YOGA POSE DETECTION AND CORRECTION

A PROJECT REPORT

Submitted by

RASIKA V V (TKM20MCA-2029)

to

The APJ Abdul Kalam Technological University

In partial fulfillment of the requirements for the award of the Degree of

MASTER OF COMPUTER APPLICATIONS

Thangal Kunju Musaliar College of Engineering
Kerala
DEPARTMENT OF COMPUTER APPLICATIONS

JULY 2022

DECLARATION

I undersigned hereby declare that the project report YOGA POSE DETECTION AND CORRECTION, submitted for partial fulfillment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by me under supervision of Prof.NATHEERA BEEVI M . This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Place: Kollam

Date: 26-07-22

RASIKA V V

DEPARTMENT OF COMPUTER APPLICATIONS

TKM COLLEGE OF ENGINEERING

C E R T I F I C A T E

This is to certify that, the project report entitled **"YOGA POSE DETECTION AND CORRECTION"** is submitted by **RASIKA V V (TKM20MCA-2029)** to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Master of Computer Applications,is a bonafide record of the project work carried out by her under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Internal Supervisor Head of the Department External Examiner

ACKNOWLEDGEMENT

First and foremost I thank GOD almighty and my parents for the success of this project. I owe sincere gratitude and heart full thanks to everyone who shared their precious time and knowledge for the successful completion of our project.

I am extremely grateful to **Dr.Fousia M Shamsudeen**, Head of the Department, for providing us with best facilities.

I would like to thank our coordinator, **Prof. Vaheetha Salam** and project guide **Prof. Natheera Beevi M**, Department of Computer Applications, who motivated me throughout the project.

I profusely thank all other faculty members in the department and all other members of TKM College of Engineering, for their guidance and inspirations throughout our course of study.

I owe my thanks to our friends and all others who have directly or indirectly helped us in the successful completion of this project.

RASIKA V V

ABSTRACT

Nowadays, yoga has become a part of life for many people. Exercises and sports technological assistance is implemented in yoga pose identification .Self-learning is an integral part in Yoga, but incorrect posture while performing yoga can lead to serious harm to muscles and ligaments of the body. Thus to prevent this I present an intuitive approach based on machine learning techniques to correct the practitioner's pose while performing various yoga asanas.

The proposed system is aimed at providing concise feedback to the practitioner so they are able perform yoga poses correctly and assist them in identifying the incorrect poses and suggest a proper feedback for improvement in order to prevent injuries as well as increase their knowledge of a particular yoga pose. By using such a system, a whole yoga class atmosphere can be created at the user's home where the yoga pose is detected and corrected by the system automatically using mediapipe algorithm. It extracts body landmarks of each of the keypoints to enable angle calculation of each body joint and provide necessary instructions to user to correct the yoga pose.So that proper instructions can be delivered to help them in very convenient manner so that they can adjust their poses and learning about their incorrectness in real time. There is also an increasing demand for the development of computer-assisted training systems from the practitioner side to help them to improve their knowledge and understanding about different types of yoga poses and also protect them from injury that might occur during learning phases.

Contents

List of Figures

Chapter 1

INTRODUCTION

Yoga, which originates from India, is a form of healthy exercise that encourages harmony between the body (physical) and mind (mental). It is one of the oldest sciences in the world that is known to be effective for sustaining and preserving physical and mental health, and spiritual evolution . Yoga practicing provides the benefits of improved flexibility, energy levels, sleep, posture, muscle strength, circulatory and cardio health, athletic performance, and reduction of injuries as well as chronic pain. Yoga is effective in treating psychiatric disorders and is more effective than therapeutic exercise for chronic lower back pain. The survey respondents (89.4percent) perceived that yoga aids the relief of joint pain, muscle pain, headaches,and depression . However, improper yoga positions have been shown to cause neck, waist, shoulder, wrist, knee, sprains, and muscle pain. Specifically, a survey of 34 countries in 2007 found that practicing yoga incorrectly generally causes neck injuries . Moreover, of the 46.6percent injuries resulting from poor practice, injuries to the lower limbs (21.9percent), and sprains (45.03percent) were observed from a survey of yoga-related injury cases (2001–2014) . The most common type of injury was found to be a sprain (23/67, 34percent) with the lower extremity (27/67, 42percent), in one Canadian study (1991–2010).

Human activity recognition is a well-established computer vision problem that has imposed several challenges over the years. It is the problem of locating keypoints and the posture of a human body from the sensor data. Activity recognition is useful in many domains including biometrics, video-surveillance, human–computer interaction, assisted living, sports arbitration, in-home health monitoring, etc .The health status of an individual can be evaluated and predicted with the

help of monitoring and recognizing their activities. Yoga posture recognition is a relatively newer application. To obtain the benefits of yoga, it is important that yoga should be practiced in the correct postures and forms, just like any other exercise. Incorrect postures can cause unproductive results to the extent of pernicious effects. This creates the necessity for a Yoga instructor to administer the appropriate individual posture. The availability of Yoga instructors is very limited, and that makes it a costly affair as well. Overall cost and resource availability make it a difficult outreach for Yoga and hence creates the need for technology-based assistance.

In this work, I propose a system that can recognize and correct yoga poses in real-time using various deep learning techniques. By using such a system, a whole yoga class atmosphere can be created at the user's home where the yoga pose is detected and corrected by the system automatically using mediapipe algorithm. MediaPipe is a ML solution for high-fidelity body pose tracking, inferring 33 3D landmarks and background segmentation mask on the whole body from RGB video frames. Finding a yoga class that is affordable and one that aligns with one's schedule is difficult. This project tries to fill that gap of on-demand yoga classes with feedback on the yoga routine in real-time. This software is just like an instructor. Whenever it is required to join any classes or hire an instructor, how the instructor helps. This will be the same as instructor oral guidance to perform any pose. If the postures are wrong, an entity must correct them, and the particular person will perform the correct postures.

1.1 Problem Definition

Self-learning is an integral part in Yoga .The practitioner have to follow certain steps and guidelines for taking maximum benefit from a particular pose and in lot of cases failing to do so may lead to serious harm due to incorrect postures. If these incorrect practices are continued for a prolonged period it may lead to long-term pain in the joint. This became the prime motivation for developing a yoga self-training systems for practitioner to exercise on their own with the help of a web based application, so that proper instructions can be delivered to help them in very convenient manner so that they can adjust their poses and learning about their incorrectness in real time and there is also an increasing demand for the development of computer-assisted training systems from the practitioner side to help them to improve their knowledge and understanding about different types of yoga poses and also protect them from injury that might occur during learning phases.

1.2 Objective

The project's major purpose is to:

- Different yoga postures were collected from camera, and learners were required to perform each yoga posture.
- It can evaluate pose quality; Which can distinguish different movements by analyzing pose characteristics.
- Therefore, automatically evaluating yoga poses and providing suggestions to alert learners using mediapipe algorithm.

Chapter 2

LITERATURE SURVEY

A literature review is an overview of the previously published works on a specific topic. The term can refer to a full scholarly paper or a section of a scholarly work such as a book, or an article. It is a comprehensive summary of previous research on a topic. The literature review surveys scholarly articles, books, and other sources relevant to a particular area of research. The review should enumerate, describe, summarize, objectively evaluate and clarify this previous research. The literature review acknowledges the work of previous researchers, and in so doing, assures the reader that your work has been well conceived. It is assumed that by mentioning a previous work in the field of study, that the author has read, evaluated, and assimiliated that work into the work at hand. A literature review creates a ”landscape” for the reader, giving her or him a full understanding of the developments in the field. This landscape informs the reader that the author has indeed assimilated all (or the vast majority of) previous, significant works in the field into her or his research.

2.1 Purpose of the Literature Review

1. It gives readers easy access to research on a particular topic by selecting high quality articles or studies that are relevant, meaningful, important and valid and summarizing them into one complete report.

2. It provides an excellent starting point for researchers beginning to do research in a new area by forcing them to summarize, evaluate, and compare original research in that specific area.

3. It ensures that researchers do not duplicate work that has already been done.

4. It can provide clues as to where future research is heading or recommend areas on which to focus.

5. It highlights the key findings.

6. It identifies inconsistencies, gaps and contradictions in the literature.

7. It provides a constructive analysis of the methodologies and approaches of other researchers.

2.2 Related Works

L Rupasinghe et al. [1] proposed a Infinity Yoga Tutor application that is able to capture user movements using the mobile camera. The system consists of two main modules, a pose estimation module which uses OpenPose to identify 25 keypoints in the human body, using the BODY25 dataset, and a pose detection module which consists of a Deep Learning model, that uses time-distributed Convolutional Neural Networks, Long Short Term Memory and SoftMax regression in order to analyze and predict user pose or asana using a sequence of frames. This module was trained to classify 6 different asanas and the selected model which uses OpenPose for pose estimation has an accuracy of 99.91percent.

Agarwal et al. [2] worked on a dataset that contained 10 different yoga poses with around 5500 images.It has been observed that pose detection techniques can be used to identify the postures and also to assist the people to perform yoga more accurately. Recognition of posture is a challenging task due to the lack availability of dataset and also to detect posture on real-time bases. To overcome this problem a large dataset has been created which contain at least 5500 images of 10 different yoga pose and used a tf-pose estimation Algorithm which draws a skeleton of a human body on the real-time bases. Angles of the joints in the human body are extracted using the tf-pose skeleton and used them as a feature to implement various machine learning models. 80percent

of the dataset has been used for training purpose and 20percent of the dataset has been used for testing. This dataset is tested on different Machine learning classification models and achieves an accuracy of 99.04percent by using a Random Forest Classifier.

Anilkumar et al. [3] proposed a method in which, when a pose is made by the user in front of his camera,mediapipe library does the geometric analysis based on the frames obtained from the camera. Articulated pose estimation in computer vision is the study of algorithms and systems that recover the pose of an articulated body. In the paper, articulated body pose estimation is the process of estimating the position of human body parts and joints in a given image. The main focus is on building a yoga monitoring system, in which the user's movements and postures are analyzed and tracked for errors in the yoga routine. The user is then notified of his/her error in the posture through a display screen or a wireless speaker. The inaccurate body pose of the user can be pointed out in real-time so that the user can rectify his/her mistakes.

Chen et al. [4] introduced a computer-assisted self-training system, using Kinect, with a user's body contour and star skeleton extraction for asana recognition, which took three yoga postures: tree posture, warrior 3 posture, and downward facing dog posture. They achieved an overall accuracy of only 82.84percent during their experiments.

Kadbhane et al. [5] captured video information using Microsoft Kinect. Kinect has a skeletal pursuit tool that may acknowledge twenty joints of a person's body.For doing Yoga we'd like proper training and also a trainer who will monitor accuracy,body movement.So to boost the popularity accuracy with reduced training times,it prepared Microsoft Kinect to recognize different mutual points of the human body in real-time and from this mutual duration it calculate various angles to live the accuracy of specific yoga poses for a user. They used the chosen ten joint points for calculation. They engineered the reference structure for each yoga position by gathering info of the joint points from human posture. They calculated the cosine similarity of vectors by finding angles between all the vectors connecting any two joint points.

Girija Gireesh et al.[6] proposed AI-Based Yoga Pose Estimation for Android Application.The importance of yoga is renowned worldwide and its health benefits, which were preached by ancient

sages, have stood the test of time. Even though yoga is becoming preeminent, there are important challenges faced while doing yoga such as performing it with incorrect form, the classes being expensive and the shortage of time in our busy lives. Computer vision techniques exhibit promising solutions for human pose estimation. However, these techniques are seldom applied in the domain of health or exercise, with no literature or projects cited specifically for yoga. This paper surveys the various technologies that can be used for pose estimation and concludes the best method based on the usability for an android application. The paper then discusses the methodology that will be used to deploy the yoga pose estimation on an android application, how the app is modeled and the working of each component is explained.

Nagalakshmi et al. [7] applied the learning of 3D landmark points, from a single image, using skinned multi-person linear (SMPL) and an encoded architecture of classification models. These models included KNN, SVM, and other deep learning models such as AlexNet, VggNet, and ResNet. Each model was utilized with the proposed dataset that was based on a collection of 13 different yoga postures. The dataset includes a half-camel posture, standing half-forward bend posture, butterfly posture, cobra posture, bridge posture, sitting posture, standing forward bend posture, child posture, corpse posture, mountain posture, tree posture, triangle posture, and twisted posture. Classification was evaluated with an accuracy of 83

Manisha et al.[8] proposed a new dataset, called Yoga-82, for the fine grained classification of human yoga postures. They modified the DenseNet201 model to classify yoga postures within their proposed Yoga-82 dataset and achieved the top-1 score of image classification performance at 79 percent of 82 multi-classes. The purpose of this study was to develop a yoga posture coaching system based on transfer learning. First collected yoga posture data for 14 yoga posture classifications and proposed a yoga posture coaching system, using an interactive display, based on transfer learning that pre-trained weights were from the CNN architecture model to recognize a yoga posture in real time. Furthermore, our yoga posture coaching system provided posture instruction feedback when a user performed yoga in front of a yoga posture coaching system.

Islam et al.[9] proposed a Yoga posture recognition system by detecting human joint points in real time using microsoft kinect.Musculoskeletal disorder is increasing in humans due to accidents

or aging which is a great concern for future world. Physical exercises can reduce this disorder. Yoga is a great medium of physical exercise. For doing yoga a trainer is important who can monitor the perfectness of different yoga poses. This paper proposed a system which can monitor human body parts movement and monitor the accuracy of different yoga poses which aids the user to practice yoga. It used Microsoft Kinect to detect different joint points of human body in real time and from those joint points it calculate various angles to measure the accuracy of a certain yoga poses for a user. The proposed system can successfully recognize different yoga poses in real time.

Kumar Yadav et al.[10] proposed an approach to accurately recognize various Yoga asanas using deep learning algorithms.A dataset of six Yoga asanas (i.e. Bhujangasana, Padmasana, Shavasana, Tadasana, Trikonasana, and Vrikshasana) has been created using 15 individuals (ten males and five females) with a normal RGB webcam and is made publicly available.A hybrid deep learning model is proposed using convolutional neural network (CNN) and long short-term memory (LSTM) for Yoga recognition on real-time videos, where CNN layer is used to extract features from keypoints of each frame obtained from OpenPose and is followed by LSTM to give temporal predictions. To the best of our knowledge, thisis the first study using an end-to-end deep learning pipeline to detect Yoga from videos. The system achieves a test accuracy of 99.04percent on single frames and 99.38percent accuracy after polling of predictions on 45 frames of the videos. Using a model with temporal data leverages the information from previous frames to give an accurate and robust result. We have also tested the system in real time for a different set of 12 persons (five males and seven females) and achieved 98.92percent accuracy.

Chapter 3

METHODOLOGY

3.1 Proposed system

The proposed system takes video sequence frames as input in real time. The output would be the predicted yoga pose along with possible feedback for angle and pose correction. The system consists of four main phases: Object detection, Keypoints extraction, Pose recognition and Pose correction. OpenCV is an open-source library that detect specific objects such as faces, eyes, hand in the real time videos. Keypoints extraction phase does the job of detecting and extracting location of important keypoints based on the user's position using mediapipe. The pose recognition phase recognise which pose it is and classifies if the pose is correct or not. The final phase is pose correction where the user is further given feedback for correction of pose.

In this project, the mediapipe algorithm is used for extracting different coordinates. By using the coordinates, different angles of joints, distance and slops between two-point have been extracted and used as an evaluation parameter for the correction of the yoga poses. Different Mathematics formulas have been used for the correction such Cosine rule, Euclidean distance. After extracting the parameters, a comparison has been done between the extracted and predefined parameters for each joint and slop for justifying the conclusion of the correctness of the posture.

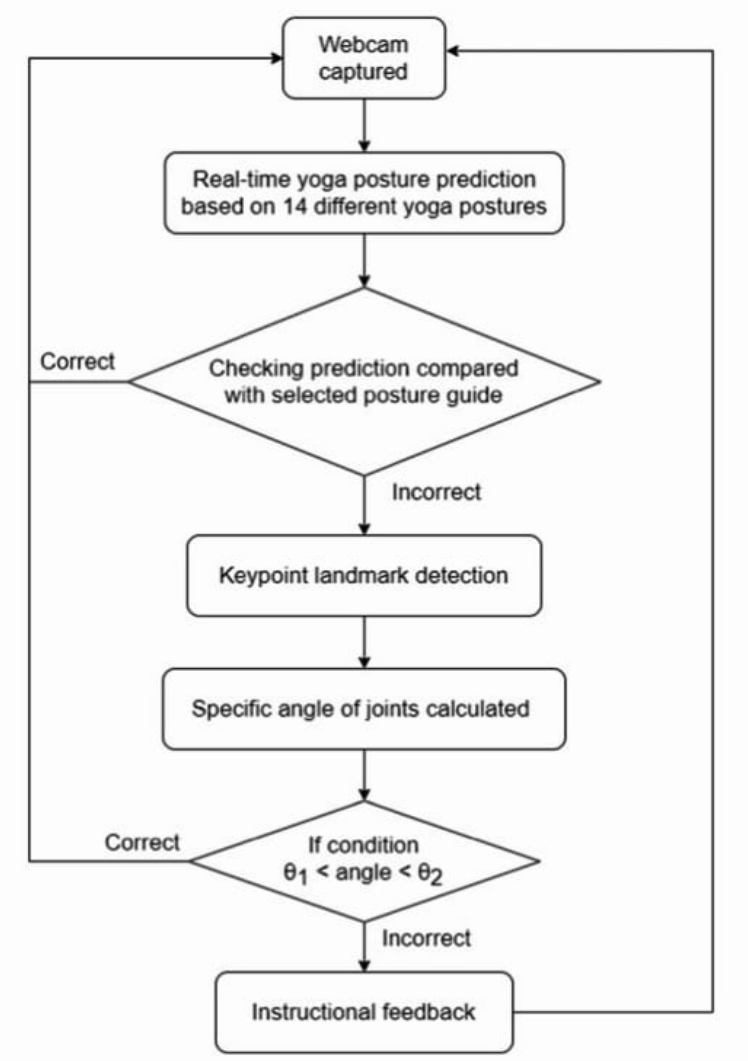


Figure 3.1: Real-time coaching feedback block diagram of yoga self-coaching system

3.2 Opencv

3.2.1 Object detection

Computer vision is a rapidly growing interdisciplinary scientific field devoted to analyzing, modifying, and high-level understanding of images.The computer vision system usually consists of multiple stages of processing, such as application of some simple filters, extractions of objects, analysis of data from extracted objects, data communication, and comparison with an existing pattern.Computer vision is divided into many sub-domains, including object recognition, 3D pose estimation, learning, indexing, and motion estimation. One of the fundamental tasks in computer vision is visual object tracking.

The Open Source Computer Vision (OpenCV) library is an open source cross-platform computer vision and machine learning software library. It was originally developed by Intel to advance CPU-intensive applications in 2000.It gives better output for real-time data.With the help of OpenCV, we can process images and videos so that the implemented algorithm can be able to identify objects such as cars, traffic signals, number plates, etc.This project focus on the single object tracking, in which an object is being tracked even if the environment consists of multiple objects. The library that use along with OpenCV-python is Mediapipe.

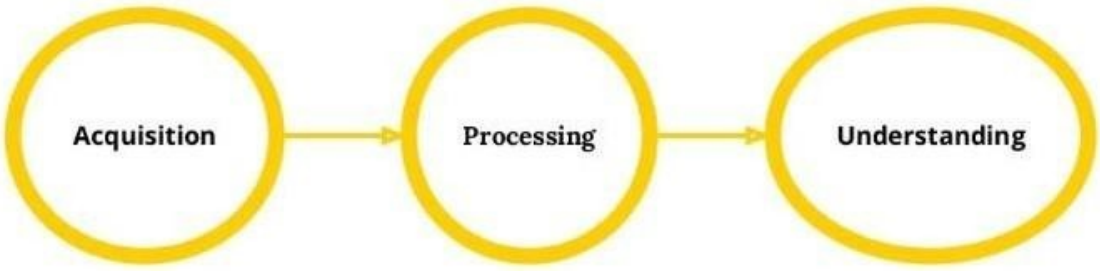


Figure 3.2: Opencv stages

- Read and write yoga images
- Capture and save yoga videos
- Process images

- Perform feature detection
- Detect specific objects such as faces, eyes, hand in the videos or images.
- Analyze the video, i.e., estimate the motion in it, and track objects in it.

3.2.2 Object tracking

In general, tracking an object in the video involves steps such as: a) choosing the tracker, b) selecting the object (target) from the starting frame with the bounding box, c) initializing the tracker with information about the frame and bounding box, and d) reading the remaining frames and finding the new bounding box of the object. The last step is usually implemented in the loop.

3.3 Mediapipe

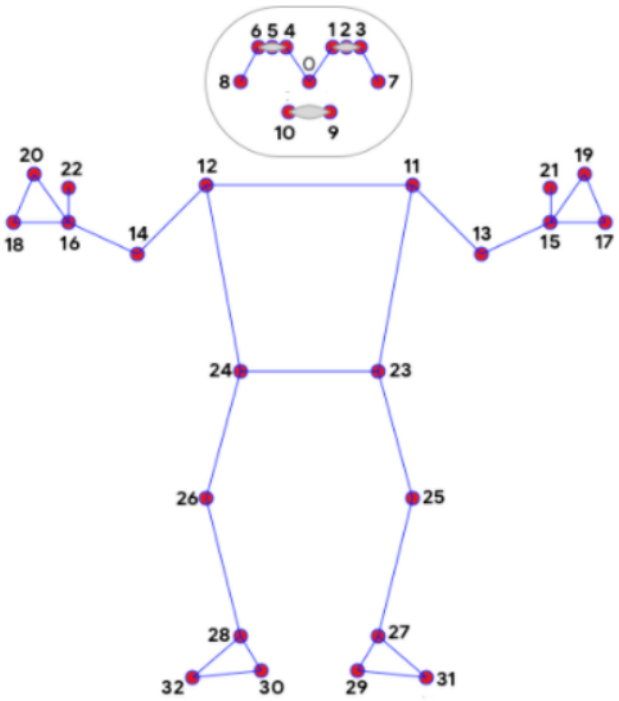
3.3.1 Pose estimation

Media Pipe is a framework for high-fidelity body pose tracking, which takes input from RGB video frames and infers 33 3D landmarks on the whole human. For instructional feedback , the keypoint of the human body was estimated using mediapipe algorithm. It extracts body landmarks of each of the keypoints to enable angle calculation of each body joint. Eg, left/right shoulder, left/right elbow ,left/right wrist , left/right hip , left/right knee and left/right ankle. We can use Openpose instead of mediapipe. OpenPose is an open-source real-time multiple- person detection system, to jointly detect human body, palm, facial, and foot keypoints .But it detects 15, 18, 27 body/foot landmarks. But mediapipe detects 33 keypoints. Working with OpenPose was very slow relative to MediaPipe. It takes a relatively long time to process videos and takes a lot of computing power even on decent machines. So here each yoga pose can be detected and corrected using mediapipe.

Blaze pose model

Mediapipe that uses a model called Blazepose model. BlazePose is a high-fidelity body pose model designed specifically to support challenging domains like yoga, fitness and dance. It can detect 33 keypoints, extending the 17 keypoint topology of the original PoseNet model. These

additional keypoints provide vital information about face, hands, and feet location with scale and rotation. Together with our face and hand models they can be used to unlock various domain-specific applications like gesture control or sign language without special hardware. Blazepose outputs the 33 keypoints according the following ordering convention.



0. nose
1. left_eye_inner
2. left_eye
3. left_eye_outer
4. right_eye_inner
5. right_eye
6. right_eye_outer
7. left_ear
8. right_ear
9. mouth_left
10. mouth_right
11. left_shoulder
12. right_shoulder
13. left_elbow
14. right_elbow
15. left_wrist
16. right_wrist
17. left_pinky
18. right_pinky
19. left_index
20. right_index
21. left_thumb
22. right_thumb
23. left_hip
24. right_hip
25. left_knee
26. right_knee
27. left_ankle
28. right_ankle
29. left_heel
30. right_heel
31. left_foot_index
32. right_foot_index

Figure 3.3: Blaze pose model

The input image is fed to the MediaPipe library for keypoint detection of the user's body. The output is a list of coordinates in the X, Y and Z axis for 33 major key-points of the human body. This list of coordinates define the location of each major body part in the input image. Using these coordinates we can build an accurate skeletal orientation of the user. In Fig. 3.3, the landmarks indicate the major joints and locations on the human body. They are indexed from 0 to 32 to indicate a total of 33 landmarks that are output from the MediaPipe library. The first 11 landmarks from 0 to 10 are used for the facial landmarking procedure. Using these landmarks or key points we can detect the face in an image as well as its orientation. The next 11 landmarks from 11 to 22 are used from the detection of the upper body. Upper body includes the shoulders, elbows, wrists,

hands and an estimate of 3 fingers namely pinky finger, index finger and thumb on both hands. The final 11 key points/landmarks from 23 to 32 are used to define the lower body consisting of the hips, knees, legs, and foot. They together give an estimate of not only the human body structure in the image but also the orientation of the body in 3D space. Pose detection is achieved using the MediaPipe. MediaPipe is an open-source, cross-platform customisable machine learning solution for real-time streaming media such as audio, video and series data. The library is supported on multiple platforms such as Android, iOS, Python, JavaScript.

In contrast to current pose models based on the standard COCO topology. BlazePose accurately localizes more keypoints, making it uniquely suited for fitness applications. we perform evaluation only for 17 keypoints from COCO topology. Blaze pose actually using a very successful deep learning recipe that is creating a 2 step detector, where you combine a computationally expensive object detector with a lightweight object tracker.

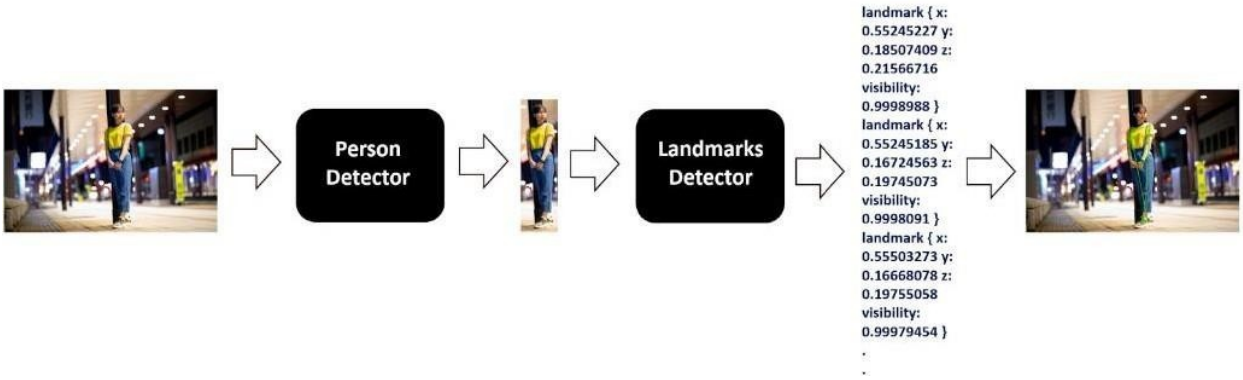


Figure 3.4: Landmark detection working

Run the detector in the first frame of the video to localize the person and provide a bounding box around it, after that the tracker takes over and it predicts the landmark points inside that bounding box ROI, the tracker continues to run on any subsequent frames in the video using the previous frame's ROI and only calls the detection model again when it fails to track the person with high confidence. Their model works best if the person is standing 2-4 meters away from the camera and one major limitation of their model is that this approach only works for single-person pose

detection, it's not applicable for multi-person detection.After performing the pose detection, we will get a list of thirty-three landmarks representing the body joint locations of the prominent person in the image. Each landmark has:

- x - It is the landmark, x-coordinate normalized to [0.0, 1.0] by the image width.
- y - It is the landmark ,y-coordinate normalized to [0.0, 1.0] by the image height.
- z - It is the landmark, z-coordinate normalized to roughly the same scale as x. It represents the landmark depth with midpoint of hips being the origin, so the smaller the value of z, the closer the landmark is to the camera.
- visibility: It is a value with range [0.0, 1.0] representing the possibility of the landmark being visible (not occluded) in the image. This is a useful variable when deciding if you want to show a particular joint because it might be occluded or partially visible in the image.

Figure 3.5: Landmark detection

3.3.2 Pose comparison

The output from the MediaPipe library only contains the coordinates of the user's major key-points in the image. A function is written in the program to get these coordinates data and then calculate the angles at each joints for eg. at elbows, shoulders, hips etc. Given three key-points we can easily calculate the angle made between the two lines using analytic geometry. Let A(x1, y1), B(x2, y2) and C(x3, y3)be the three points. Let the lines AB and BC intersect at B, then the angle between AB and BC can be calculated as: Slope of the line AB is given as,

1) m1=(y2-y1)/(x2-x1)

m1 is the slope between the line joining the point A and B. y1 and x1 are the coordinates of point A and x2 and y2 are the coordinates of point B. This same principle is then again applied on the line BC to get the slope of line BC. Slope of the line BC is given as,

2) m2= (y3-y2)/(x3-x2)

m2 is the slope between the line joining the point B and C. y2 and x2 are the coordinates of the point B and x3 and y3 are the coordinates of point C.The point B is the common point between the three points A, B and C and so the angle is formed at the joint B. This is the angle between AB and BC. Now, the angle between AB and BC can be calculated as,

3)tan $\theta = m1 - m2/1 + m1.m2$

In Eqn. 3, the tan is calculated which can be either positive or negative based on the angle given. By taking the inverse of the tan we get the angle made at B between AB and BC.

AB and BC can be considered as two bones or skeletal structures of the human body. Assuming the line AB as the elbow and line BC as the hand, the angle made between the elbow and hand can be calculated in this manner. On further applying this analysis to all the other joints we can calculate the angles made at each joint. These angles are calculated for all the 6 yoga poses.during a yoga routine the user may not always be in the center of the frame. So the measurements will keep on changing. But if the user is assumed as the center then the angles can be measured by keeping the user as the center. Therefore even if the user is not at the center of the frame we can calculate the angles because the angle calculation is done by keeping the user as the center point.

3.3.3 Pose classification

Create a function that will be capable of classifying different yoga poses using the calculated angles of various joints. The function will be capable of identifying the following yoga poses:

Tree pose

Tree Pose (also known as Vrikshasana) is another yoga pose for which the person has to keep one leg straight and bend the other leg at a required angle. The pose can be classified easily using the following combination of body part angles:

- Around 180° angle at one knee
- Around 35° (if right knee) or 335° (if left knee) angle at the other knee

T pose

T Pose (also known as a bind pose or reference pose), To make this pose, one has to stand up like a tree with both hands wide open as branches. The following body part angles are required to make this one:

- Around 180° at both elbows
- Around 90° angle at both shoulders
- Around 180° angle at both knees

Mountain pose

Though Tadasana (Mountain Pose) might seem simple, it is actually a deeply vital foundational pose for your yoga practice. It is a basic yoga pose where the yogi plants their feet on the ground as their body stands firm and straight.

- Around 180°angle at both knees
- Around 20° angle at both shoulders

Goddess pose

Goddess Pose or Utkata Konasana, is a powerful pose that strengthens the feminine power and helps you connect to the space just beneath and behind your navel that is described as your energetic center. Goddess pose is a squat. Your thighs and entire lower body will turn on and tone. Because the feet are turned out and you sit down, not back, your inner thighs will get a nice stretch and tone. In addition, goddess pose works the core muscles, chest, and back as you hold the posture.

- Around 90°angle at both Shoulder
- Around 90°angle at both Elbow
- Around 100°angle at both Knees

Warrior-I pose

Warrior I stretches your chest, lungs, shoulders, neck, belly and groin. It also strengthens your shoulders, arms, and back muscles, as well as your calves, ankles, and thighs.Keep your upper body facing the front edge of the mat and your shoulders level. Bend your right knee, the right thigh towards parallel to the floor. Keep the weight in the front heel and big toe. Press the back foot down and lift from the inner arch.

Check if one leg is straight

- Knee around 170°

Check if the other leg is bended at the required angle

- Knee around 100°
- Shoulder around 220°
- Elbow around 180°

Warrior-II pose

Warrior II pose, or Virabhadrasana II in Sanskrit, is a standing yoga posture that stretches the shoulder, chest, and groin, and builds strength in the legs, torso, and spine. Warrior II is one of five other warrior poses in a modern yoga practice. It can be classified using the following combination of body part angles:

- Around 180° at both elbows
- Around 90° angle at both shoulders
- Around 180° angle at one knee

3.3.4 Feedback to the user

Giving feedback to the user is of utmost importance so that the user knows what he/she is doing wrong. This helps in guiding the user to correct posture and thus learning to practice the yoga pose correctly. The feedback regarding the performance of the user is provided in real-time via the display messages. When the user deviates beyond the threshold value the user is notified. Users can observe the correction and make necessary adjustments to his/her pose to accurately practice the yoga routine. The feedback can be in the form of a visual alert on the screen.

Each user has varying levels of flexibility that is, one user may not be able to bend or flex his/her body as much as the other user. So in order to tackle that issue, a user changeable threshold parameter is included. Each user can set the threshold as per his/her requirements. A new user can set the threshold, to say 20 degrees so that he/she can have a deviation of about 20 degrees in either direction. An experienced user can set it to less than 10degrees so that he/she can practice the pose accurately. This feature allows even beginner users to slowly and steadily improve his/her body flexibility to do yoga.

In Fig 3.6 the pose requires the hand to be held straight and stretched forming a 180 degrees at the elbow joint. The database contains the data regarding the angle formed corresponding to the pose. So in this case the data stored is 180 degrees. When a user starts practicing this pose, his/her pose angle is compared with this reference pose. The yoga routine starts and the camera starts capturing the pose made by the user frame by frame in real-time. This image is then fed

to the MediaPipe library to get the coordinates of the key points which is then in turn fed to the geometric analysis function which outputs the angle formed at the elbow joint. This angle data is then compared with the reference data to check if the pose is correct.

From Fig 3.7, it is evident that the user has made a pose with an angle of 175 degrees. Assuming the user sets the threshold as 5 degrees, this is acceptable.

In Fig 3.8, The pose made by the user has an angle of 170 degrees which has deviated largely from the threshold angle of 5 degrees set by the user. When such a large deviation occurs, an alert message is displayed on the screen to indicate to the user that the posture is wrong and correction is required. The user can either correct his posture from this feedback or set the threshold to an even larger angle, say 10degrees to make the pose easier to practice.

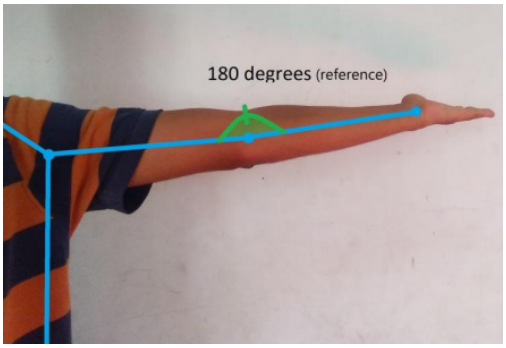


Figure 3.6: Reference Angle Data for Comparison

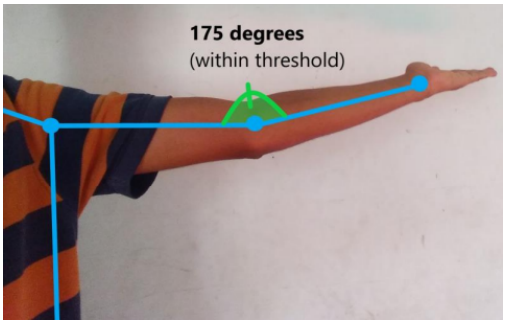


Figure 3.7: User's Hand Pose Of 175 Degrees Is Within The Threshold Of + or -5 Degrees

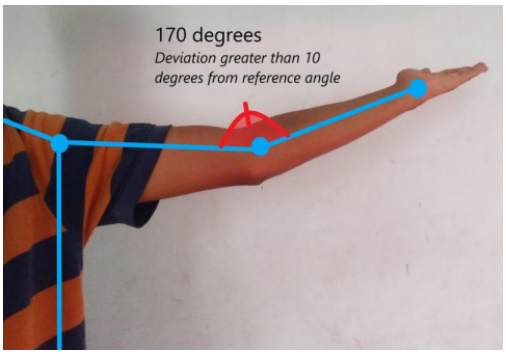


Figure 3.8: User's Hand Pose Of 170 Degrees Is Beyond The Threshold Limit Of 10 Degrees

3.4 Software Requirement and Specification

The tools used for the project are :

• Python

• Jupiter Notebook

• Django

• HTML

• CSS

• Bootstrap

3.4.1 Python

Python is a general purpose, dynamic, high-level, and interpreted programming language. It supports Object Oriented programming approach to develop applications. It is simple and easy to learn and provides lots of high-level data structures. Python was invented by Guido van Rossum in 1991 at CWI in Netherland. Python has wide range of libraries and frameworks widely used in various fields such as machine learning, artificial intelligence, web applications, etc. Python has become a staple in data science, allowing data analysts and other professionals to use the language to conduct complex statistical calculations, create data visualizations, build machine learning algorithms, manipulate and analyze data, and complete other data-related tasks.

Python is often used to develop the back end of a website or application. Python's role in web

development can include sending data to and from servers, processing data and communicating with databases, URL routing, and ensuring security. Python offers several frameworks for web development. Commonly used ones include Django and Flask. It's open source, which means it's free to use and distribute, even for commercial purposes. It has a simple syntax that mimics natural language, so it's easier to read and understand. This makes it quicker to build projects, and faster to improve on them. Python has a large and active community that contributes to Python's pool of modules and libraries, and acts as a helpful resource for other programmer.

3.4.2 Jupyter Notebook

Jupyter Notebook is the latest web-based interactive development environment for notebooks, code, and data. Its flexible interface allows users to configure and arrange workflows in data science, scientific computing, computational journalism, and machine learning. A modular design invites extensions to expand and enrich functionality.

The best method of installing the Jupyter Notebooks is by the installation of the Anaconda package. The Jupyter Notebook and the Jupyter Lab comes pre-installed in the Anaconda package, and you don't have to install this on your own. One of the requirements here is Python, either Python 3.3 or greater or Python 2.7. The general recommendation is that you use the Anaconda distribution to install both Python and the notebook application. The advantage of Anaconda is that you have access to over 720 packages that can easily be installed with Anaconda's conda, a package, dependency, and environment manager. Jupyter supports over 40 programming languages, including Python, R, Julia, and Scala. Notebooks can be shared with others using email, Dropbox, GitHub and the Jupyter Notebook Viewer. Your code can produce rich, interactive output: HTML, images, videos, LaTeX, and custom MIME types.

3.4.3 Django

Django is a web application framework written in Python programming language. It is based on MVT (Model View Template) design pattern. The Django is very demanding due to its rapid development feature. It takes less time to build application after collecting client requirement. By using Django, we can build web applications in very less time. Django is designed in such a manner

that it handles much of configure things automatically, so we can focus on application development only. Django was design and developed by Lawrence journalworld in 2003 and publicly released under BSD license in July 2005. Django takes security seriously and helps developers to avoid many common security mistakes, such as SQL injection, cross-site scripting, cross-site request forgery etc. Its user authentication system provides a secure way to manage user accounts and passwords. Django is scalable in nature and has ability to quickly and flexibly switch from small to large scale application project.

Django includes various helping task modules and libraries which can be used to handle common Web development tasks. Django takes care of user authentication, content administration, site maps, RSS feeds etc. Django is versatile in nature which allows it to build applications for different-different domains. Now a days, Companies are using Django to build various types of applications like: content management systems, social networks sites or scientific computing platforms etc. Django is an open source web application framework. It is publicly available without cost. It can be downloaded with source code from the public repository. Open source reduces the total cost of the application development. It is an one of the most popular web framework.

3.4.4 HTML

HTML is an acronym which stands for Hyper Text Markup Language which is used for creating web pages and web applications. It is very easy to make an effective presentation with HTML because it has a lot of formatting tags. It is a markup language, so it provides a flexible way to design web pages along with the text.

It facilitates programmers to add a link on the web pages (by html anchor tag), so it enhances the interest of browsing of the user. HTML elements are the fundamental constituents of HTML pages. Images and other objects, such as interactive forms, can be embedded in the rendered page using HTML constructs. HTML enables the creation of structured documents by assigning structural semantics to text elements such as headings, paragraphs, lists, links, and other elements. It is platform-independent because it can be displayed on any platform like Windows, Linux, and Macintosh, etc. HTML is a case-insensitive language, which means we can use tags either in lower-case or upper-case. It facilitates the programmer to add Graphics, Videos, and Sound to the

web pages which makes it more attractive and interactive.

3.4.5 CSS

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file. CSS provides more detailed attributes than plain HTML to define the look and feel of the website. It saves a lot of time. It controls the layout of multiple web pages at one time. It sets the font-size, font-family, color, background color on the page. It allows us to add effects or animations to the website. We use CSS to display animations like buttons, effects, loaders or spinners, and also animated backgrounds.

3.4.6 Bootstrap

Bootstrap is an HTML, CSS JS Library that focuses on simplifying the development of informative web pages.The primary purpose of adding it to a web project is to apply Bootstrap's choices of color, size, font and layout to that project. As such, the primary factor is whether the developers in charge find those choices to their liking. Once added to a project, Bootstrap provides basic style definitions for all HTML elements. The result is a uniform appearance for prose, tables and form elements across web browsers. In addition, developers can take advantage of CSS classes defined in Bootstrap to further customize the appearance of their contents. For example, Bootstrap has provisioned for light- and dark-colored tables, page headings, more prominent quotes, and text with a highlight.

Bootstrap also comes with several JavaScript components which do not require other libraries like jQuery. They provide additional user interface elements such as dialog boxes, tooltips, progress bars, navigation drop-downs, and carousels. Each Bootstrap component consists of an HTML structure, CSS declarations, and in some cases accompanying JavaScript code. They also extend the functionality of some existing interface elements, including for example an auto-complete function for input fields.

Chapter 4

RESULTS AND DISCUSSIONS

A user can choose the specific yoga posture from different postures. The yoga self-coaching system recognize the yoga postures and to output the predicted result and give real-time guidance for incorrect postures. The resulting recognized posture of the user will pop up with a success message of "Correct :)" when the user achieves in the right posture and illustrated the fail message of "Incorrect :("when the user has not achieved the correct posture. Furthermore, the system will provide instructional feedback to a user, to provide guidance enabling them to perform the posture correctly, based on the angle of each joint, calculated from keypoints extracted when a user performs the incorrect yoga posture.

Giving feedback to the user is of utmost importance so that the user knows what he/she is doing wrong. This helps in guiding the user to correct posture and thus learning to practice the yoga pose correctly. The feedback regarding the performance of the user is provided in real-time via the display messages. When the user deviates beyond the threshold value the user is notified. Users can observe the correction and make necessary adjustments to his/her pose to accurately practice the yoga routine. The feedback can be in the form of a visual alert on the screen.

4.1 Graphical User Interface

User interface is important to meet user expectations and support the effective functionality of your site. A well-executed user interface facilitates effective interaction between the user and the program, app or machine through contrasting visuals, clean design and responsiveness.

The front end of the project is created using Bootstrap.This includes a page with different yoga poses,login and registration form for user.User can select any yoga pose from 6 poses for practice.Django is used as back end.

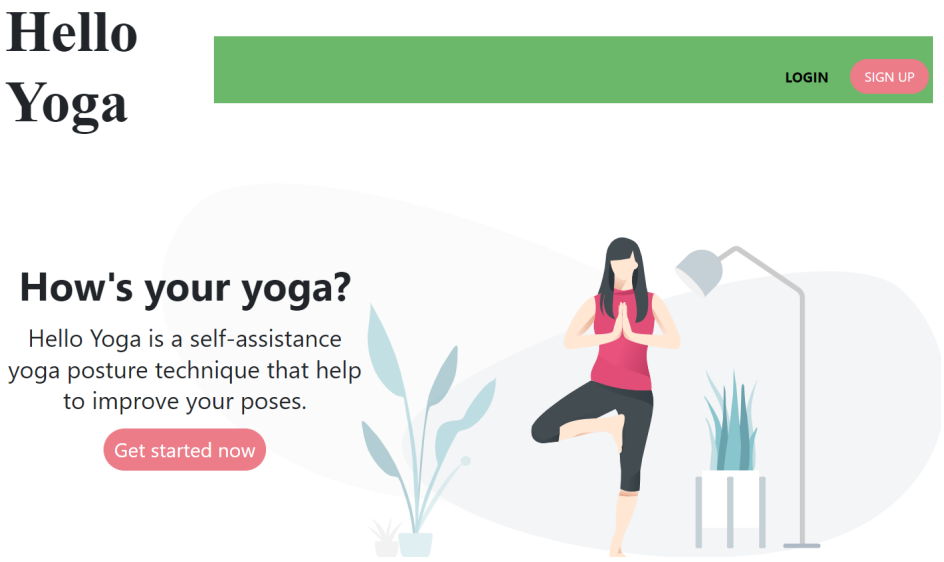


Figure 4.1: Yoga homepage

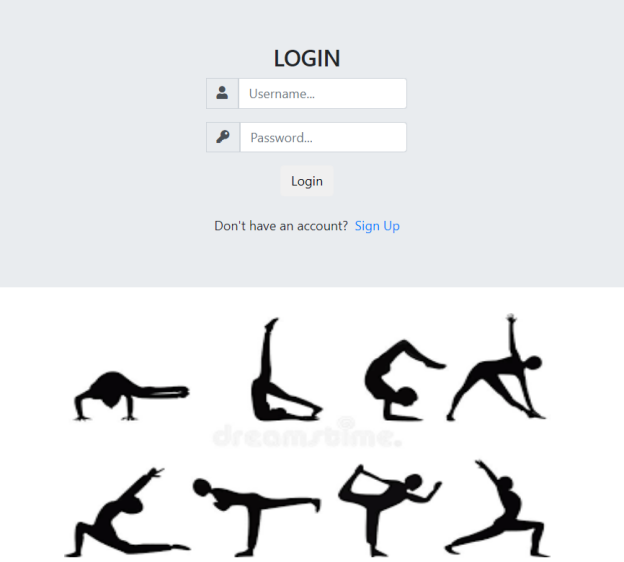


Figure 4.2: User login

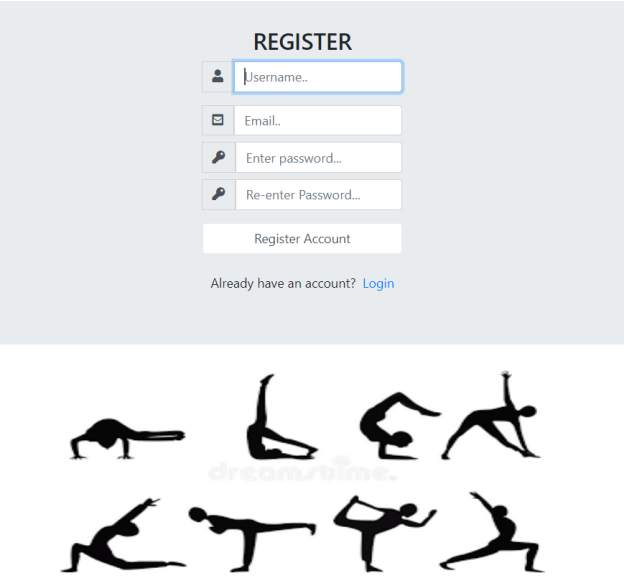


Figure 4.3: User registration

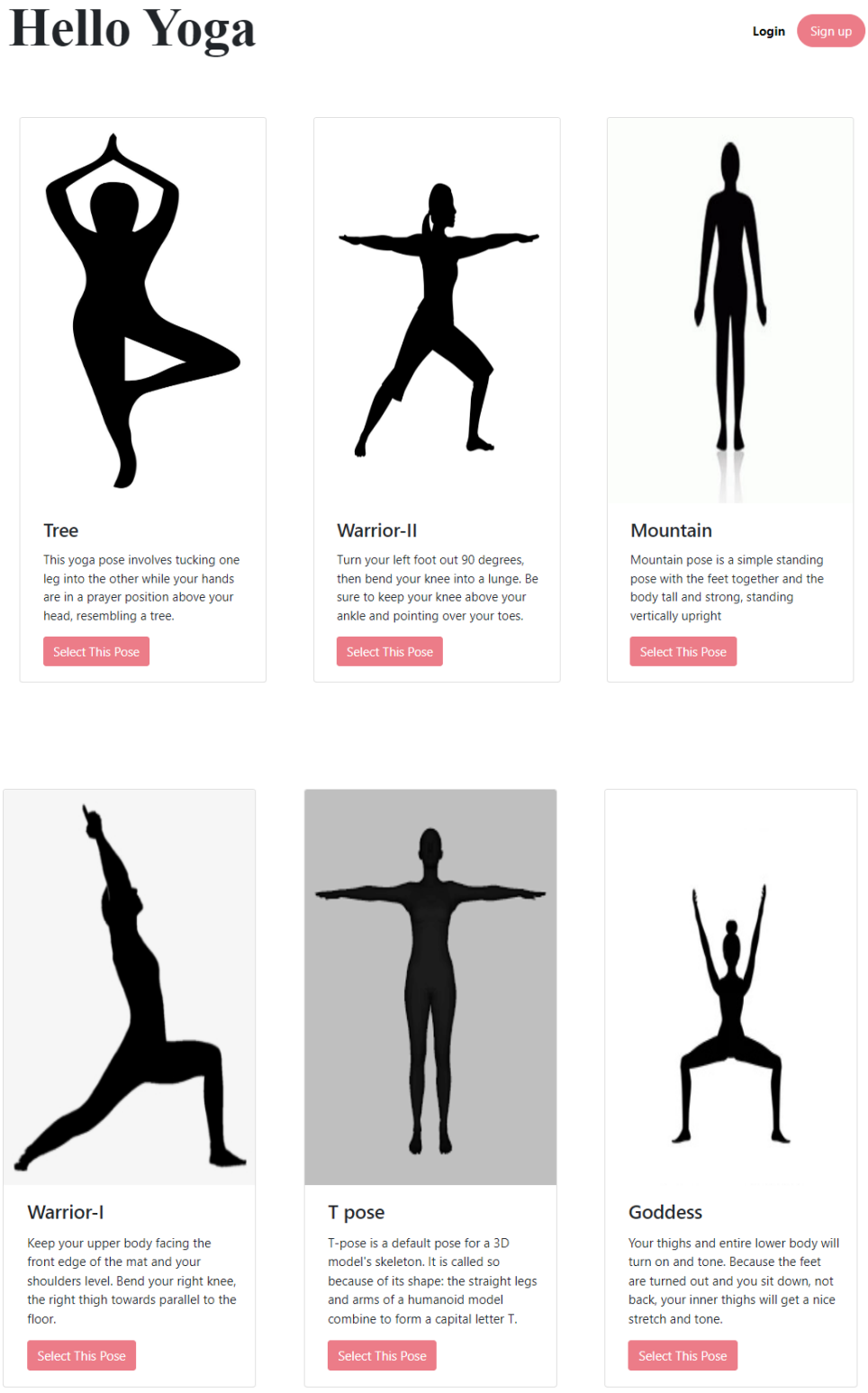


Figure 4.4: Different yoga poses

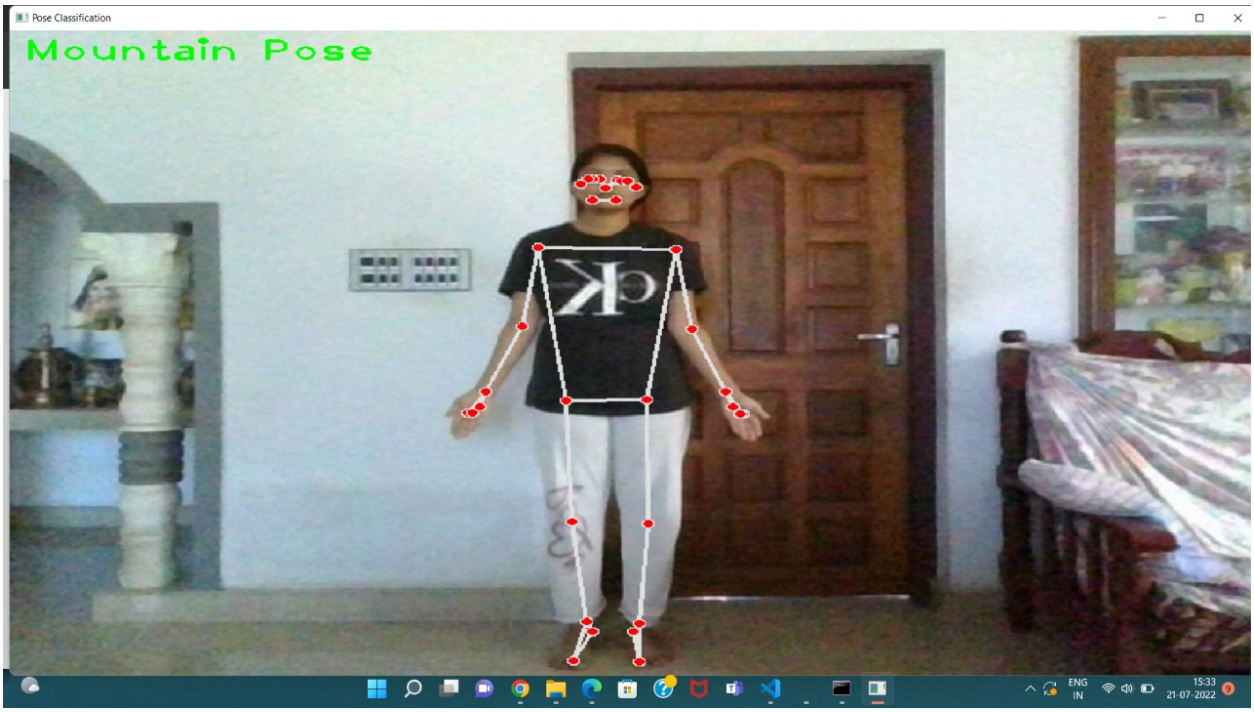


Figure 4.5: Mountain pose

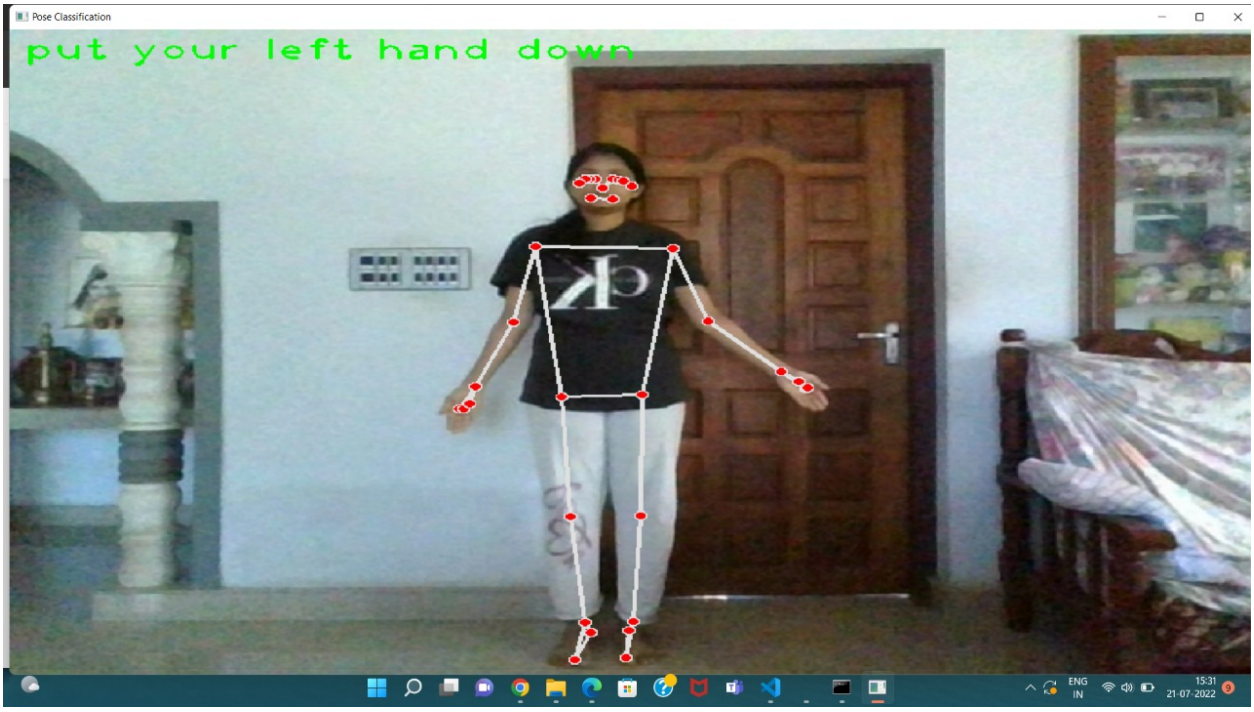


Figure 4.6: Mountain pose correction

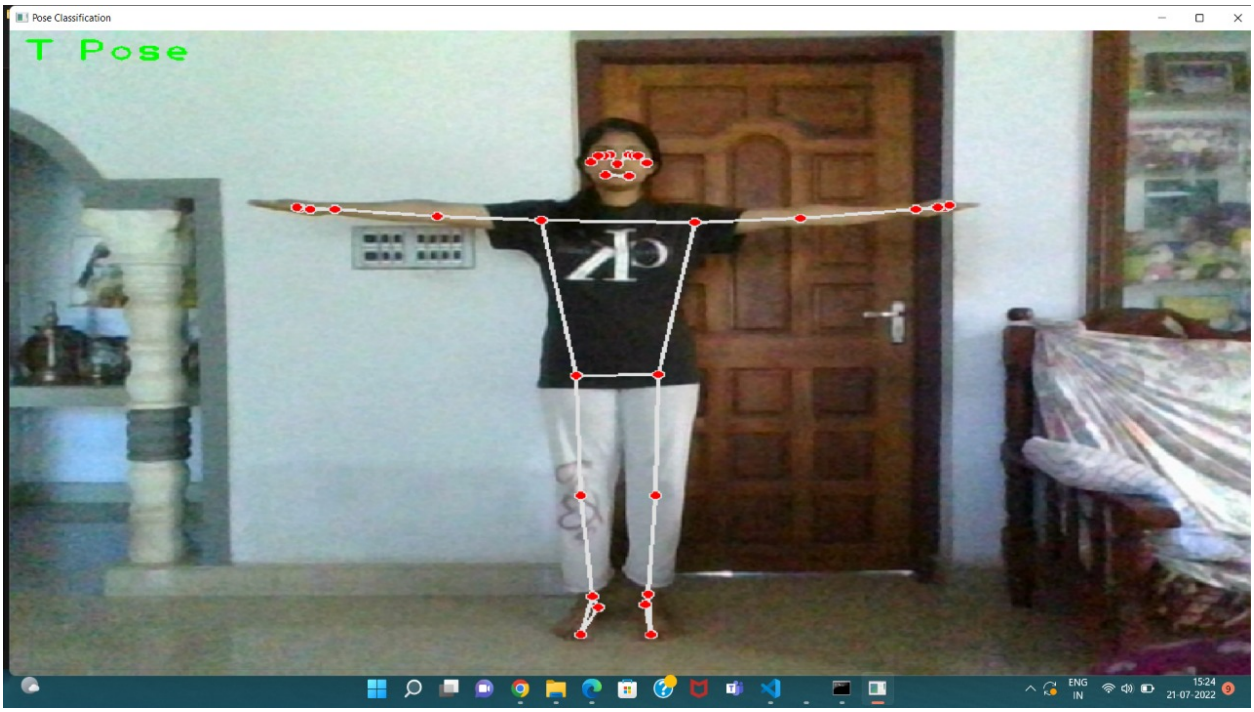


Figure 4.7: T pose

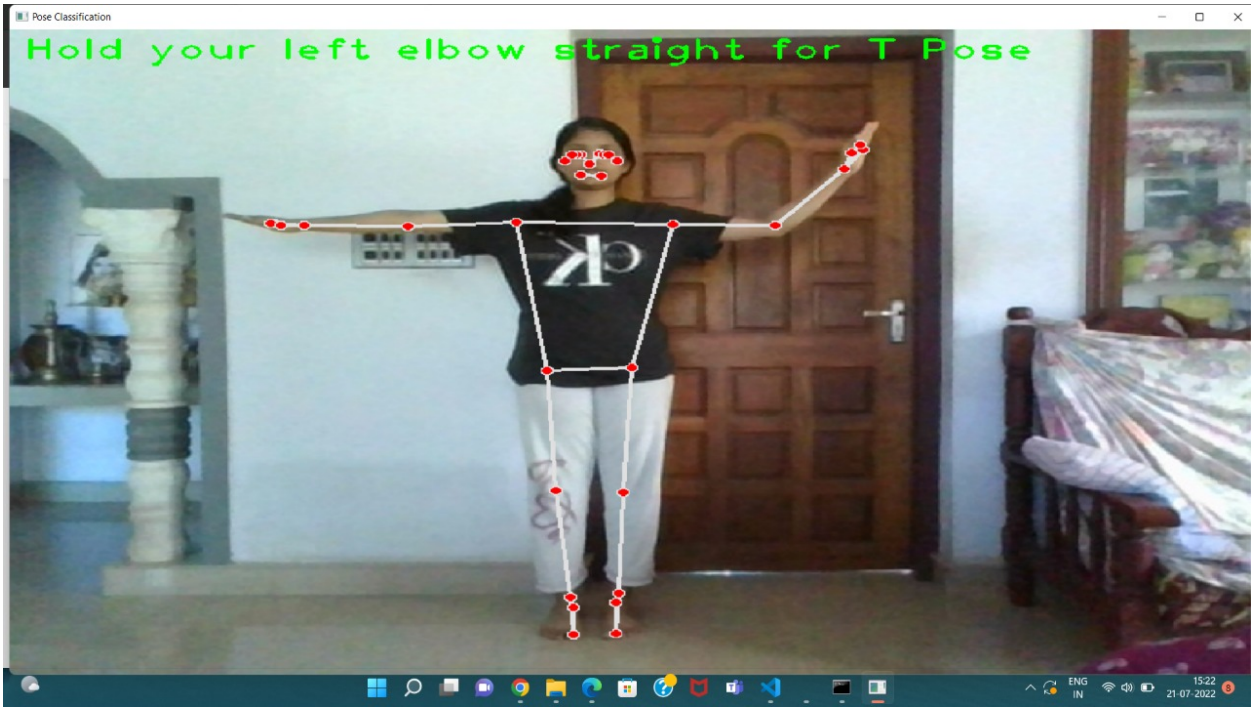


Figure 4.8: T pose correction

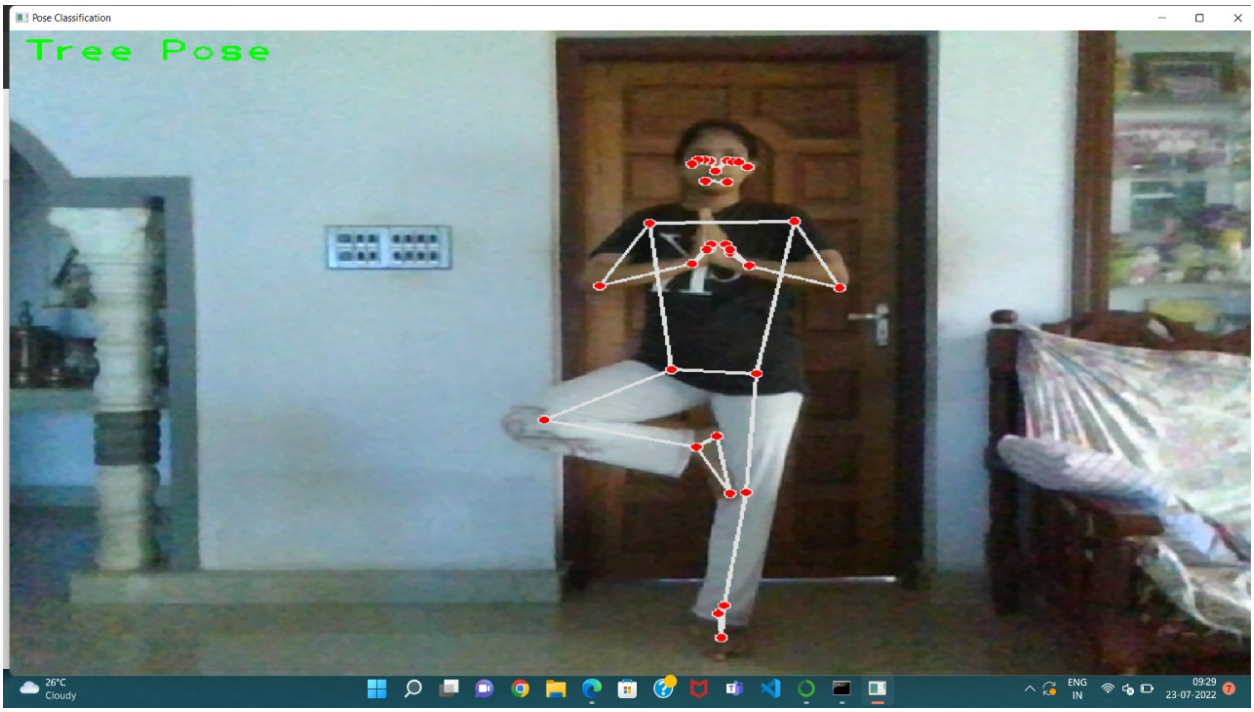


Figure 4.9: Tree pose

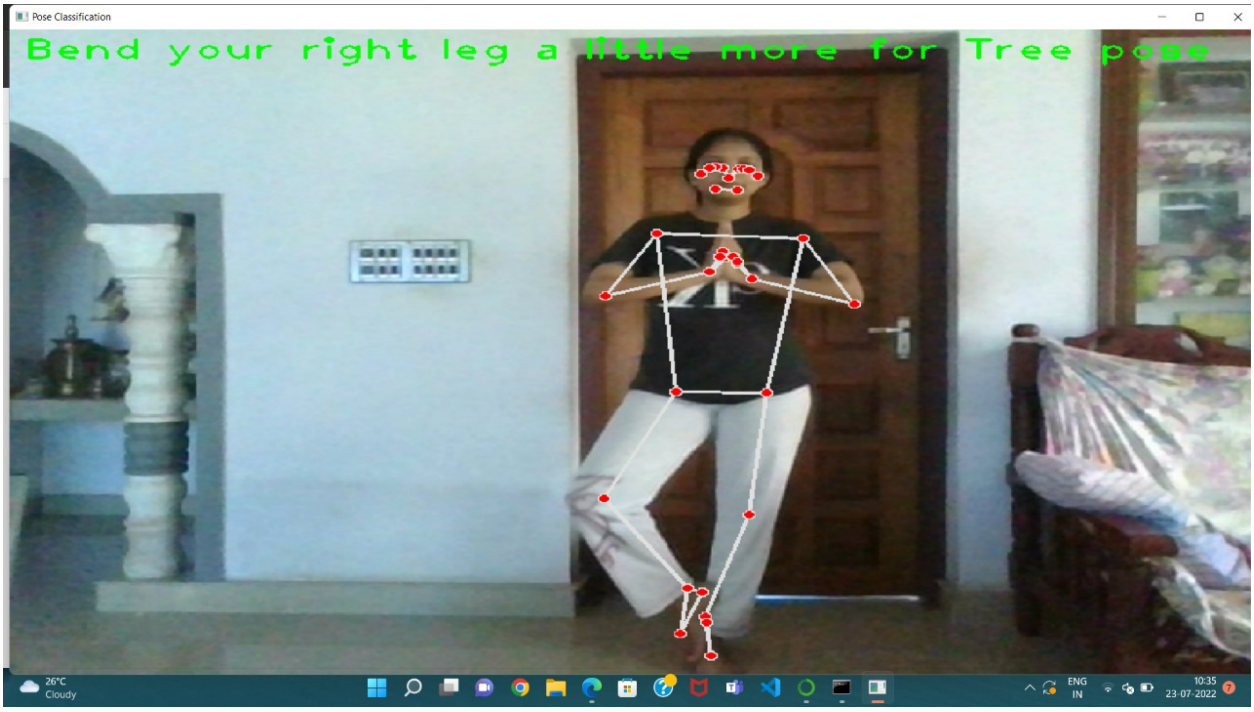


Figure 4.10: Tree pose

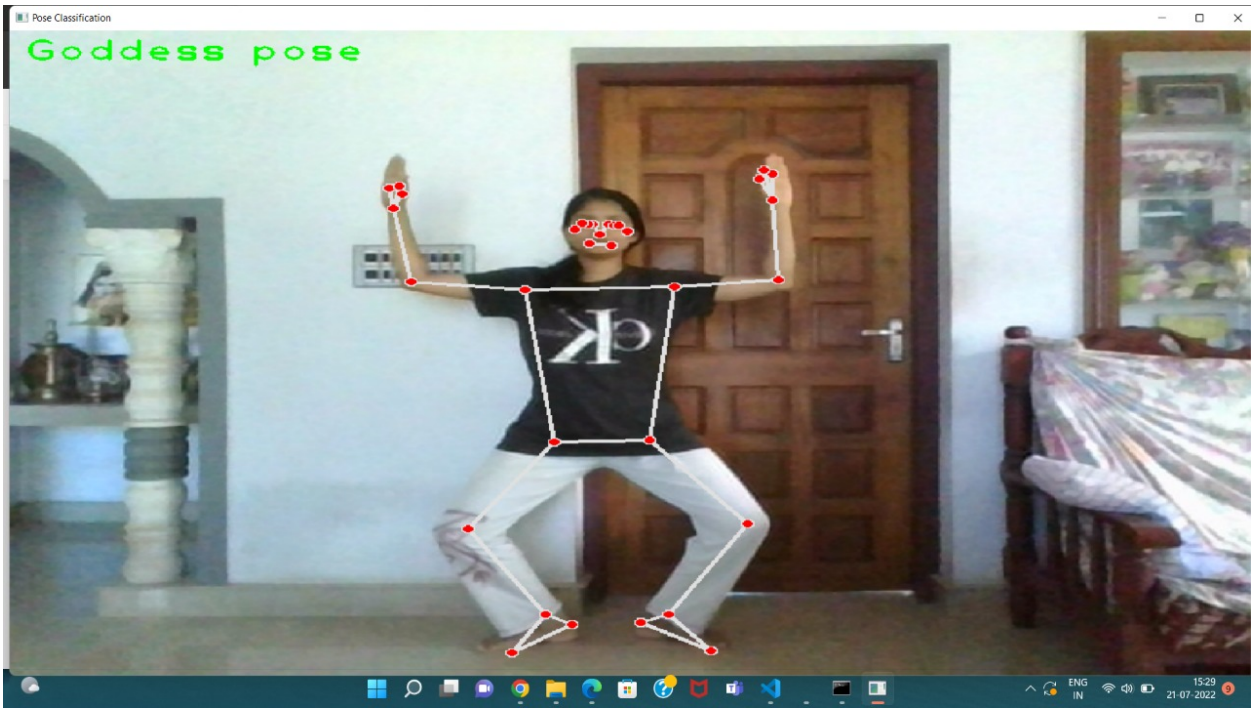


Figure 4.11: Goddess pose

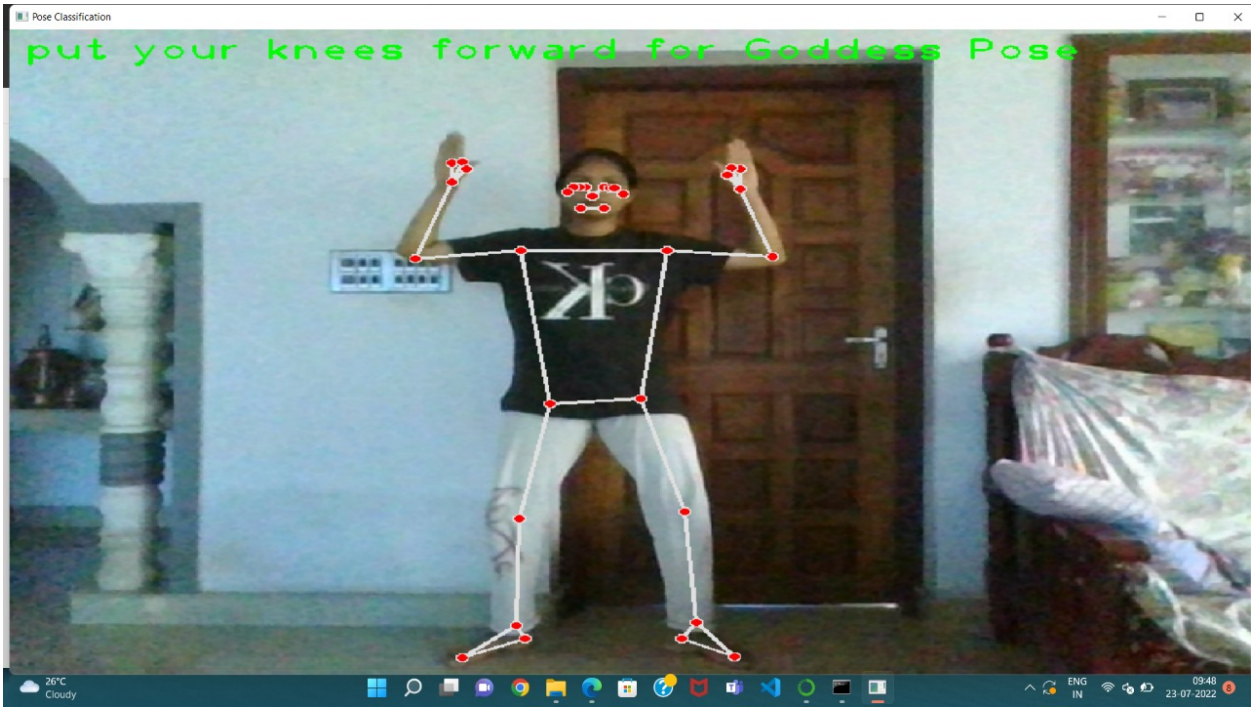


Figure 4.12: Goddess pose correction

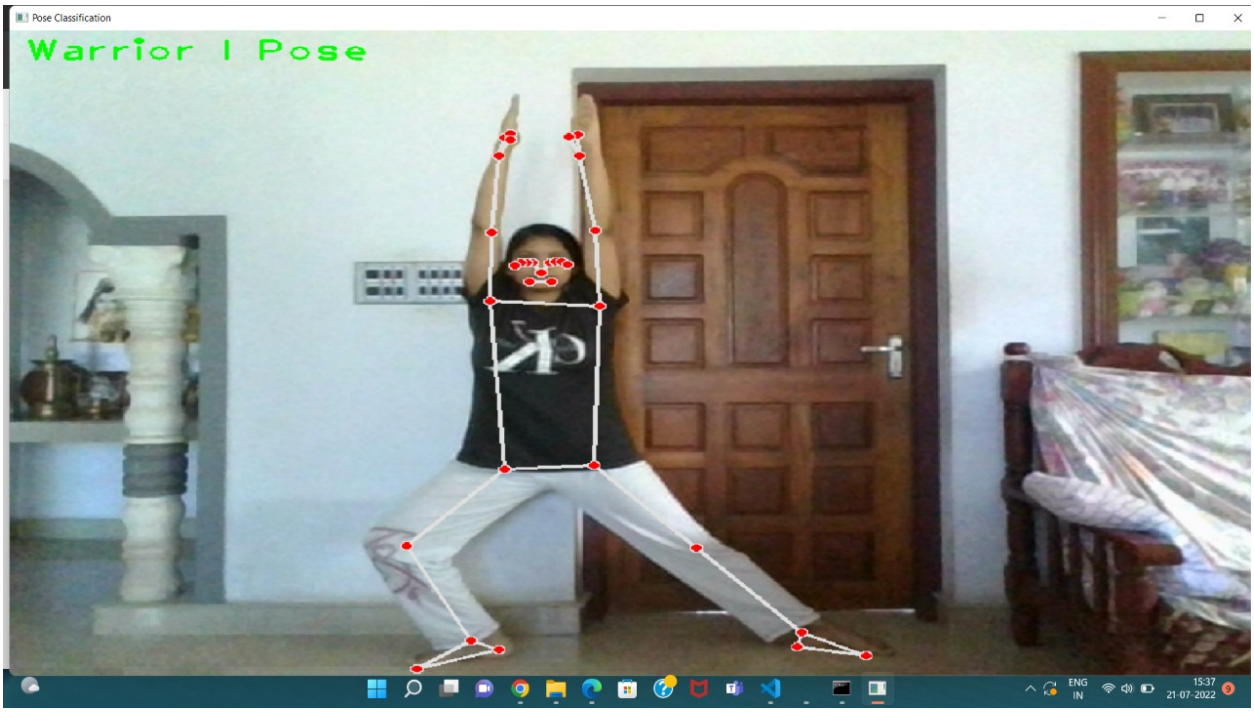


Figure 4.13: Warrior-I pose

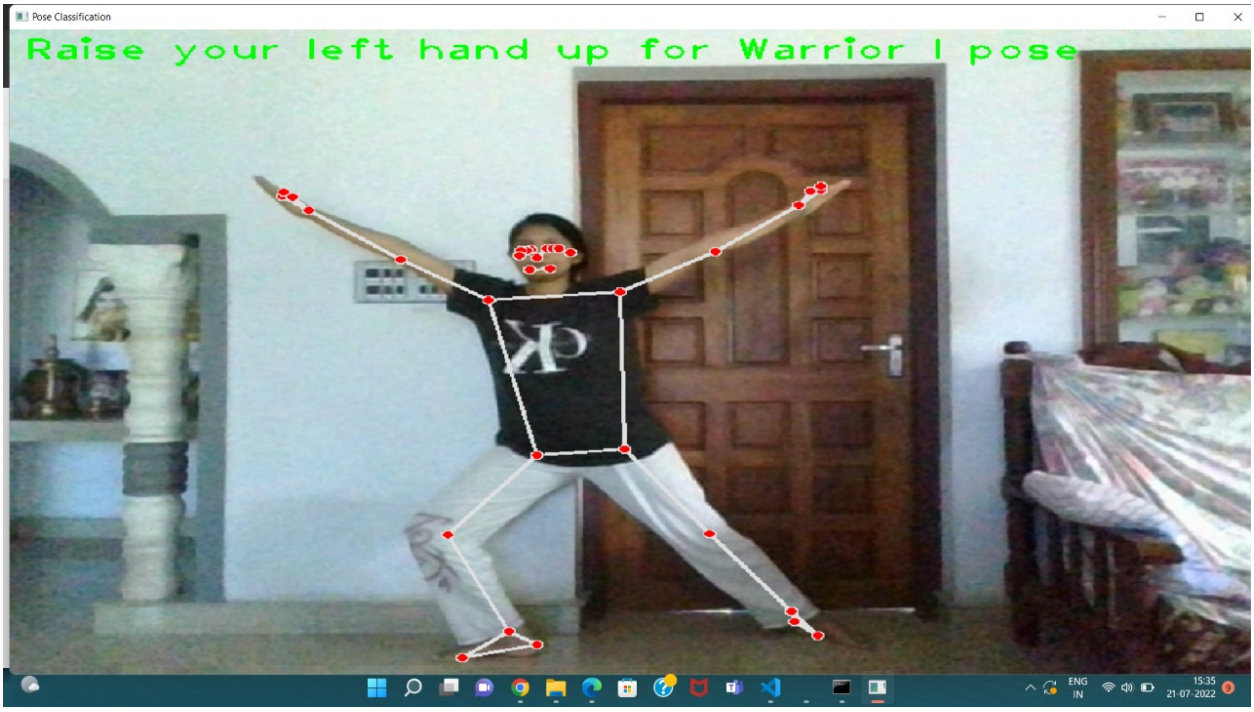


Figure 4.14: Warrior-I pose correction

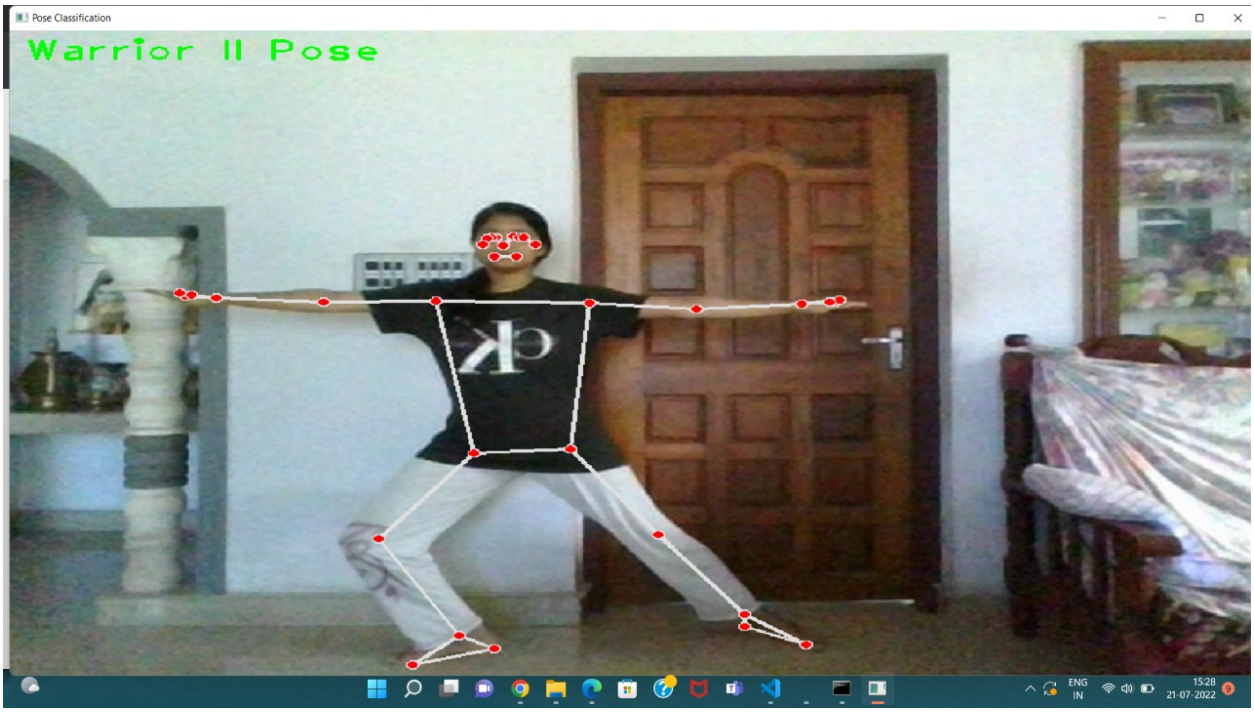


Figure 4.15: Warrior-II pose

Figure 4.16: Warrior-II pose correction

Chapter 5

CONCLUSION

Human poses are seemingly complex, still emerging applications such as yoga posture detection have made the implementation of the technology imperative. An artificial intelligence-based resource that allows people to do yoga and other activities properly can helps to improve the popularity and benefits of these practices. The use of yoga posture estimation in fitness can help people to avoid injuries and increase their effectiveness. Here different yoga postures were collected from camera, and learners were required to perform each yoga posture. It can evaluate pose quality, Which can distinguish different movements by analyzing pose characteristics. Therefore, it automatically evaluating yoga poses and providing suggestions to alert learners using mediapipe algorithm.

In this project, the mediapipe algorithm is used for extracting different coordinates. By using the coordinates, different angles of joints, distance and slops between two-point have been extracted and used as an evaluation parameter for the correction of the yoga poses. It increasing demand for the development of computer-assisted training systems from the practitioner side to help them to improve their knowledge and understanding about different types of yoga poses and also protect them from injury that might occur during learning phases. The method proposed in this work can help the development of applications and other resources for making yoga and its benefits accessible to everyone.

5.1 Future Enhancement

Currently, the proposed model classifies yoga asanas into 6 categories. As there are so many yoga asanas, developing a posture estimate model that works for all of them is a tedious task. Then it can be expanded by adding required yoga pose key points.The technology may also used to make real-time predictions and self learning on a mobile device.The display of the model is determined by the nature of Mediapipe's current evaluation, which may or may not work well in situations when there is overlap between persons or across body parts.There are several instances of real-life applications in which a single individual posture evaluation will not be enough. To include many poses and to get model works on many poses is challenging enough.

In summary, I developed a yoga self-coaching system that can predict yoga posture and confirm instruction feedback in real time.I hope to include more AI-assisted complex yoga asanas in the future.Since the start of Covid-19, home training has increased and our developed system supports this in our opinion.The yoga self-coaching system is used to recognize the correct yoga posture and provide instruction in real time.

References

[1] F. Rishan, B. De Silva, S. Alawathugoda, S. Nijabdeen, L. Rupasinghe and C. Liyanapathirana, "Infinity Yoga Tutor: Yoga Posture Detection and Correction System," 2020 5th International Conference on Information Technology Research (ICITR), 2020, pp. 1-6, doi: 10.1109/ICITR51448.2020.9310832.

[2] Agrawal, Y., Shah, Y., Sharma, A. "Implementation of machine learning technique for identification of yoga poses". In 2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT),2020, 40–43. doi:10.1109/CSNT48778.2020.9115758

[3] Anilkumar, A., K.t., A., Sajan, S., K.a., S, "Pose Estimated Yoga Monitoring System", SSRN Electronic Journal. doi:10.2139/ssrn.3882498 Zhang Z (2012) Microsoft kinect sensor and its effect, IEEE Multi 19:4–10. https://doi.org/10.1109/ MMUL.2012.24

[4] Chun J, Park S, Ji M ," 3D human pose estimation from RGB-D images using deep learning method". In: Proceedings of the 2018 International Conference on Sensor, Signal and Image Processing, 2018,pp 51–55, https://doi.org/10.1145/3290589.3290591

[5] Kadbhane,S., Datir,K., Jagdale,T., Dhongade,S., Jagtap,G."Yoga Posture Recognition", International Journal of Advanced Research in Computer and Communication Engineering, 2021,143–147. doi:10.17148/IJARCCE.2021.10128

[6] Girija Gireesh Chiddarwar, Abhishek Ranjane, Mugdha Chindhe ," AI-Based Yoga Pose Estimation for Android Application", International Journal of Innovative Science and Research Technology,2020.

[7] Charles J, Pfster T, Magee D, Hogg D, Zisserman A ,"Personalizing human video pose estimation",2016,arXiv arXiv:1511.06676

[8] Verma M, Kumawat S, Nakashima Y, Raman S, "Yoga-82: a new dataset for fine- grained classification of human poses", IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), arXiv arXiv:2004.10362

[9] M. U. Islam, H. Mahmud, F. B. Ashraf, I. Hossain and M. K. Hasan, "Yoga posture recognition by detecting human joint points in real time using microsoft kinect," 2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC), 2017, pp. 668-673, doi: 10.1109/R10-HTC.2017.8289047.

[10] Yadav, Santosh Singh, Amitojdeep Gupta, Abhishek Raheja, Jagdish, "Real-time Yoga recognition using deep learning. Neural Computing and Applications",Springer,2019,31.https://link.springer.com/article/10.1007/s00521-019. 10.1007/s00521-019-04232-7.

[11] S. Liaqat, K. Dashtipour, K. Arshad, K. Assaleh and N. Ramzan, "A Hybrid Posture Detection Framework: Integrating Machine Learning and Deep Neural Networks," in IEEE Sensors Journal, vol. 21, no. 7, pp. 9515-9522, 1 April1, 2021, doi: 10.1109/JSEN.2021.3055898.

[12] A. Sharma, Y. Agrawal, Y. Shah and P. Jain, "iYogacare: Real-Time Yoga Recognition and Self-Correction for Smart Healthcare," in IEEE Consumer Electronics Magazine, 2022,doi: 10.1109/MCE.2022.3171054.

[13] Alzahrani, M., Kammoun, S. Human Activity Recognition: Challenges and Process Stages.International Journal of Innovative Research in Computer and Communication Engineering,2016, 4(5),1111-1118.

[14] 18. A. Gupta and H. P. Gupta, "YogaHelp: Leveraging Motion Sensors for Learning Correct Execution of Yoga With Feedback," in IEEE Transactions on Artificial Intelligence, vol. 2, no. 4, pp. 362-371, Aug. 2021, doi: 10.1109/TAI.2021.3096175.

APPENDIX

A Screenshots

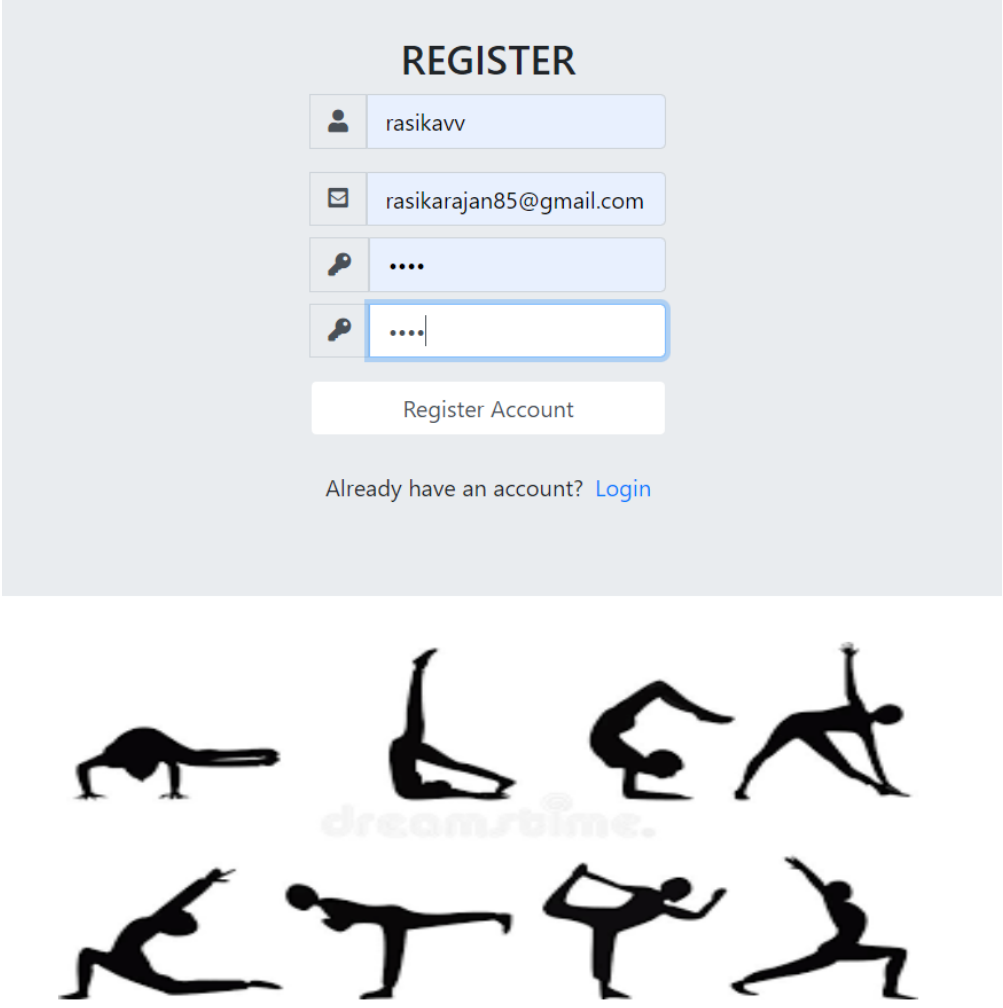


Figure A.1 : User registration

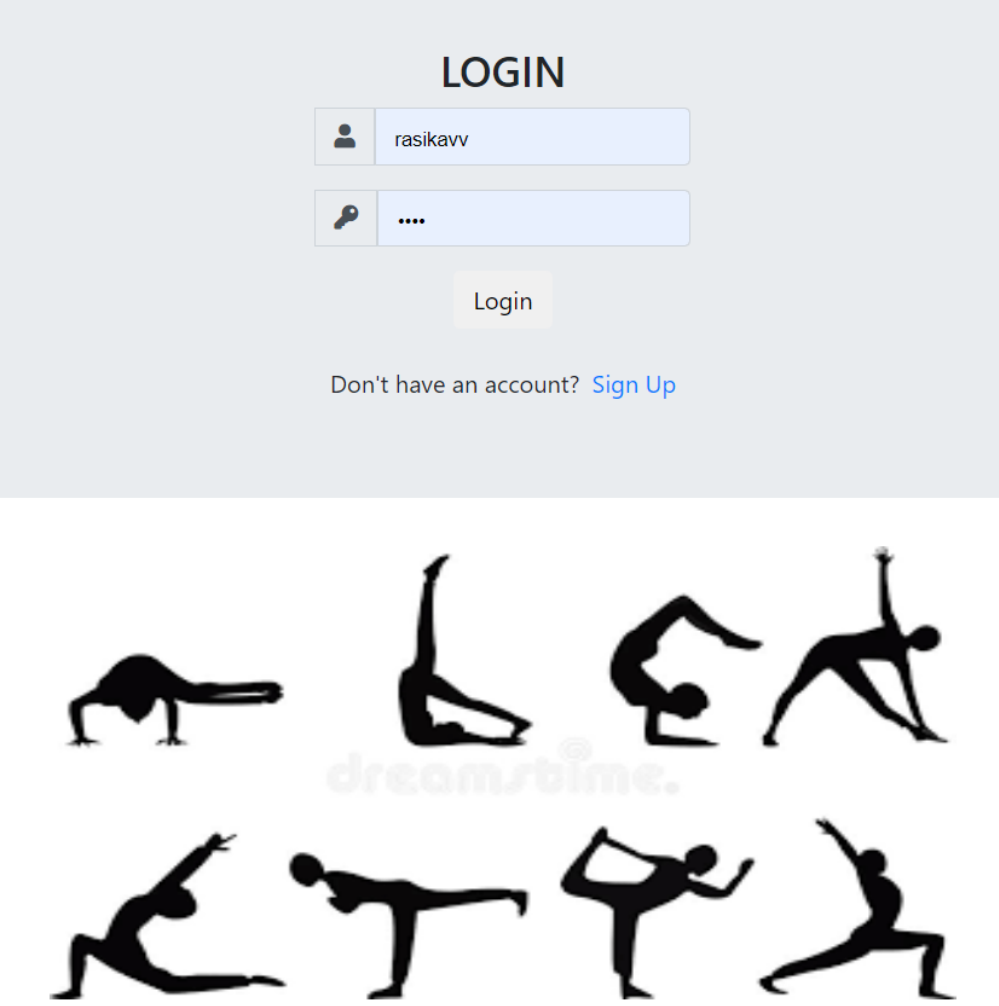


Figure A.2 : User login