HOME WORKOUT WITH VOICE ASSISTANT

MINOR PROJECT II

SUBMITTED BY:

JYOTI BAJAJ (19103145)

Under the supervision of:

Dr. INDU CHAWLA

Department of CSE/IT

Jaypee Institute of Information Technology University, Noida

MAY 2022

ACKNOWLEDGEMENT

We would like to place on record our deep sense of gratitude to Dr. INDU CHAWLA, ASSISTANT PROFESSOR (SENIOR GRADE), Jaypee Institute of Information Technology, India for her generous guidance, help and useful suggestions.

We express our sincere gratitude to Dr. NEETU SARDANA and Dr. PRANTIK BISWAS for their stimulating guidance, continuous encouragement and supervision throughout the course of present work.

DECLARATION

We hereby declare that this submission is our own work and that, to the best of our knowledge and beliefs, it contains no material previously published or written by another person nor material which has been accepted for the award of any other degree or diploma from a university or other institute of higher learning, except where due acknowledgment has been made in the text.

Student ID	Student Name	Student signature
19103145	Jyoti Bajaj	Jyoti
19104024	Ritik Agrawal	Ritik Agrawal
19104052	Shambhavi Agrawal	Shambhavi Agrawal

CERTIFICATE

I, Indu Chawla, declare that above submitted project with Titled HOME WORKOUT WITH HUMAN VOICE ASSISTANT was conducted in my supervision. The project is original and neither the project was copied from External sources not it was submitted earlier in JIIT. I authenticate this project.

Supervisor

Dr. Indu Chawla

ABSTRACT

Exercise is an important factor to maintain a healthy life. One does not need Expensive gyms equipments or memberships to maintain their health. Rather they can still achieve it by being consistent in their home workout. The goal of our tool is to help people perform their daily workouts. The task performed here is counting the number of repetitions of exercise the user is performing live. It analysis the accuracy of the body posture of the user for the exercise that the user is performing. If the posture is correct, the tool will return the count of repetitions and calories burnt by the user will be calculated. At last it summarises the workout routine of the user.

TABLE OF CONTENTS

1. INTRODUCTION

Health and Fitness are very well known terms that has a major effect on standard of living in everyone’s life. A Healthy Body implies Healthy mind which is much needed for proper functioning of body. Though, Health should be valued more than wealth but unfortunately due to metro lifestyle of people we don’t often see that happening. Everyone in today’s era is so busy in running behind materialistic things that they often forget to value their own health. That is one the main reason health quality is lacking in our daily lifestyle. But Since the beginning of COVID Pandemic when humans actually felt the fear of losing their near and dear ones, it made them realize the importance of a healthy lifestyle.

So, when people started giving equal importance to health in their life, they made necessarily changes to improve their health quality. Some of them are less processed food, more nutritious food, daily walking, home workouts etc. During this tough times when everything was shut down including gym people had no choice but to start working out at home. However, for the beginner with no previous knowledge of exercises it was difficult for them to analyse if their actions worked correct or not. So here we are introducing you to a tool that will help people to perform their daily home workout without the help of trainer or assistant. This tool provides them a virtual voice trainer that helps them in counting the repetitions of exercise and analyse their body posture.

To build this idea into a tool we have used pretrained model of mediapipe. For the better convenience of the user we have added voice assistant to our tool. For the voice assistant tool we have used speech recognition and pyaudio library. This helps in converting user audio to text and text to audio. With the help of voice assistant user need to speak out the exercise name and repetitions he /she wants to perform. After successful voice detection of the user input the voice assistant will then forward the user to the virtual trainer tool.

The virtual trainer reads and capture the current user feed with the help of webcam and visualize the body pose of the user. With the help of mediapipe components the tool will detect and draw all the landmarks on the user body and then we will use the required landmarks for the particular exercise to count the repetitions of the action and the correctness of the body posture.

Then further when the number of repetitions desired by the user is achieved, the tool will end the session and finally summarises the session performed by the user. If the user wishes to continue the session for other exercises then he/she can again command the voice assistant to start the whole process again.

2. BACKGROUND STUDY

2.1 Sensor based[15]

Sensor based methods rely on data from inertial measurement units (IMUs). The most common ones are accelerometers and gyroscopes. These sensors generate multivariate time-series data, which is typically segregated using the sliding-window technique before sending for feature extraction and classification.

Traditional pattern recognition techniques like Hidden Markov Models (HMMs), decision trees, and Support Vector Machines (SVMs) have already shown strong results on a wide range of activity recognition tasks, but their reliance on hand - crafted features and potential to only learn shallow representations limit their effectiveness and generalisation.

2.2 Video based[14]

The research area of skeletal-based HAR has emerged with the release of the Microsoft Kinect depth sensor and body tracking SDK in 2010, and the more recent development of techniques for real-time and accurate RGB-based human pose estimation. Skeletal sequence data consists of body joint trajectories, and can, similarly to inertial sensor data, be viewed as multivariate time-series data.

The approach focuses on using different deep learning techniques to identify the joints. In particular, recurrent neural networks (RNNs) and convolutional neural network (CNNs) have been extensively used. Although they are not able to identify points with 100 percent accuracy but it’s much better than other techniques. Inspired by the success we'll be using the same technique to build our model.

2.3 Wearable Devices[10]

Wearables are used to track users' exercise motions and visually compare them to optimal posture and movements for guidance. Guidance focuses solely on tracking user motions, but we also want to identify exercises. MyHealthAssistant, for instance, used a subject-specific Bayesian classifier to accurately classify 13 workouts and count repetitions with 92 percent accuracy. Such systems often use noise-free exercise datasets to demonstrate initial viability. but they usually fail to accurately segment exercises from other activities for real-world data, which is often noisy.

2.4 Multimodal Deep Learning[7]

Fitness tracking devices have risen in popularity in recent years, but limitations in terms of their accuracy and failure to track many common exercises presents a need for improved fitness tracking solutions. This work proposes a multimodal deep learning approach to leverage multiple data sources for robust and accurate activity segmentation, exercise recognition and repetition counting.

We use deep learning methods ,due to their demonstrated ability to learn generalized hierarchical representations, which are important for dealing with the high intra-class variability found in HAR datasets. In addition, deep learning methods can be trained end-to-end directly on the raw data, avoiding the need for designing handcrafted features of shallower learning methods.

2.5 Vision Based[8]

Capturing and tracking a regimen is challenging. Manual tracking is most accurate, but this is tedious for end users. Thus, numerous commercial and academic efforts have focused on automatically tracking and quantifying physical activity, the most pervasive being step count captured by a worn device.

We present a vision-based system that uses off-the-shelf cameras to automate exercise tracking and provide high-fidelity analytics, such as repetition count, without any user- or environment specific training or intervention. Instead of requiring each user in the gym to wear a sensor on their body, GymCam is an external single-point sensing solution, i.e., a single camera placed in a gym can track all people and exercises simultaneously.

2.6 Using Machine Learning & Computer Vision[13]

We propose a system that can be used with any camera based device capable of streaming the user workout to analyze and detect errors. The system is robust enough to handle out of sync user and reference videos, along with any camera positioning artifacts .Our system uses techniques from Deep Learning and Computer Vision to tackle the problem. We start with a trainer recording his/her workout video which may be a single rep or a set of reps.

A trainer records the workout which is uploaded to a cloud service and body part co-ordinates are extracted using Deep Learning CNN model. Trainer's video with extracted metadata is sent to the user device. Next a user performs and records a workout using some device (mobile or TV). The body parts are extracted for every kth user frame on the server. For intermediate frames, body-parts are calculated on user device using optical flow tracking.

3. REQUIREMENT ANALYSIS

➢ Language Used

PYTHON

➢ Libraries Used

1. **Speech Recognition** - Speech recognition is a machine's ability to listen to spoken words and identify them. We have used speech recognition in Python to convert the spoken words into text, make a query or give a reply. We have even used it to respond to these spoken words.

2. **Python Pyttsx3** - Pyttsx3 is a text-to-speech conversion library in Python.

3. **PyAudio** – PyAudio provides Python bindings for Port Audio, the cross-platform audio I/O library. With PyAudio, one can easily use Python **to play and record audio on a variety of platforms**.

4. **Numpy** - Numpy is used to work with arrays. It also has functions for working in Linear Algebra.

5. **Medipipe** - MediaPipe is **a Framework for building machine learning pipelines for processing time-series data like video, audio, etc**

6. **CV2** - OpenCV is a Python open-source library, which is used for **computer vision in Artificial intelligence, Machine Learning, face recognition, etc**.

7. **DateTime** - Python Datetime module **supplies classes to work with date and time**. These classes provide a number of functions to deal with dates, times and time intervals.

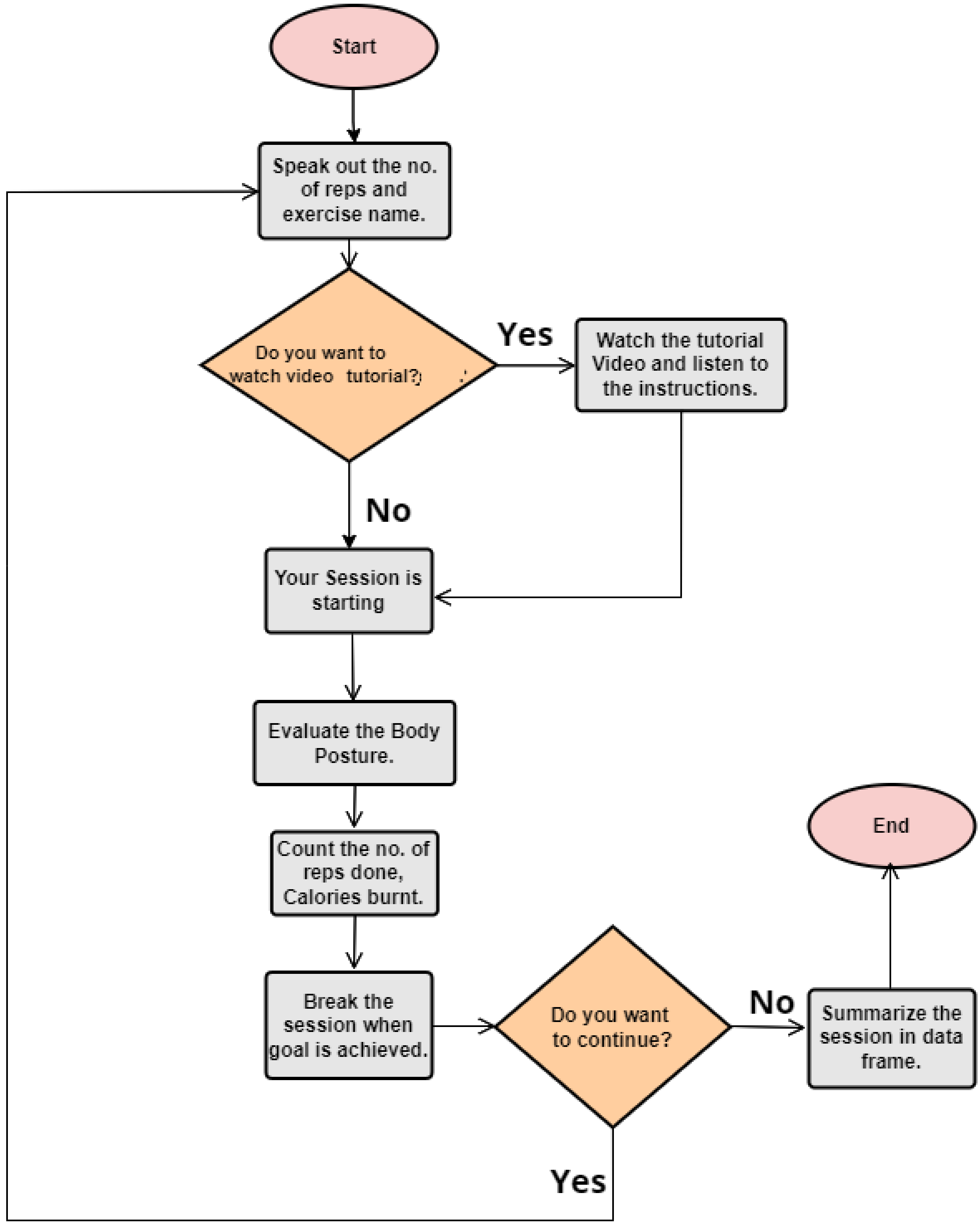
8. **Time** - The Python time() function **retrieves the current time**. The time is represented as the number of seconds since January 1, 1970.

4.DETAILED DESIGN

We propose a system that can be used with any camera based device capable of streaming the user workout to analyze.

- User will be asked to speak out the reps count and the exercise name.
- User will be asked if they want to watch any tutorial video related to that exercise or not.
- If the user wishes to watch then the tool will play a short tutorial for the user.
- Then, the body pose estimation image feed will start playing and it will count repetitions of the particular exercise and also the calories burnt.
- The real time body pose estimation model will also check if the user is correctly doing the exercise, if not then the count of reps will not increase.
- When the repetitions are completed by the user then the body pose estimation window and ask the user if they want to do more.
- If the user says to end the session then a summary dataframe will be displayed on the screen telling the user about the exercise completed with its corresponding repetitions.
- If the user wishes to continue, the voice assistant will wait for the user instruction and then perform accordingly.

5.FLOW CHART


Start
Speak out the no. of reps and exercise name.
Do you want to watch video tutorial?
Yes
Watch the tutorial Video and listen to the instructions.
No
Your Session is starting
Evaluate the Body Posture.
Count the no. of reps done, Calories burnt.
Break the session when goal is achieved.
Do you want to continue?
No
Summarize the session in data frame.
End
Yes

6.IMPLEMENTATION

6.1 Set up Voice Assistant

Importing Speech Recognition and pyaudio library for recognising the user input audio. Then, identify the exercise name and repetitions he/she wants to perform.

Tell the assistant if you want to watch the tutorial video or not.

6.2 Set Up Media Pipe

Importing Media pipe (It gives all the pose estimation libraries) for visualising our poses and for drawing the utilities.

6.3 Set up Video Capture

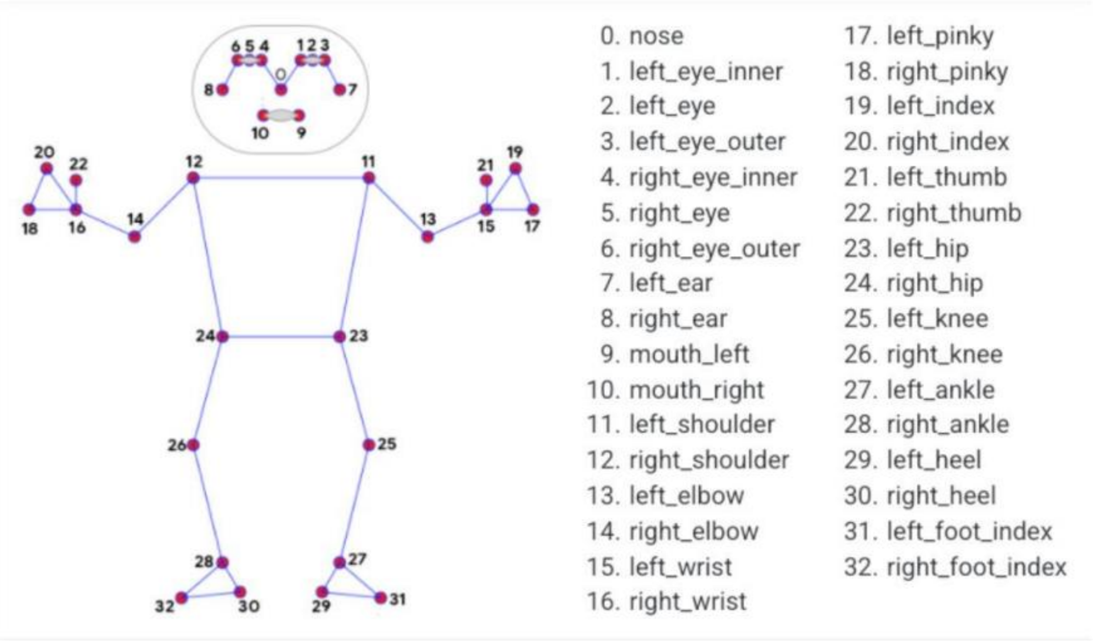
Accessing our webcam to read and capture the live feed to visualise the body pose of the user.

6.4 Estimate Pose

Applying, the media pipe components to recolor our image in the format of BGR to RGB. Then we will process our image and store the detections. Again, recolor the image from RGB to BGR.

6.5 Extract Joint coordinates

There are total of 33 landmarks present in our body. With the help of Media pipe components, we extract and draw those landmarks. Then, show the connection of different landmarks.



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: Visualizing the 33 human body landmark coordinates.

6.6 Calculate Angle between Coordinates

Calculate angle between landmarks that changes due to movement during exercises.

For example:- During Bicep curls there is movement between Shoulder, Elbow and Wrist. So, we need to calculate angle between them.

Let coordinates of Shoulder is (x_1,y_1) , Elbow is (x_2,y_2) and Wrist is (x_3,y_3).

$$\textit{Angle [in Radians]} = \tan^{-1}(y3-y2,x3-x2)-\tan^{-1}(y2-y1,x2-x1)$$

Then, convert the angle into degrees.

6.7 Count Number of repetitions

Set counter to zero and stage to none. Whenever, angle between those landmarks changes, change the stage to up and down and increase counter every time.

6.8 Calories burnt and time spent

Set Calories burnt to zero and whenever a count of repetition increases, increase calories by product of count with calories burnt in one repetition of exercise.

For example:-

Calories burnt in one repetition of bicep curls is 0.1875. Then calories burnt in every 10 repetitions will be 1.875.

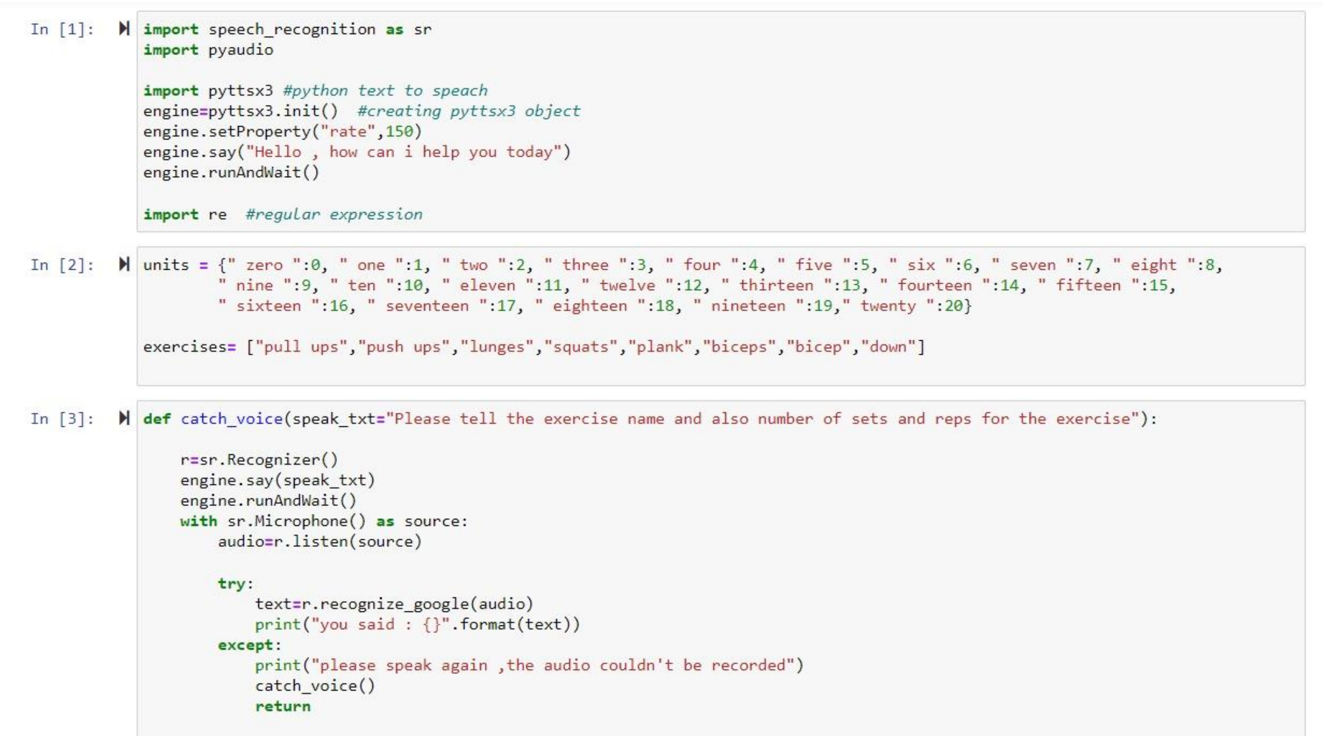
Using the Time library retrieve the current time for the user session.

6.9 Summarize the whole session

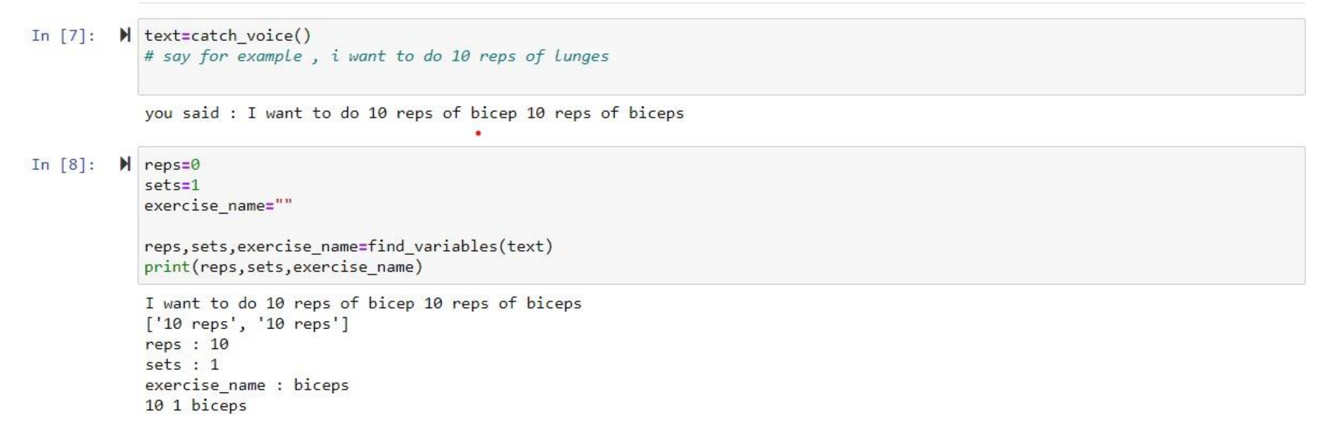
If the user wishes to stop the whole session of user will be stored in a CSV file.

7.RESULTS

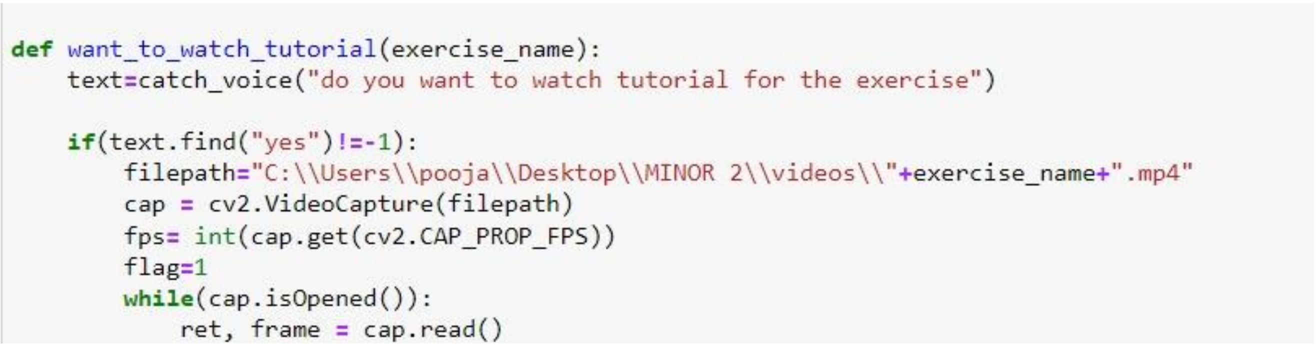
➢ Voice Assistant



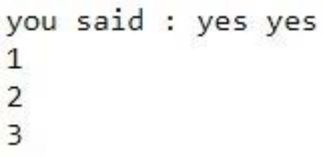
```
In [1]:  import speech_recognition as sr
         import pyaudio

         import pyttsx3 #python text to speach
         engine=pyttsx3.init()  #creating pyttsx3 object
         engine.setProperty("rate",150)
         engine.say("Hello , how can i help you today")
         engine.runAndWait()

         import re  #regular expression

In [2]:  units = {" zero ":0, " one ":1, " two ":2, " three ":3, " four ":4, " five ":5, " six ":6, " seven ":7, " eight ":8,
                 " nine ":9, " ten ":10, " eleven ":11, " twelve ":12, " thirteen ":13, " fourteen ":14, " fifteen ":15,
                 " sixteen ":16, " seventeen ":17, " eighteen ":18, " nineteen ":19," twenty ":20}

         exercises= ["pull ups","push ups","lunges","squats","plank","biceps","bicep","down"]

In [3]:  def catch_voice(speak_txt="Please tell the exercise name and also number of sets and reps for the exercise"):

             r=sr.Recognizer()
             engine.say(speak_txt)
             engine.runAndWait()
             with sr.Microphone() as source:
                 audio=r.listen(source)

                 try:
                     text=r.recognize_google(audio)
                     print("you said : {}".format(text))
                 except:
                     print("please speak again ,the audio couldn't be recorded")
                     catch_voice()
                     return
```



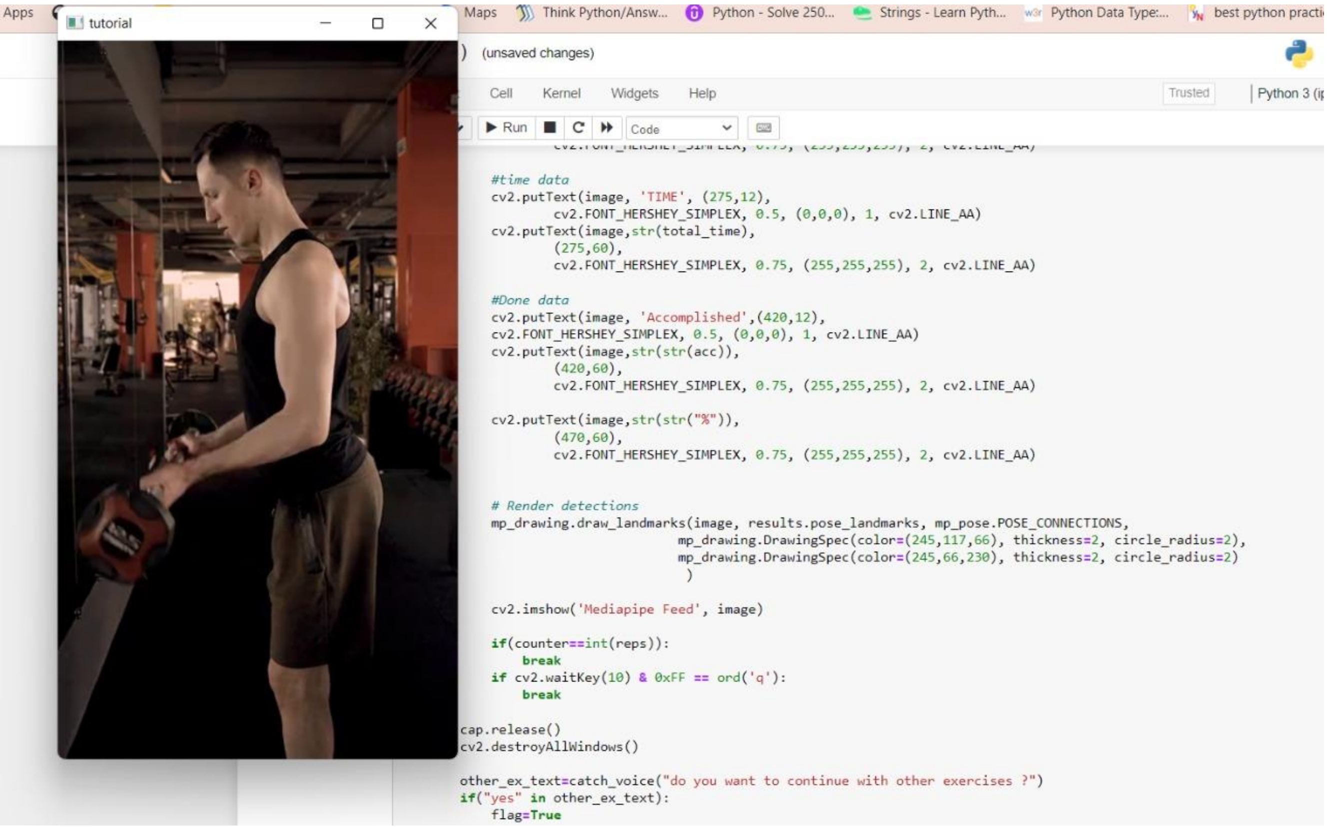
```
In [7]:  text=catch_voice()
         # say for example , i want to do 10 reps of lunges

         you said : I want to do 10 reps of bicep 10 reps of biceps

In [8]:  reps=0
         sets=1
         exercise_name=""

         reps,sets,exercise_name=find_variables(text)
         print(reps,sets,exercise_name)

         I want to do 10 reps of bicep 10 reps of biceps
         ['10 reps', '10 reps']
         reps : 10
         sets : 1
         exercise_name : biceps
         10 1 biceps
```



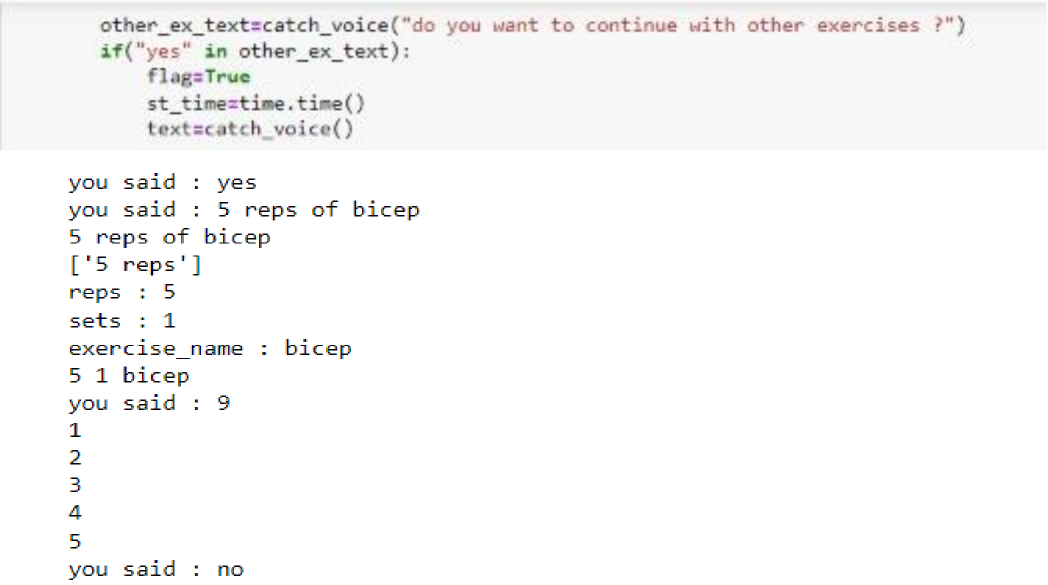
```
def want_to_watch_tutorial(exercise_name):
    text=catch_voice("do you want to watch tutorial for the exercise")

    if(text.find("yes")!=-1):
        filepath="C:\\Users\\pooja\\Desktop\\MINOR 2\\videos\\"+exercise_name+".mp4"
        cap = cv2.VideoCapture(filepath)
        fps= int(cap.get(cv2.CAP_PROP_FPS))
        flag=1
        while(cap.isOpened()):
            ret, frame = cap.read()
```



```
you said : yes yes
1
2
3
```

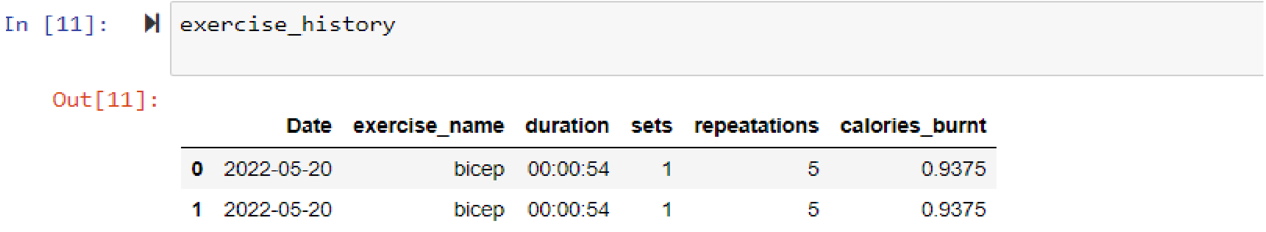


```
            cv2.FONT_HERSHEY_SIMPLEX, 0.75, (255,255,255), 2, cv2.LINE_AA)

    #time data
    cv2.putText(image, 'TIME', (275,12),
            cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0,0,0), 1, cv2.LINE_AA)
    cv2.putText(image,str(total_time),
            (275,60),
            cv2.FONT_HERSHEY_SIMPLEX, 0.75, (255,255,255), 2, cv2.LINE_AA)

    #Done data
    cv2.putText(image, 'Accomplished',(420,12),
    cv2.FONT_HERSHEY_SIMPLEX, 0.5, (0,0,0), 1, cv2.LINE_AA)
    cv2.putText(image,str(str(acc)),
            (420,60),
            cv2.FONT_HERSHEY_SIMPLEX, 0.75, (255,255,255), 2, cv2.LINE_AA)

    cv2.putText(image,str(str("%")),
            (470,60),
            cv2.FONT_HERSHEY_SIMPLEX, 0.75, (255,255,255), 2, cv2.LINE_AA)


    # Render detections
    mp_drawing.draw_landmarks(image, results.pose_landmarks, mp_pose.POSE_CONNECTIONS,
                            mp_drawing.DrawingSpec(color=(245,117,66), thickness=2, circle_radius=2),
                            mp_drawing.DrawingSpec(color=(245,66,230), thickness=2, circle_radius=2)
                             )

    cv2.imshow('Mediapipe Feed', image)

    if(counter==int(reps)):
        break
    if cv2.waitKey(10) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

other_ex_text=catch_voice("do you want to continue with other exercises ?")
if("yes" in other_ex_text):
    flag=True
```



```
other_ex_text=catch_voice("do you want to continue with other exercises ?")
if("yes" in other_ex_text):
    flag=True
    st_time=time.time()
    text=catch_voice()
```

```
you said : yes
you said : 5 reps of bicep
5 reps of bicep
['5 reps']
reps : 5
sets : 1
exercise_name : bicep
5 1 bicep
you said : 9
1
2
3
4
5
you said : no
```



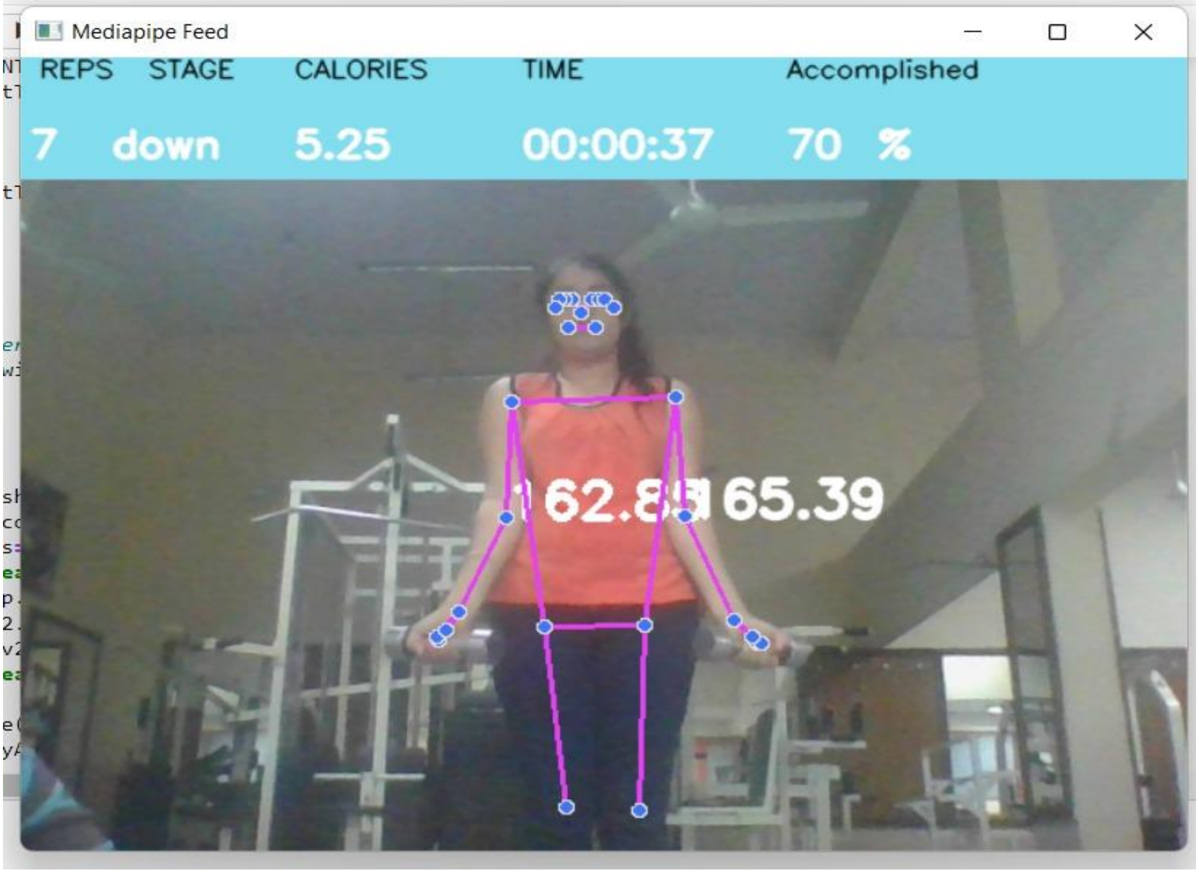
```
In [11]:  exercise_history
```

Out[11]:

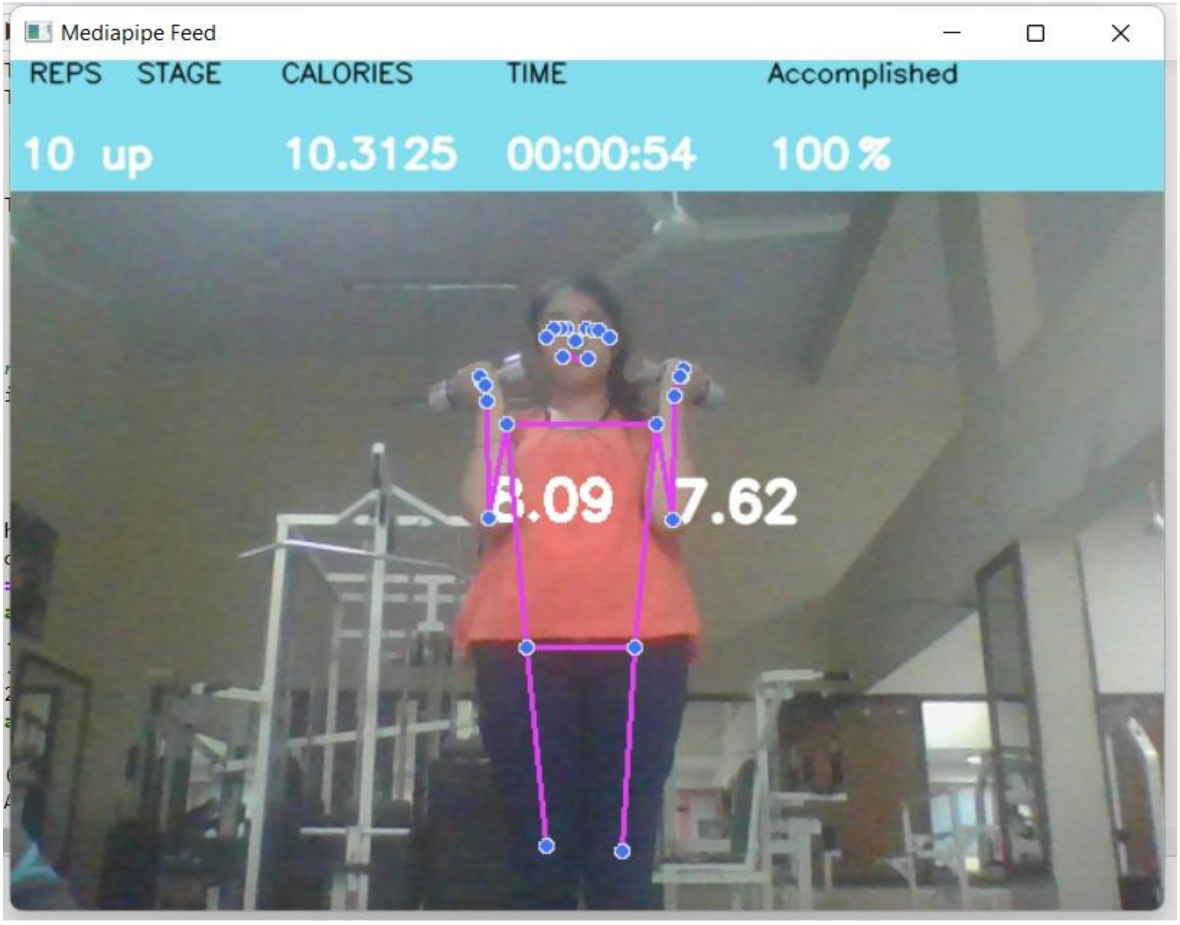
	Date	exercise_name	duration	sets	repeatations	calories_burnt
0	2022-05-20	bicep	00:00:54	1	5	0.9375
1	2022-05-20	bicep	00:00:54	1	5	0.9375

➢ Virtual Trainer

1. Curl Biceps-

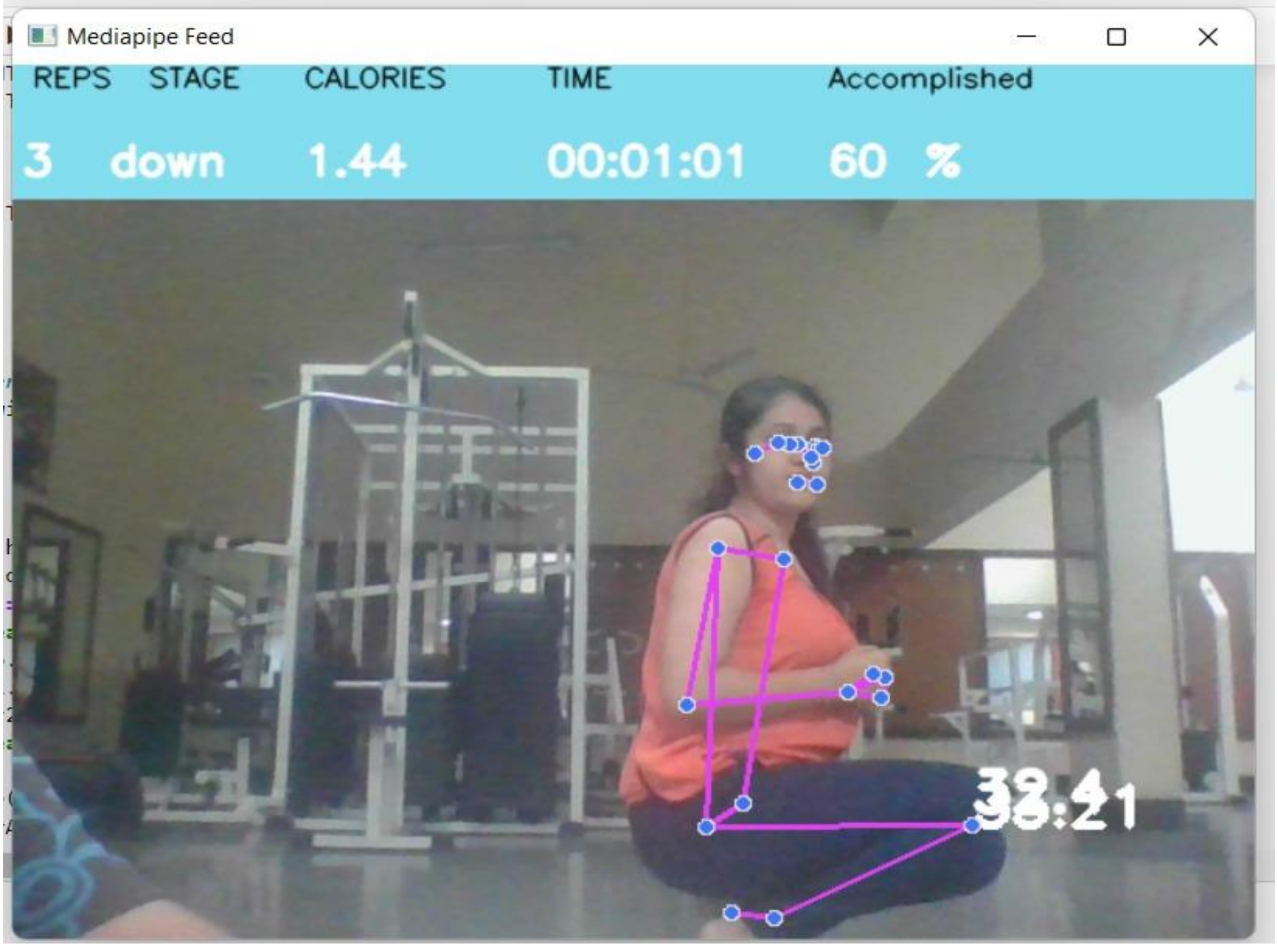


REPS	STAGE	CALORIES	TIME	Accomplished
7	down	5.25	00:00:37	70 %



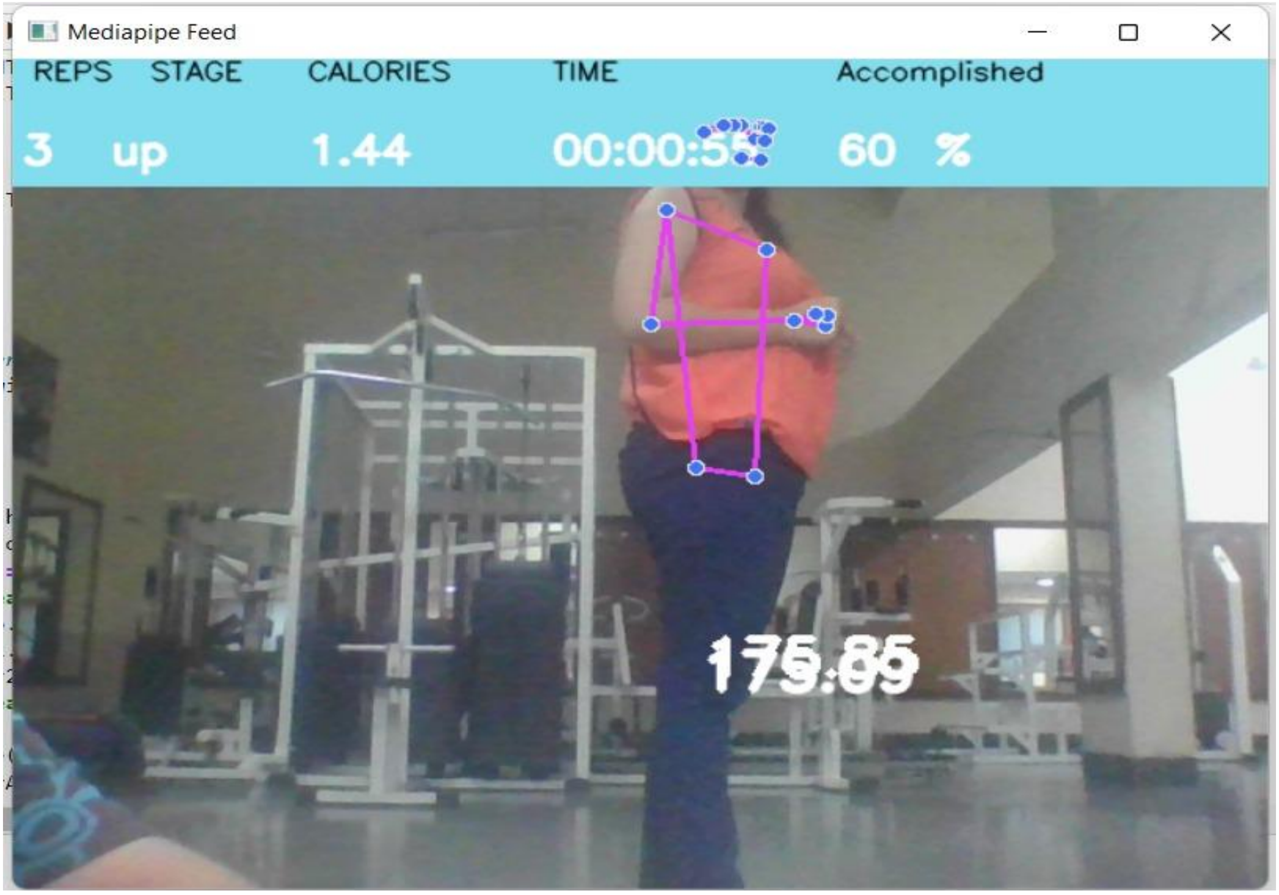
REPS	STAGE	CALORIES	TIME	Accomplished
10	up	10.3125	00:00:54	100 %

2. Squats-



REPS STAGE CALORIES TIME Accomplished

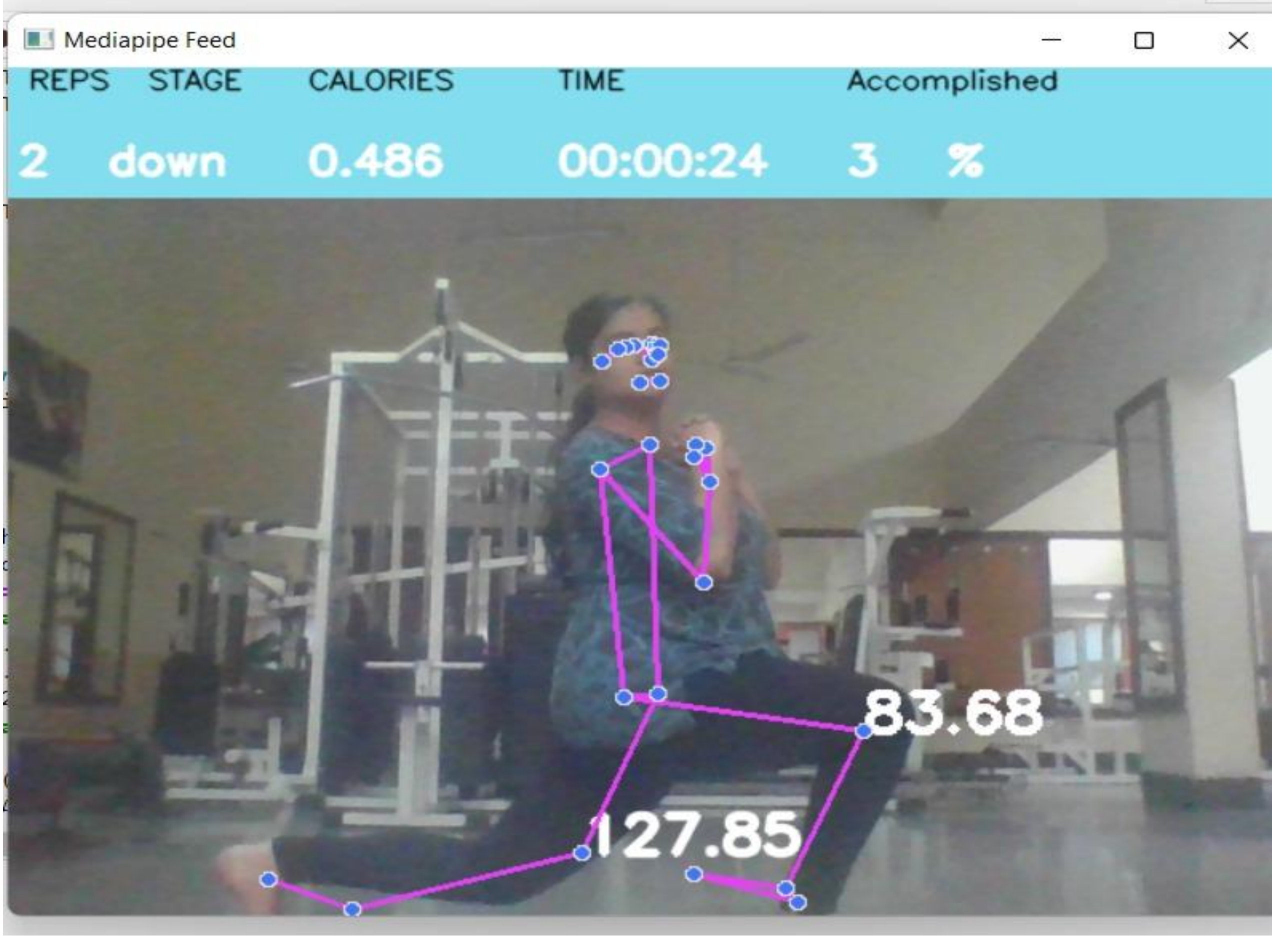
3 down 1.44 00:01:01 60 %



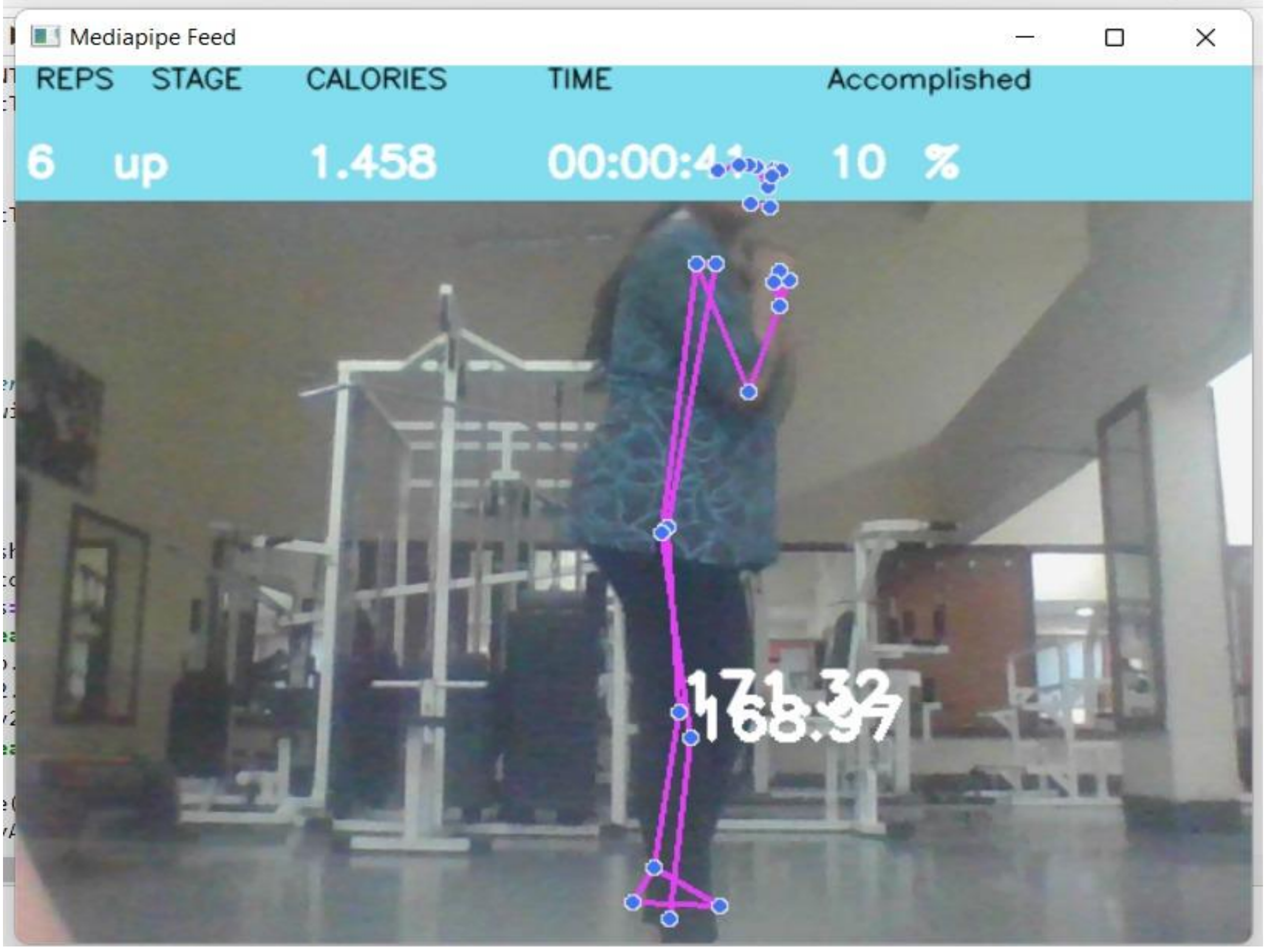
REPS STAGE CALORIES TIME Accomplished

3 up 1.44 00:00:5 60 %

3. Lunges-

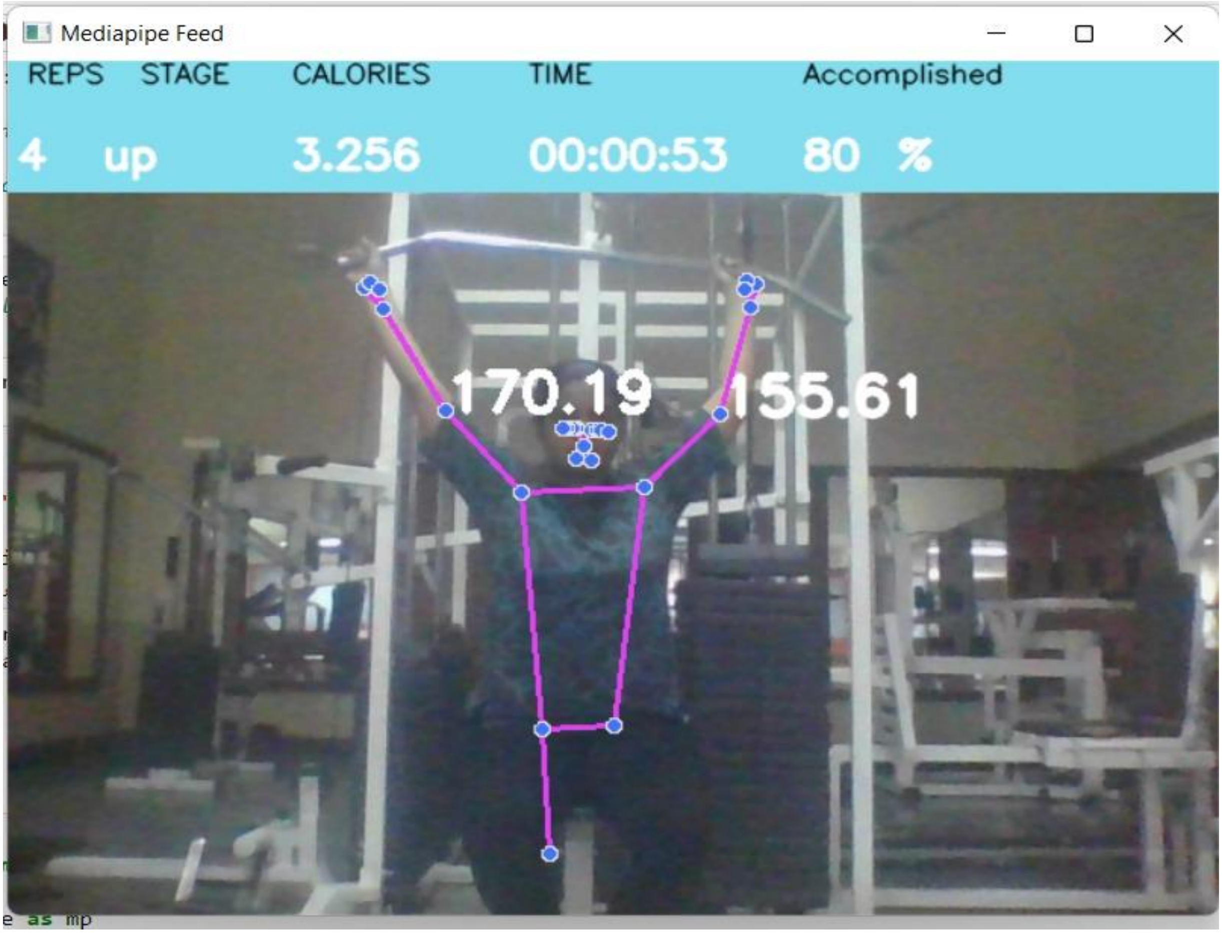


```
REPS   STAGE   CALORIES   TIME       Accomplished
2      down    0.486      00:00:24   3   %
```

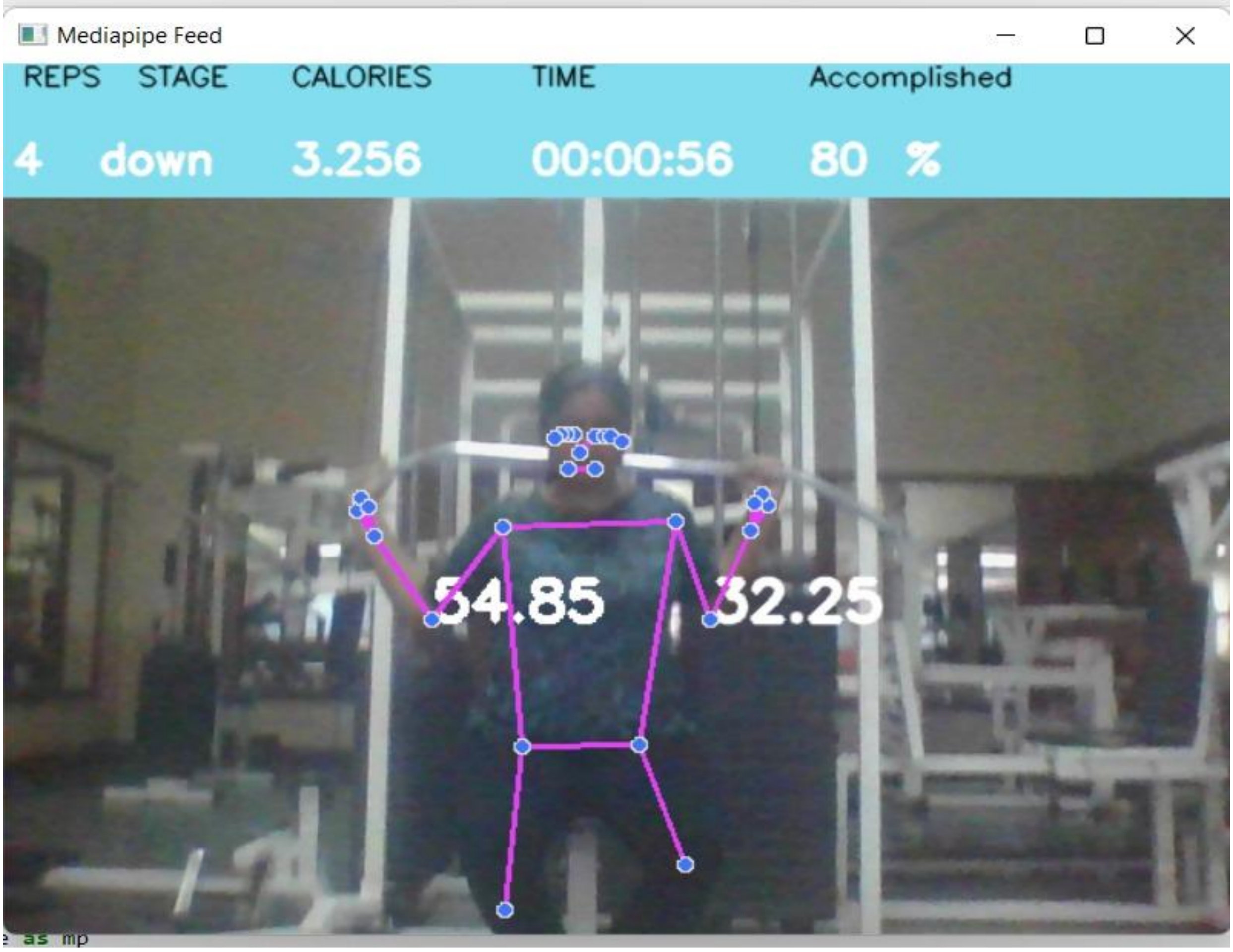


```
REPS   STAGE   CALORIES   TIME       Accomplished
6      up      1.458      00:00:41   10  %
```

4. Grip Lat Pulldown-



```
REPS   STAGE   CALORIES   TIME         Accomplished
4      up      3.256      00:00:53     80  %
```



```
REPS   STAGE   CALORIES   TIME         Accomplished
4      down    3.256      00:00:56     80  %
```

8. CONCLUSION

In this project, we present a tool for body recognition and repetition counting for different exercises. Our counting method was based on recognizing the exercises and analyzing the body angles required for that exercise. If the angles measured are correct count of repetitions will increase. This approach is straightforward and requires pre trained model of mediapipe and we have shown it works fine. Users will also be to know the amount of calories they burnt while doing the specific exercise. Users will also be able to analyze their performance by checking their previous record from the accessible csv file stored after every session.

The limitations and challenges we faced while working with the pretrained model is that we were not able to detect errors in the body posture of the users.

9.FUTURE WORK

This project can be extended with new features including:-

- **Application based**

This tool can be converted into a full fledged mobile application that a user can have on a go at any point they want to use it and it is more portable.

- **Multiple user at the same time**

In this tool we are focusing on only one user at a single time but in future it can be trained in a way that it can be used by multiple users at the same time.

- **Complete voice guided**

For now this tool is not completely voice guided but with more work on this it can be so that it is more convenient for the user to get live feedback of his/her session without going back and forth near the screen.

- **Better Accuracy**

As here we are using a pre trained data model its accuracy can't be considered perfect. So, for better accuracy we can train our own larger dataset.

- **User Specific Diet Planner**

As we know physical activities are not the only factor contributing to healthy lifestyle. We need healthy diet as well to maintain our health. So this tool can also have an extension that would suggest user to follow up the proper diet chart.

- **User Specific Fitness Planner**

As we know some physical activities require a specific body type and weight and can't be done by everybody especially beginners. So this extension will provide the user with specific list of exercises he/she should perform according to their body.

10. REFERENCES

[1] 3D Human Pose Estimation via Deep Learning from 2D annotations
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7785134

[2] The Analysis of the Impact of Yoga on Healthcare and Conventional Strategies for Human Pose Recognition
https://turcomat.org/index.php/turkbilmat/article/view/4032/3452

[3] Activity Recognition using Cell Phone Accelerometers
https://www.cis.fordham.edu/wisdm/includes/files/sensorKDD-2010.pdf

[4] A Novel Fitness Tracker Using Edge Machine Learning
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9140602

[5] Pose-Conditioned Joint Angle Limits for 3D Human Pose Reconstruction
https://openaccess.thecvf.com/content_cvpr_2015/papers/Akhter_Pose-Conditioned_Joint_Angle_2015_CVPR_paper.pdf

[6] Home based versus centre based cardiac rehabilitation: Cochrane systematic review and meta-analysis
https://www.bmj.com/content/340/bmj.b5631

[7] MM-Fit: Multimodal Deep Learning for Automatic Exercise Logging across Sensing Devices
https://dl.acm.org/doi/pdf/10.1145/3432701

[8] GymCam: Detecting, Recognizing and Tracking Simultaneous Exercises in Unconstrained Scenes
https://dl.acm.org/doi/pdf/10.1145/3287063

[9] Live Repetition Counting
https://openaccess.thecvf.com/content_iccv_2015/papers/Levy_Live_Repetition_Counting_ICCV_2015_paper.pdf

[10] RecoFit: Using a Wearable Sensor to Find, Recognize, and Count Repetitive Exercises
https://www.microsoft.com/en-us/research/wp-content/uploads/2016/12/p3225-morris.pdf

[11] **Predicting Physical Exercise Adherence in Fitness Apps Using a Deep Learning Approach**
https://www.mdpi.com/1660-4601/18/20/10769/htm

[12] **Recognition and Repetition Counting for LME exercises in Exercise-based CVD Rehabilitation: A Comparative Study using Artificial Intelligence Models**
https://www.preprints.org/manuscript/202007.0634/v1

[13] **Realtime Indoor Workout Analysis Using Machine Learning & Computer Vision**
https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8856547

[14] **Recognition and Repetition Counting for Complex Physical Exercises with Deep Learning**
https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6387025/pdf/sensors-19-00714.pdf

[15] **Automatic Classification of Squat Posture Using Inertial Sensors: Deep Learning Approach**
https://www.mdpi.com/1424-8220/20/2/361/htm

[16] **Recognition and Repetition Counting for Local Muscular Endurance Exercises in Exercise-Based Rehabilitation: A Comparative Study Using Artificial Intelligence Models**
https://www.mdpi.com/1424-8220/20/17/4791