VIRTUAL FITNESS TRAINER USING AI

Rohit Tukaram Jagadale*1, Akib Shakil Faras*2, Shreyas Praveen Sonar*3

*1,2,3 Student, Department Of Computer Engineering, Navsahyadri Education Society's Group of Institutions, Pune, Maharashtra, India

ABSTRACT

Obesity, a prevalent issue affecting numerous individuals globally, is often attributed to a sedentary lifestyle. Research indicates that maintaining fitness is crucial for promoting a healthy way of living and is used to assess one’s health-related quality of life. While engaging a fitness trainer can be an effective approach to encourage regular exercise and overall well-being, it may not always be feasible or affordable in certain situations. It is worth noting that exercise has numerous health benefits, but if performed incorrectly, it can be both ineffective and potentially hazardous. Individuals who work out without proper supervision often make mistakes such as using improper forms, which can lead to severe consequences, such as hamstring injuries or falls. In our project, we introduce the Virtual Fitness Trainer web based application, designed to identify the user’s exercise poses and offer personalized, detailed recommendations for improving their form. The Pose Trainer utilizes cutting-edge pose estimation technology called “BlazePose” from “MediaPipe” to detect the user’s pose and assess their exercise performance, providing valuable feedback. The Virtual Fitness Trainer supports six common exercises and can be utilized on Windows or Linux computers equipped with a GPU and a webcam.

Keywords: Virtual Fitness Trainer, Machine Learning, Pose detection, BlazePose, Health, Workout.

I. INTRODUCTION

Virtual assistants have become an integral part of our daily lives, playing a significant role in various activities. The field of AI has emerged as a promising area of exploration, and our project aims to leverage this technology in the form of an AI-based workout trainer. Introducing the Virtual Fitness Trainer, a web based application designed to detect users’ exercise poses, track repetitions, and provide recommendations for improving their form. To achieve this, we utilize the BlazePose tool from MediaPipe for pose detection during workout sessions. By analyzing the pose’s form using real-time video and comparing it to a dataset, the Virtual Fitness Trainer accurately counts repetitions for a specific exercise. This application caters to individuals who may be unable to afford gym memberships or feel uncomfortable working out in public settings. It also benefits those who have access to gyms and trainers but struggle with scheduling and consistency, allowing them to exercise more efficiently in the comfort of their own homes. The project’s primary focus is on creating an AI-based trainer that enhances the exercise experience by assessing the quality and quantity of repetitions through pose estimation. This project aims to make exercise easier and more enjoyable. In this overview, we will delve into the algorithms employed, examine the advantages and disadvantages, assess its efficiency in comparison to existing technologies and applications, and explore potential future enhancements.

II. LITERATURE SURVEY

In the paper[1] they propose a method to build a smart gym trainer using human pose estimation. They use the following methods -

1) Obtain the model weights, which can be obtained from public sample datasets available online. Then load the model weights into the network.

2) The system must read the input video as a set of frames that are then used as input to the network. Then the system then makes predictions and identifies key points.

3) The system generates a skeletal structure or stick figure based on the identified key points. Then the angles between various limbs and other body parts can be determined using the skeletal structure or stick figure. Based on the angles that were determined, instructions such as the angle at which the user’s arm is bent can be provided.

4) A graph can then be generated based on the angles of joints and their movements, which can be used to compare the user’s graph to the ideal graph of an athlete.
III. LIMITATION OF THIS PAPER

1) There is no proper implementation for particular exercises
2) No exercise Repo counter mechanism
3) No progress bar for exercise

In the paper[2], they propose a method for efficiently detecting the 2D poses of multiple individuals in an image. This method employs a non-parametric approach called Part Affinity Fields (PAFs), which learns to associate body parts with individuals in the image. The model utilizes global context and allows for a greedy bottom-up parsing step to maintain high accuracy and achieve real-time performance, regardless of the number of people present in the image. The architecture is designed to learn part locations and associations simultaneously via two branches of the sequential prediction process. Their method outperformed all competitors in the COCO 2016 key points challenge and significantly surpassed the previous state-of-the-art result in both performance and efficiency on the MPII Multi-Person benchmark. In the paper[3], the researcher's goal is to develop an on-device, single person-specific human pose estimation model that can facilitate various performance-demanding applications such as Sign Language, Yoga/Fitness tracking, and AR. This model operates in near-real time on a mobile CPU and can be accelerated to super-real time latency on a mobile GPU. With its 33 key point topology consistent with Blaze Face and Blaze Palm, it serves as a foundation for subsequent hand pose and facial geometry estimation models. Our approach native scales to a larger number of key points, 3D support, and additional key point attributes because it is not reliant on heat maps/offset maps, which require an additional full-resolution layer for each new feature type. In the research paper[4], the researchers introduced an efficient solution in their research paper to tackle the challenge of detecting poses when there are multiple people in a real-time frame. Their approach involves training the model to detect the points of the user and segregating them based on the affinity of different points in the frame, using a bottom-up approach that is highly accurate and efficient in terms of performance, regardless of the number of people in the frame. Their method outperformed other approaches by 8.5% mAP on a dataset of 288 frame images, achieving higher accuracy and precision in real-time. Unlike previous solutions, this approach did not require definition in the training stages. However, OpenPose has a disadvantage in that it does not provide any data on the depth, and it also requires significant computing power. According to the findings of a recent survey conducted by Better[6], the primary reasons cited by individuals for not attending the gym are a shortage of time and a lack of confidence. Training fees are also a key factor regarding why people avoid the exercises and going to the gym. The study indicates that male gym-goers are more inclined to shell out more money on memberships, comprising 68% and 74% of those paying £61-80 and £80 or more per month, respectively. In contrast, the majority of women opt for the more modest £16-25 price range. In a country like India, following survey report we get

IV. PROPOSED SYSTEM

There are numerous fitness applications available in the market that enable users to monitor their health and receive personalized workout plans to achieve their fitness goals. However, these apps primarily focus on providing data and lack a dedicated space for users to perform their workouts without going to a gym or having a personal trainer. Additionally, they do not provide oversight to ensure users are performing exercises correctly. To address these limitations, we present our project, the Virtual Fitness Trainer. The Virtual Fitness Trainer is an innovative system designed to detect and analyze human body movements during workouts, offering crucial feedback on exercise form and automatically counting repetitions. This allows users to concentrate on their workouts rather than keeping track of counts. One key advantage of our system is the ability to exercise at home or any desired location without the need for constant guidance. The system leverages computer vision technology to execute its functions. Specifically, it utilizes the state-of-the-art pose detection tool called "BlazePose" from "MediaPipe" to accurately detect and analyze the user's body form during workouts. OpenCV is employed to mark an exoskeleton on the user's body and display the repetition count on the screen. By combining computer vision, pose detection, and machine learning techniques, our Virtual Fitness Trainer overcomes the limitations of existing systems, providing users with the flexibility to work out effectively anytime and anywhere with guidance and feedback.
V. SYSTEM OVERVIEW

The front-end application showcases a selection of six exercises: "Squats," "Curls," "Jumping Jacks," "Push Ups," "Lateral Rise," and "Pull Ups." Users have the option to choose any exercise they prefer. Upon selecting an exercise, users are directed to a dedicated exercise page where they can find detailed instructions and, if desired, a video demonstration of the exercise. When users are prepared to begin, they can initiate the exercise with the help of a webcam which displays the live feed on the screen. The subsequent stage involves processing the live video stream captured from the user's device webcam and rendering it in a manner that allows each frame to be analyzed for exercise accuracy. The system utilizes a highly precise pose detection module called "Blazepose" from MediaPipe. This pose estimation tool employs a 33-key-point method to detect and transmit key-point data for further processing. By utilizing the Blazepose tool, which leverages machine learning techniques, the system tracks the user’s movements from the real-time camera feed. OpenCV is employed to display the colorful lines representing the 33-key-point exoskeleton, along with displaying the repetition count for the corresponding exercise. With the same system, suggests possible next movement which users have to do and also gives feedback for any wrong exercise pose.

![Image 1: System Architecture](image1.jpg)

![Image 2: Block Diagram](image2.jpg)

**Implementation And Algorithm:**

1] User Interface:

To provide users with convenient access to our Virtual fitness trainer program module without the need to manually execute commands or encounter an unappealing window pop-up, we have developed an appealing and immersive User Interface (UI). This UI ensures a seamless and visually pleasing experience for users interacting with the program. We utilize the Flask Python framework in conjunction with HTML, CSS, and Bootstrap to construct a user interface. Flask is a powerful and adaptable web framework that empowers developers to rapidly create web applications with minimal overhead. It provides the fundamental elements for web development, such as URL routing, request handling, and response generation. Furthermore, developers can augment its functionality by incorporating third-party extensions, allowing for customization to meet
specific requirements while avoiding unnecessary complexity. The project consists of two main sections: the home page and individual exercise pages. The index.html page serves as the home page for our website, featuring four sections: "Exercises," "About," "Team," and "Contact Us." When a user wishes to perform an exercise, they can select from the available options in the exercises section [image 4] on the home page. This action will open the corresponding exercise page, which displays a live camera feed and provides guidance on executing the specific exercise with proper posture. As the user initiates the exercise, a pose estimation tool utilizes a 33-key-point method to detect angles between different joints. It provides the repetition count, suggests the next possible movement for the user, and offers feedback on any incorrect exercise poses.

Image 3 Home Page

Image 4 Exercise Section

2] Algorithm:
The application utilizes the OpenCV library in Python to capture the live webcam feed of the user. From this real-time feed, a single frame is extracted and processed using the "BlazePose" tool from "MediaPipe" for human pose detection. The MediaPipe pose estimation tool employs a 33-keypoint approach, which involves detecting and analyzing key points to estimate the pose. By leveraging the BlazePose tool, which utilizes machine learning techniques, the system tracks the pose of the user from each frame of the live camera feed. Unlike the standard COCO topology, which includes 17 landmarks across the torso, arms, legs, and face, the BlazePose tool introduces a new topology of 33 human body key-points. This topology encompasses COCO, BlazeFace, and BlazePalm typologies, allowing the system to accurately determine body semantics solely based on pose predictions. This enhanced topology ensures consistent and precise pose estimation.
The implemented solution employs a two-step detector-tracker machine learning pipeline. In the first step, the detector component identifies the region-of-interest (ROI) within each frame to detect the presence of a person/pose. Following that, the tracker component utilizes an ROI-cropped frame as input to predict the pose landmarks and segmentation masks within the identified ROI. When processing video data, the detector is run only for the initial frame, and subsequent frames are processed using the tracker alone, unless the confidence level falls below a specified threshold. In such cases, the detector is reactivated to ensure accurate detection.

VI. METHODOLOGY

The Virtual Fitness Trainer application can be viewed as a pipeline system, we will now discuss from a technical standpoint. The process begins with capturing the user's real-time feed from their webcam while performing an exercise. It concludes with the Pose Trainer application delivering feedback and Number of repetition based on the selected exercise. It’s important to note that the application does not impose any specific requirements on the camera type. However, it is necessary for the user to maintain a sufficient distance from the camera to ensure their entire body remains visible throughout the exercise.

Calculating the angle between two joints -

\[
\text{radians} = \text{np.arctan2}(c[1] - b[1], c[0] - b[0]) - \text{np.arctan2}(a[1] - b[1], a[0] - b[0])
\]

\[
\text{angle} = \text{np.abs}(\text{radians} \times 180.0 / \text{np.pi}) \text{if angle} > 180.0: \text{angle} = 360 - \text{angle}
\]

Here, the code calculates the angle between the vectors formed by points a, b, and c. The np.arctan2 function is used to compute the arctangent of the differences in y and x coordinates of the vectors. The result is stored in radians. Then, the absolute value of radians multiplied by 180.0/np.pi is assigned to an angle to convert the angle from radians to degrees. Lat two lines ensures that the angle returned is within the range of 0 to 180 degrees. Once we obtain the specific angle between two joints, we utilize this angle to measure the number of repetitions and provide feedback for a particular exercise. The logic used for this process is exercise-dependent and tailored to each specific exercise. By applying exercise-specific algorithms and rules, we can accurately determine the number of repetitions performed and offer appropriate feedback based on the angle measurements obtained.
VII. RESULT

As the outcome of the AI-powered virtual fitness trainer application, users receive rep counts and recommendations for the next steps specific to the exercise they have chosen from the exercise section[Image 4]. Figure 5 illustrates the results of the curl exercise. At the top of the figure, a live camera feed displays the user, showing the number of reps completed and offering suggestions for the subsequent steps. On the right side of the figure, a video is presented, demonstrating proper technique and postures to enhance or perform the exercise correctly.

Image 7 Curl Exercise

VIII. ADVANTAGE

1) In the market, there are several applications available that provide guidance on exercises. However, our application goes beyond simply recommending exercises. It also utilizes computer vision to guide users on correct posture and accurately count repetitions.

2) Our application monitors users in real-time, ensuring they maintain proper form and perform quality repetitions throughout their workout. This real-time monitoring helps educate beginners on various exercise routines and correct postures to prevent injuries.

3) The application's versatility allows it to be utilized not only by individuals at home but also in gyms as a smart trainer. This reduces the need for constant human intervention.

4) Our primary goal is to raise awareness about the importance of good health and fitness among the general public.

IX. LIMITATIONS

1) Accuracy limitations: Although Mediapipe's pose estimation is quite accurate, it can be affected by environmental factors such as lighting, clothing, camera quality, and more.

2) Range of motion limitations: Mediapipe's pose estimation works best when the person being analyzed is standing upright and facing the camera, limiting its effectiveness for detecting poses in more complex movements like acrobatics or certain yoga poses.

3) Limited joint detection: Mediapipe's pose estimation only detects certain joints like the elbows, wrists, and knees, which may not be sufficient for certain exercises that require the detection of other joints or body parts such as the hips or ankles.

4) Applicability limitations: Mediapipe's pose estimation was primarily trained on data from people of certain age, gender, and body types, making it less effective for people outside of these parameters, such as children, the elderly, or individuals with disabilities.

5) Depth perception limitations: While Mediapipe's pose estimation can detect the positions of joints in 2D space, it cannot detect depth, which can limit its applicability to exercises or movements that require depth perception, such as weightlifting or certain yoga poses.
X. FUTURE SCOPE

The possibilities for advancing exercise technology are virtually endless. One such innovation that holds tremendous promise for the future is the ability to detect human pose in three-dimensional space. While this technology is currently limited by 2D pose estimations, frameworks like Unity 3D may hold the key to unlocking its full potential. By converting 2D pose estimations into 3D, we could derive more exercises that were previously not feasible. Of course, this would require significant advancements in computational power and expertise. However, given the exciting potential of this cutting-edge software, it's only a matter of time before we see its development come to fruition. As such, we anticipate that this technology will play a crucial role in the evolution of exercise technology, paving the way for an even more diverse range of exercises and fitness programs.

XI. CONCLUSION

In our modern, hectic lives, finding time to prioritize our health and engage in regular exercise has become increasingly challenging. This lack of focus on fitness often leads to various health issues. Our primary objective is to raise awareness about the significance of good health and fitness among the general population and assist them in achieving their wellness goals. By harnessing the power of Artificial Intelligence (AI) and Machine Learning (ML) in the realm of fitness, we can address many of these challenges. Fitness applications and devices have simplified our lives and streamlined our fitness journeys. These tools empower individuals to conveniently perform workouts at home, increasing efficiency and reducing the risk of errors. Throughout this process, we have acquired knowledge on utilizing various Python libraries and packages and have witnessed the immense benefits that machine learning can offer in improving human well-being.

XII. REFERENCES