Yoga Pose Detection Using Machine Learning

Piyush Yadav[1], Tarun Thanvi[2], Hamid Samani[3], Kirtida Naik[4]
[1, 2, 3, 4] Dept of Computer Engineering
[1, 2, 3, 4] VIVA Institute of Technology, Virar

Abstract- *The motivation for this project came from the quarantine and lockdowns. During these long months of lockdown, it's difficult for people to go out to exercise, run and do things they can to stay fit. Yoga is something that can be done indoors easily and has a lot of benefits. But it can be tricky to get any yoga pose correct if a person doesn't do yoga regularly. With the help of yoga pose detection, people can not only classify the pose but also check how well their posture as compared to a perfect yoga pose is. The Human Pose Estimator model detects humans and their poses in a given image. The model first detects the humans in the input image and then identifies the body parts, including nose, neck, eyes, shoulders, elbows, wrists, hips, knees, and ankles. Next, each pair of associated body parts is connected by a "pose line".*

Keywords- Machine Learning, Reviews, AI, CNN,, OPEN POSE MODEL.

I. INTRODUCTION

Yoga is one of the best exercises that we all do in our daily life Yoga-Tracker classifies various Yoga poses and not just static pictures but also real time yoga poses performed in front of the camera. It tracks how long you hold a yoga pose and keeps a track record which motivates you in your journey. It educates users of various poses and it's benefits along with proper guidance on how to perform them. It boasts of compatibility over all devices, Web, Android and IOS. Yoga pose estimation from video or a real-time feed plays a crucial role in various fields such as full- body gesture control, quantifying physical exercise, and sign language recognition. Pose Estimation is a general problem in Computer Vision where we detect the position and orientation of an object. This usually means detecting key point locations that describe the object Mediapipe is a framework mainly used for building multimodal audio, video, or any time series data. With the help of the MediaPipe framework, an impressive ML pipeline can be built for instance of inference models like TensorFlow, TFLite, and also for media processing function. For example, in the problem of face pose estimation (a.k.a facial landmark detection), we detect landmarks on a human face.

We will focus on human pose estimation, where it is required to detect and localize the major parts/joints of the body (e.g., shoulders, ankle, knee, wrist etc.).

The landmark model in MediaPipe Pose predicts the location of 32 pose landmark. The detector is inspired by our own lightweight BlazeFace model, used in MediaPipe Face Detection, as a proxy for a person detector. It explicitly predicts two additional virtual keypoints that firmly describe the human body center, rotation and scale as a circle. Also, various steps have taken to perform this architecture. Yoga detection is done with help of one main library media pipe. We have to pass image or video as an input through Integrated camera with help of that input i.e., image or video landmarks are generated on the image which helps to give skeleton of the body with that skeleton we will decide position that is a function will decide pose belongs to which category. We have used z co-ordinate to find pose in 3d dimension.

II. LITERATURESURVEY

learning provides an end-to-end architecture that allows automatic learning of key information from images. One popular deep learning model which has been widely used for pose.

A. Singh, A. Gupta, and J. Raheja [2], OpenPose is a real-time multi-person system presented by Perceptual Computing Lab of Carnegie Mellon University (CMU) to jointly detect a human body, hand, facial, and foot keypoints on single images. It is a major revolution in the field of pose recognition and provides the human body joint locations using convolutional neural networks (CNNs)-based architecture. CNNs are most widely used among deep learning architectures for vision applications. Traditional machine learning algorithms use handcrafted features, while CNNs learn some representational features automatically. U. Rafi, B. Leibe, J.Gall, and I. Kostrikov [3] describes that Yoga pose is detected based on the angles extracted from the Skeleton joints of TF pose estimation algorithm. 94.28% accuracy altogether was attained of all machine learning models. The data preprocessing and model training was done on Google Colab and Ubuntu 18.04.4 LTS terminal. Future ideas also include expansion of YOGI dataset on more yoga poses and implement deep learning modules for better performance. In addition to that an audio guidance system will also be included. features are stored in a CSV file and labeled accordingly.

S. Haque, A. Rabby, M. Laboni, N. Neehal, and S. Hossain [4]. yoga is that the part of existence during a number of the people. The human pose estimation is that the deep-rooted trouble in computer vision. That has exposed in many challenges inside the beyond. They have many fields to capture the posture like video surveillance, biometric, webcam, sort of the equipment, etc. The pose detection techniques have observed and it'll be used to identify the posture and thus the accuracy of the yoga posture in machine learning techniques. M. Islam, H. Mahmud, F. Ashraf, I. Hossain and M. Hasan [5], describes that the proposed models right now characterize just 6 yoga asanas. There are various yoga asanas, and subsequently making a posture assessment model that can be effective for all the asanas is a testing issue. The dataset can be extended my adding more yoga presents performed by people in indoor setting as well as open air. The exhibition of the models depends upon the nature of OpenPose present assessment which may not perform well in instances of cover between individuals or cover between body parts S. Patil, A. Pawar, and A. Peshave [6]: Yoga and sports have been attracting people groups for endless years; however, countless individuals receive Yoga as a significant aspect of their life from the most recent decade. Studies have been done on yoga pose identification; however, recognition of pose is still tricky because of the lack of a real-time benchmark dataset. The development of self-training systems with the current approaches, for example, using an RGB camera, is computationally expensive W. Gong, X. Zhang, J. Gonzàlez, A. Sobral, T. Bouwmans, C. Tu, and H. Zahzah [7], is detecting human figures in image and video (pose estimation techniques) are primarily based on variations of Convolutional Neural Networks. The are several approaches of how they can be used to recognize Yoga poses. However, some of them have a limited applicability, as their architecture is too heavy to be implemented in mobile devices, such as smartphones G. Ning, P. Liu, X. Fan and C. Zhan [8], the process of forming the Dataset, including a custom video annotation tool and a frame capturing script, was described, and the shape of the Dataset itself was demonstrated. In addition, the CNN model building and training process, along with the actual methods used for it, was analyzed, step by step. In the end, experimental results and findings, including the highest accuracy achieved by the CNN Model, were presented. D. Mehta, O. Sotnychenko, F. Mueller and W. Xu [9], This paper, to classify the sun salutation yoga poses and 4 machine learning model are classified. The yoga page is defected it supported the angle extracted from the pose estimation algorithm. The accurate result's gotten of machine learning models KNN. The get result of 96% of the machine learning Model. Future implementation of variety of individuals (group of persons) doing many pages finding the results of SVM models. Gao Z, Zhang H, Liu AA et al [10] describes that Pose estimation is a hot topic now-a-day. It is being used in video-surveillance system to sport analysis tasks. Some of the classical problem can be solved using pose estimation like: person count in a frame, fall detection, smart fitness tracking app etc. Basically, by using pose estimation we can observe the movement of human and take any decision. Before of deep learning arena HoG and SIFT based approach used in feature extraction. But because of CNN these feature extraction process become more accurate using lots of data. Lui Chang,Zast Rie[11], describes that Yoga pose is detected based on the angles extracted from the Skeleton joints of TF pose estimation algorithm. 94.28% accuracy altogether was attained of all machine learning models. The data preprocessing and model training was done on Google Colab and Ubuntu 18.04.4 LTS terminal. Future ideas also include expansion of YOGI dataset on more yoga poses and implement deep learning modules for better performance. Fangtre Rid,Altrene Mist[12],fall detection, smart fitness tracking app etc. Basically, by using pose estimation we can observe the movement of human and take any decision. Before of deep learning arena HoG and SIFT based approach used in feature extraction. But because of CNN these feature extraction process become more accurate using lots of data. C.Tu,G Rie[13],Yoga exercising becomes more and more popular throughout the years. This study introduces an idea of a smartphone application providing Yoga workout feedback, by displaying a few Yoga pose frames from a recorded workout session. To achieve it, a Convolutional Neural Network model is trained to detect the individual Yoga poses. Zhang Min,Rye[14], CNN Model, were presented. I experimented with more than a CNN models, of which the most successful one detects Yoga poses with the 91% accuracy when implementing the sigmoid activation function at the output layer, instead of the traditional SoftMax. However, the main contributions of this project are the tools for a dataset forming process, and the Dataset itself. J.Patel,Vijay Diwedi[15],Yoga and sports have been attracting people groups for endless years; however, countless individuals receive Yoga as a significant aspect of their life from the most recent decade. Studies have been done on yoga pose identification; however, recognition of pose is still tricky because of the lack of a real-time benchmark dataset.

III. METHODOLOGY

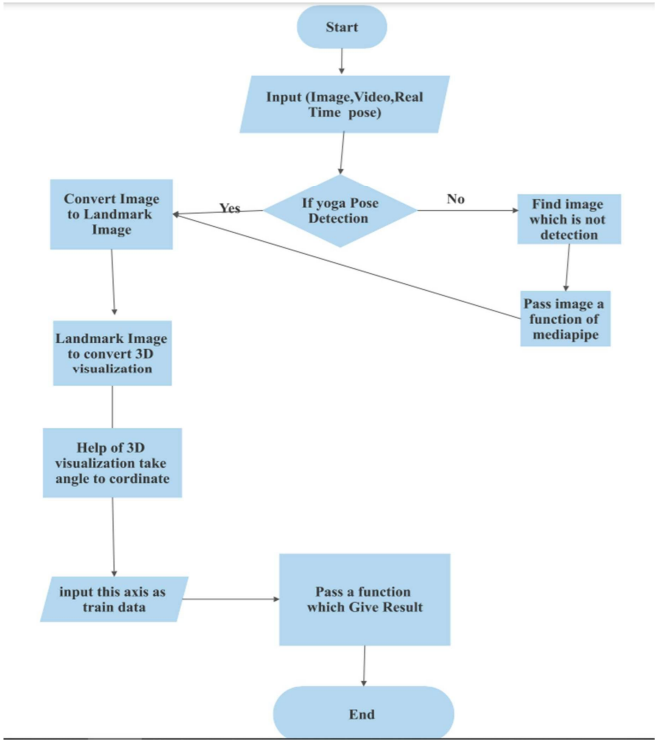

Fig 1: Block Diagram of Yoga Pose Detection

This chapter gives the overview of the Proposed system. Yoga is one of the best exercises that we all do in our daily life Yoga-Tracker classifies various Yoga poses and not just static pictures but also real time yoga poses performed in front of the camera. It tracks how long you hold a yoga pose and keeps a track record which motivates you in your journey. It educates users of various poses and it's benefits along with proper guidance on how to perform them. It boasts of compatibility over all devices, Web, Android and IOS. Yoga pose estimation from video or a real-time feed plays a crucial role in various fields such as full- body gesture control, quantifying physical exercise, and sign language recognition. Pose Estimation is a general problem in Computer Vision where we detect the position and orientation of an object.

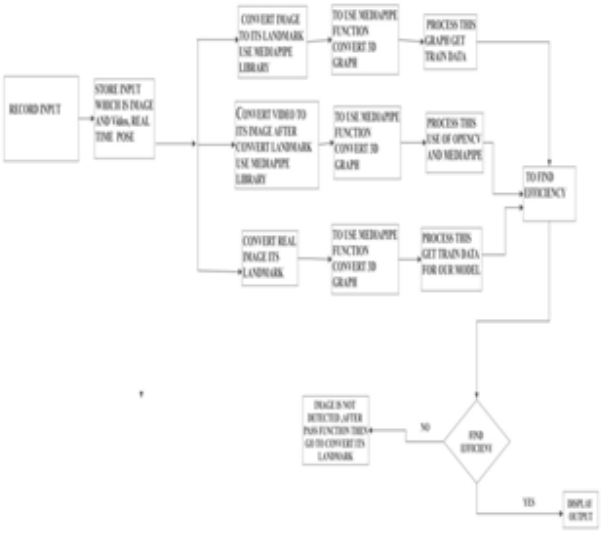

Fig 2: Block Diagram of Yoga Pose Detection

Shows the system flow diagram of the proposed system. The flow starts from Input take as an image, video and real time pose. After detection input, if the images detected then convert its landmark which is find joint. Help of landmark convert its 3d graph and take as a train data our model.

Shows the block diagram of the yoga pose Detection System. The block diagram has three Part divided. One is image of yoga pose and Second is Real input of Yoga pose. Third of video of yoga pose. The output of the two model will be given to check Detection condition if condition is true convert landmark, 3d graph and get train data and process it false then convert to image.

IV. RESULTS

1. Libraries

```
In [18]:
1 import math
2 import cv2
3 import numpy as np
4 from time import time
5 import mediapipe as mp
6 import matplotlib.pyplot as plt
7 from mpl_toolkits.mplot3d import Axes3D
```

Fig 3: Libraries

Fig 3 is we first of all import all the necessary packages and modules like NumPy, math, OpenCV, mediapipe and time, and after that we initialize the pose detection model.

2. Take Pose

```
mp_pose = mp.solutions.pose

pose = mp_pose.Pose(static_image_mode=True, min_detection_confidence=0.3, model_complexity=2)

mp_drawing = mp.solutions.drawing_utils
```

Fig 4: Take Pose

Fig 4 that is Install all Package to take Pose

3. Sample image:

```
sample_img = cv2.imread('D:\\PIYUSH\BE_Project\\images1\\cobrapose.jpg')

plt.figure(figsize = [10, 10])

plt.title("Sample Image");plt.axis('off');plt.imshow(sample_img[:,:,::-1]);plt.show()
```

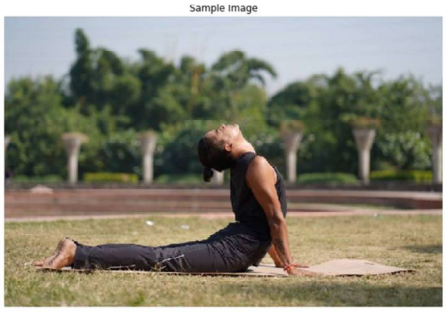

Fig 5: Sample image

Fig 5 is sample image using the function cv2.imread() and then display that image using the matplotlib library.

4. Co-ordinates of input image:

```
results = pose.process(cv2.cvtColor(sample_img, cv2.COLOR_BGR2RGB))

if results.pose_landmarks:

    for i in range(2):

        print(f'{mp_pose.PoseLandmark(i).name}:\n{results.pose_landmarks.landmark[mp_pose.PoseLandmark(i).value]}')
```

```
NOSE:
x: 0.5032839179039001
y: 0.37498009204864502
z: -0.12435511499643326
visibility: 0.9904091414642334

LEFT_EYE_INNER:
x: 0.4915270805358887
y: 0.3926351964473724
z: -0.10135632008314133
visibility: 0.9981725215911865
```

Fig 6: Co-ordinates of input image

Fig 6 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.

5. Landmarks on sample image:

Fig 7: Landmarks on sample image

Fig 7 detected landmarks on the sample image using the function mp.solutions.drawing_utils.draw_landmarks() and display the resultant image using the matplotlib library.

6.3-D image of sample input:

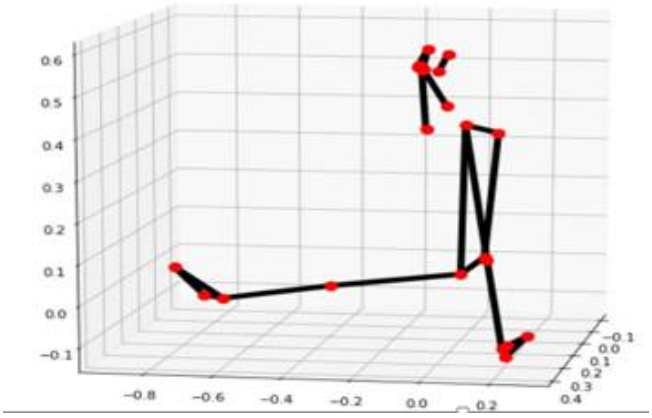

Fig 8: 3-D image of sample input

Fig 8 visualize the landmarks in three-dimensions (3D) using the function mp.solutions.drawing_utils.plot_landmarks().

7. Input using webcam:

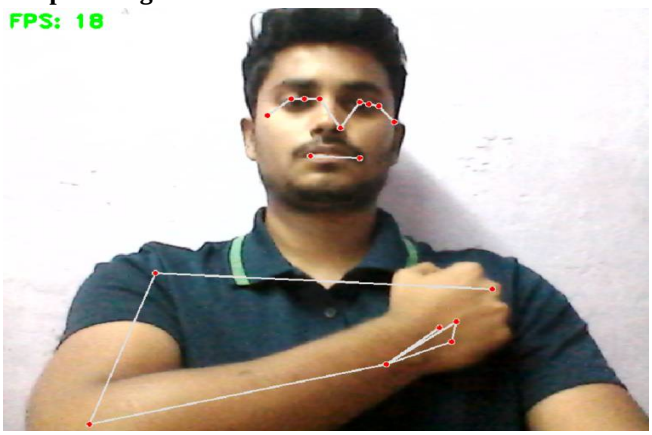

Fig 9: Input using webcam

Fig 9 is the image which is taken in real time by webcam, and it forms landmarks on it.

8. Result on sample image:

Fig 10: Result on sample image

Fig 10 is the output image of sample image. Here we give the sample image as input and our system is detecting and showing which pose it is.

9. Unknown Pose:

Fig 11: Unknown Pose

Fig 11 is the unknown pose. If we perform any random pose in front of webcam than it will show us unknown pose.

10. Real time input:

Fig 12: Real time input

Fig 12 is the actual output of our system here we just performing T-pose in front of webcam and it shows the output.

V. CONCLUSION

The proposed system will detect the yoga pose which we will provide it through an image as well as real time video. By using the function of pose detection, the system will detect and give the output us which yoga pose we are doing and what's the accuracy of it. We trained our model to capable for detecting the real time yoga video as well as any pose of yoga in image format. By using the Machine Learning libraries and mediapipe our system will firstly recognize the given data and after forming the landmarks on it system will make angles of every joints and using our pose detection function system will detect the Yoga Pose.

REFERENCES

[1] L. Sigal. " pose estimation", Ency. of Comput. Vision, Springer 2011.
[2] A. Singh, A. Gupta, and J. Raheja, "Real-time yoga recognition using deep learning", Neural Comput. and Appl., May 2019.
[3] U. Rafi, B. Leibe, J.Gall, and I. Kostrikov, "An efficient convolutional network for human pose estimation", British Mach. Vision Conf., 2016.
[4] S. Haque, A. Rabby, M. Laboni, N. Neehal, and S. Hossain, "ExNET: deep neural network for exercise pose detection", Recent Trends in Image Process. and Pattern Recog., 2019.
[5] M. Islam, H. Mahmud, F. Ashraf, I. Hossain and M. Hasan, "Yoga posture recognition by detecting human joint points in real time using microsoft kinect",
[6] S. Patil, A. Pawar, and A. Peshave, "Yoga tutor: visualization and analysis using SURF algorithm",Proc. IEEE Control Syst. Graduate Research Colloq
[7] W. Gong, X. Zhang, J. Gonzàlez, A. Sobral, T. Bouwmans, C. Tu, and H. Zahzah, "Human pose

estimation from monocular images: a comprehensive survey",

[8] G. Ning, P. Liu, X. Fan and C. Zhan, "A top-down approach to articulated human pose estimation and tracking",

[9] D. Mehta, O. Sotnychenko, F. Mueller and W. Xu, "XNect: real-time multi-person 3D human pose estimation with a single RGB camera",2019.

[10] Gao Z, Zhang H, Liu AA et al (2016) Human action recognition on depth dataset. Neural Computer.

[11] L. Migal. " yoga pose estimation", Ency. of Comput. Vision, Springer 2011.

[12] Aftab, H. Mahmud, F. Ashraf, I. Hossain and M. Hasan, "3D Yoga posture recognition by detecting human joint points in real time using microsoft kinect", IEEE Region 10 Humanit. Tech. Conf., pp. 668-67, 2019.

[13] C.Tu,G Rie real-time multi-person 3D human pose estimation with a single RGB camera",2019

[14] Zhang Min,Rye "Real-time yoga recognition using deep learning", Neural Comput. and Appl., May 2018.

[15] J.Patel,Vijay Diwedi deep neural network for exercise pose detection", Recent Trends in Image Process. and Pattern Recog., 2021.