktop application. Tkinter provides a powerful object-oriented interface for our desktop application to run on.

3.2 Algorithm

Application uses OpenCV library of python to capture the real-time webcam feed of the user. Then the single frame from that real-time feed is used to detect human pose using" BlazePose" tool from "MediaPipe". The MediaPipe pose estimation tool uses 33 key points approach wherein it detects the key points and accordingly uses them to study the data set to estimate the pose. It tracks the pose from the real-time camera frame by using the blaze pose tool that has a Machine Learning approach in pose detection. The current standard for human body pose detection is the COCO topology, which consists of 17 landmarks across the torso, arms, legs, and face. However, with the BlazePose tool, we present a new topology of 33 human body keypoints, which is a superset of COCO, BlazeFace and BlazePalm topologies. This allows the system to determine body semantics from pose prediction alone that is consistent and precise.

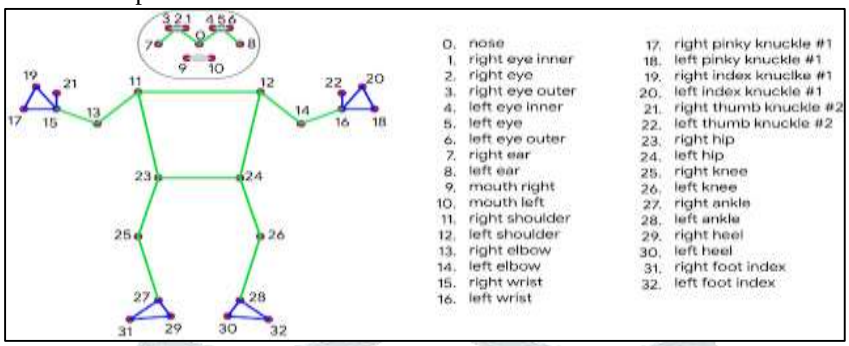


Fig 3.2 33 Human Body Key-points

The dataset which is prepared is then used by the MediaPipe blazepose tool to capture the pose and then it is saved with a label in model file using pickle library of python. This data is then used to train the ML model using classification algorithm named "Random Forest Classifier". A random forest classifier is an estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy.

Application uses users real-time webcam feed to extract their pose keypoints and pass it into the ml model which gives system the estimation based on the label which is given to it while training the model. If the model gives an accuracy rate between 0.9 to 0.92 or greater it counts as up or down based on the label. To calculate a rep, system consider a cycle from up position to down position. System also uses a critique model which have been trained using the same classification algorithm given above to check users form during the workout and if it's below 0.9 accuracy then it gives constructive feedback to the user to correct their form.

3.3 Dataset

Application's dataset consists of data that are collected using both real-time webcam feed and pre-recorded videos. Because it gives more accurate results while training our Machine Learning model. The dataset contains hundreds to thousands of sample landmark keypoints for the Machine Learning model to recognize them. The dataset contains 20,000+ coordinates of keypoints of 6 different exercises that we present the user. The dataset is split into 0.7:0.3 ratio, where 70% of the total pictures is used for training the ML model and 30% to test the performance of the ML model as it is training.

3.4 Methodology

We now describe the AI Fitness Trainer application from a technical perspective as a pipeline system, consisting of multiple system stages (see Figure 3.2). Pose training starts from the user real-time feed from the webcam of an exercise, and ends with the Pose Trainer application providing specific voice feedback on the exercise that the user has chosen. However, there are no requirements on camera type, but distance from camera and the user needs to be at least where users whole body is visible. Application provides user with six different exercises by collecting 1000 of dataset of those exercise and using that data to train different models for those exercises to have most precision. Application basically first detects the landmark positions on the body in the video with the help of BlazePose tool of MediaPipe. Then it passes the points to the ML model to find the accuracy of users form and to calculate the number of repetitions of the exercise and display the count and give critique on the user's form.

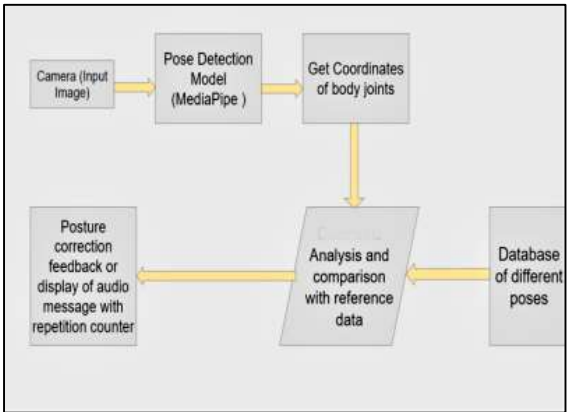


Fig 3.2 Pipeline System

IV. RESULTS AND DISCUSSION

4.1 Results of Descriptive Statics of Study Variables

Table 4.1: Confusion Matrix for Random Forest Classifier

	Precision	Recall	F1 Score	Examples
		Squats		
Correct	0.92	1.00	0.80	72
Incorrect	1.00	0.67	0.89	28
Avg/Total	0.96	0.83	0.84	100
		Bicep Curls		
Correct	0.85	0.89	0.89	54
Incorrect	0.73	0.70	0.80	46
Avg/Total	0.79	0.80	0.84	100
		Jumping Jacks		
Correct	0.89	0.90	0.86	69
Incorrect	0.82	0.74	0.83	31
Avg/Total	0.86	0.82	0.85	100
		Shoulder Press		
Correct	0.67	0.86	0.75	44
Incorrect	0.83	0.62	0.71	56
Avg/Total	0.75	0.73	0.73	100
		Static Lunges		
Correct	0.96	0.75	0.86	64
Incorrect	0.71	0.97	0.84	36
Avg/Total	0.84	0.86	0.85	100
		Bridge		
Correct	0.72	0.86	0.81	49
Incorrect	0.84	0.67	0.71	51
Avg/Total	0.78	0.77	0.76	100

Table 6.1 discusses the precision and recall of each exercise which have been tested and used in the application by using confusion matrix. A confusion matrix is a way to express how many of classifier's prediction were correct, and when incorrect, where the classifier got confused. Confusion matrix not only gives you insight into the errors being made by your classifier but also types of errors that are being made. This breakdown helps you to overcomes the limitation of using classification. Precision shows the accuracy of the positive class. It measures how likely the exercise done is correct. Precision alone is not very helpful because it ignores the negative class. The metric is usually paired with Recall metric. Recall computes the ratio of positive classes correctly detected. This metric shows how good the classifier.

4.2 Accuracy

You have a classifier that takes test examples and hypothesizes classes for each. On every test example, its guess is either right or wrong. You simply measure the number of correct decisions your classifier makes, divide by the total number of test examples, and the result is the accuracy of your classifier. Accuracy is one metric for evaluating classification models. Informally, accuracy is the fraction of predictions our model got right. Formally, accuracy has the following definition:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

For binary classification, accuracy can also be calculated in terms of positives and negatives as follows:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

Where TP = True Positives, TN = True Negatives, FP = False Positives, and FN = False Negatives.

$$\text{Accuracy} = \frac{28+72}{28+72+18+1} = 0.89$$

Therefore, accuracy of the system is calculated to be 0.89 which means the system is 89% accurate while checking user's exercise form

4.3 Results

Whenever you open the application, you have to first select the exercise which you have to perform for your workout from six exercise displayed on the screen. You can select only one exercise at a time, once you have selected the exercise then you will land on the selected exercise page. After that when you are ready after placing the system having webcam you have to click on the start button below. After clicking the start button, the application will open the camera to capture your whole body. Once the webcam feed can be seen on screen with exoskeleton on it you can go ahead and start your workout. During your

workout the app will also count your reps and display on screen and will also give you voice critique feedback. After you have done with that exercise you can stop the clicking on the stop button, the application will close the real-time feed of your webcam and also stop the reps counter so you can check how many reps you have done. Each page of exercise has detailed instruction with visual representation of how to do that exercise. Also, there is a back button that will take you on the first page so you can again select the desired exercise you want to do.

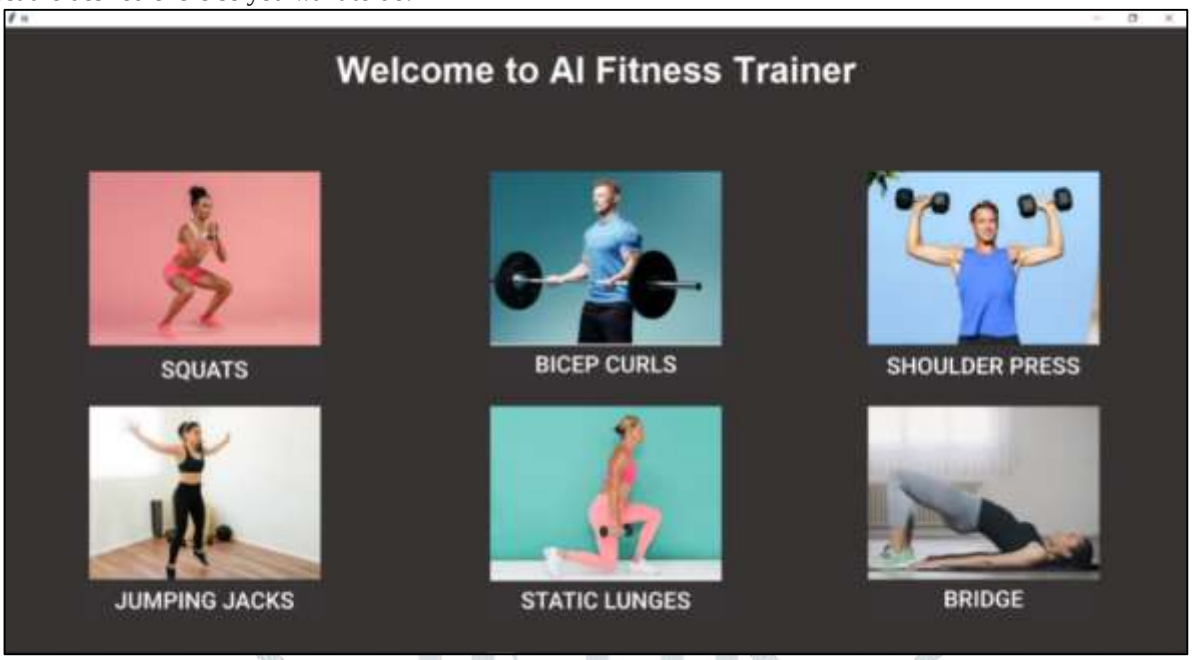


Fig 4.1 Screenshot of the Home Page

Figure 4.1 shows the homepage of the application which gives the user choice to select the exercise they want to do from six exercise given.

Below Figures present the first exercise Squats process and similar User interface are there for six exercises mentioned in Fig 4.1.

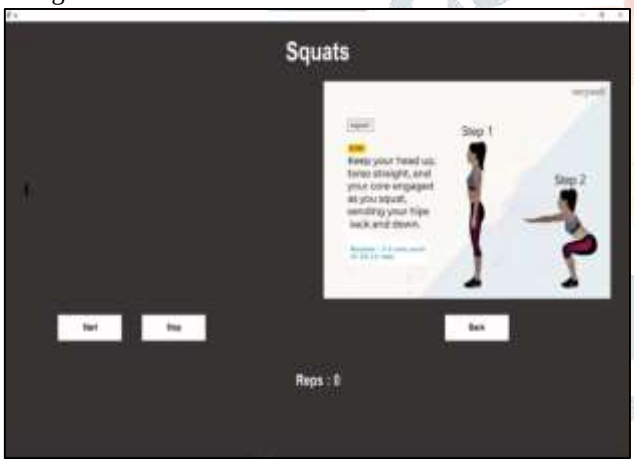
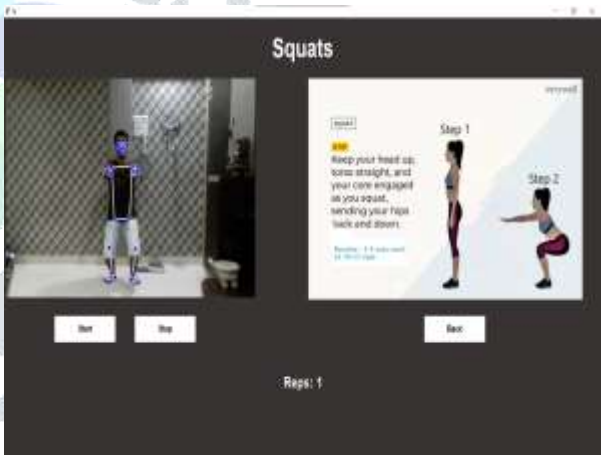


Fig 4.2 a. Squats Exercise Page



4.2. b. Squats Exercise Page after pressing start button

Figure 4.2.a shows the starting page of Squats Exercise. There are three buttons named as 'START', 'STOP' and 'BACK'. Before starting the exercise there are steps to be followed mentioned in written as well as visual format with its routine. It also shows REP COUNT which is initially at zero.

As shown in the Figure.4.2. b user has pressed the START button on Squats exercise page which simultaneously starts the webcam. The application maps the user's body with 33 keypoints and checks the accuracy of user's pose and increase the rep count accordingly. The application has a feature which gives verbal critique for user to correct the pose.

V. Conclusion and Future Scope

5.1 Conclusion

These days our life is becoming busier and that we hardly find time in our schedules to be healthy and fit and exercise daily. This ends up with many health issues. Our main motive is to spread the importance of good health and fitness among common people and help them to achieve it. Implementation of Artificial Intelligence and Machine Learning in the field of fitness can solve many problems. The fitness applications and devices are making our lives easier and eases our fitness journey. Individuals can use this application to do their own workouts at home, hence making them more efficient are less error-prone. During this process, we learnt how to use the many python libraries and package and how the application of machine learning can be beneficial to humans.

In this report, we introduce AI Fitness Trainer, an end-to-end computer vision application that uses pose estimation, and machine learning to provide personalized feedback on fitness exercise form. We use the output of pose estimation to evaluate real-time webcam feed of user doing exercise through human pose keypoints. We work with six different exercises to provide personalized feedback on specific exercise improvements, as well as machine learning algorithms to automatically determine posture correctness using only labeled input videos. The application monitors the user in real-time keeping track of the quality repetitions of a particular exercise, thus keeping his form intact and correct throughout their workout. This will educate amateurs about different exercise routines and their correct postures to prevent injuries

We also have tested the working of our random forest classifier algorithm using confusion matrix to find the precision and recall of each exercise. Further we also discuss the accuracy of the same classifier and how it effects the accuracy of exercises.

5.2 Future Scope

There is a lot of scope of development in this project like it can be upgraded to support more exercises. The data collected by the AI Fitness Trainer can be saved and processed for the next sessions. The trainer will suggest you work out plan and its intensity according to your body type and weight. This application can be developed into a complete android/iOS application for ease of use. The application can not only be used at homes but by increasing the scope can be used in gyms as smart trainers thus reducing the human intervention. Further application will also be able to provide personalized workout and diet plans. This application will offer different styles of yoga to give user an overall fitness regime. AI powered solutions in smart clothing and wearables can make considerable development. garments and other things that can be worn consist of sensors linked to systems that help to accurate biomechanics (e.g., golfer swinging), enhance athlete's performance metrics and improve health.

5.3 Limitations

1. The application can estimate the poses and count repetitions for a limited number of exercises as pose estimation using computer vision for some exercises and postures can be difficult.
2. The application is developed as a cross-web application and is not used as a mobile android/iOS application.
3. The application cannot capture multiple people in the frame in the real-time system.

REFERENCES

[1] S. Jin, L. Xu, J. Xu, C. Wang, W. Liu, C. Qian, W. Ouyang and P. Luo, “Whole-Body Human Pose Estimation in the Wild”, In book: Computer Vision – ECCV 2020 (pp.196-214).
[2] G. Taware, R. Agarwal, P. Dhende, P. Jondhalekar and Prof. S. Hule, “AI Based Workout Assistant and Fitness Guide”, International Journal of Engineering Research & Technology (IJERT) Vol. 10 Issue 11, November-2021.
[3] F. Zhang, V. Bazarevsky, A. Vakunov, A. Tkachenka, G. Sung, C.L. Chang and M. Grundmann, “MediaPipe Hands: On-device Real-time Hand Tracking.” ArXiv, 2020.
[4] Y. Kartynnik, A. Ablavatski, I. Grishchenko, and M. Grundman, “Real-time facial surface geometry from monocular video on mobile gpus.”, IEEE/CVPR Workshop on Computer Vision for Augmented and Virtual Reality, July 2019.
[5] S. Kreiss, L. Bertoni and A. Alahi, “PifPaf: Composite Fields for Human Pose Estimation”, IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2019.
[6] M. Eichner, M. Marin-Jimenez, A. Zisserman, and V. Ferrari, “2d articulated human pose estimation and retrieval in (almost) unconstrained still images”, International Journal of Computer Vision Vol. 99, September 2012.
[7] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang and M. Grundmann, “BlazePose: On-device Real-time Body Pose tracking”, ArXiv, June 2020.
[8] J. Hruthika, P.K. Chaitanya, and G.S. Chaithanya, “Deep Learning Based Human Pose Estimation Using OpenCV”, Int. j. innov. eng. res. technol., vol. 7, no. 12, pp. 246–253, December 2020.
[9] V. Bazarevsky, Y. Kartynnik, A. Vakunov, K. Raveendran and M. Grundmann, “BlazeFace: Sub-millisecond Neural Face Detection on Mobile GPUs”, ArXiv, July 2019.
[10] A. Newell, J. Deng and Z. Huang, “Associative embedding: End-to-end learning for joint detection and grouping”, 31st Conference on Neural Information Processing Systems (NIPS 2017), pp. 2274–2284, December 2017.
[11] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake, “Real-time human pose recognition in parts from single depth images”, The 24th IEEE Conference on Computer Vision and Pattern Recognition, pp. 1297-1304, June 2011.
[12] C. Zheng, W. Wu, T. Yang, S. Zhu, C. Chen, R. Liu, J. Shen, N. Kehtarnavaz and M. Shah, “Deep Learning-Based Human Pose Estimation”, J. ACM, Vol. 37, No. 4, Article 111, August 2018.