**Mini Project Report on**



**Text Classification**



**Submitted in partial fulfillment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**Student Name University Roll No.**

**Sahil Budakoti 2021400**

***Under the Mentorship of***

**Ms. Meenakshi Maindola**

**Designation**

**Associate Professor**



**Department of Computer Science and Engineering**

**Graphic Era (Deemed to be University)**

**Dehradun, Uttarakhand**

**January 2024**



**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in the project report entitled **“Yoga Pose Detection”** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineeringof the Graphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of **Ms. Meenakshi Maindola, Associate Professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

Name University Roll number

Sahil Budakoti2021400

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | **1 - 2** |
| Chapter 2 | Literature Survey | **3** |
| Chapter 3 | Methodology | **4 – 6** |
| Chapter 4 | Result and Discussion | **7 – 9** |
| Chapter 5 | Conclusion and Future Work | **10** |
|  | References | **11** |

**Chapter 1**

**Introduction**

**Abstract – Yoga is an ancient practice that originated in India, which involves physical, mental, and spiritual exercise to enhance overall being. The importance of yoga poses is acknowledged around the whole world and proves its health benefits preached by ancient sages. Thus combining Computer vision techniques with your yoga pose provides a promising solution for accessing good human postures.**

**1.1 Background**

The field of yoga pose detection using mediapipe and OpenCV in machine learning is an innovative area of study, which merges the power of computer vision and artificial intelligence to rapidly identify and categorize various yoga poses with exceptional accuracy. This technology has immense potential to transform the way we engage in yoga by offering immediate feedback on our form, posture, and alignment. Its applications are not limited to yoga alone, as it can also be utilized in a range of industries like sports, healthcare, and fitness. This mini-project report would benefit from a detailed analysis of the technical aspects of this technology, a discussion of the challenges faced during its development, and an exploration of its numerous potential applications in diverse fields.

**1.2 Objectives**

The main objective of this project is to investigate and implement a machine-learning algorithm to create a model that can detect different yoga poses. As we all know yoga which was being practiced only in India, is now being practiced around the whole world and the most important trait of yoga is its correct posture. By using advanced deep learning techniques, this project emphasizes achieving higher accuracy on detecting different yoga poses, by taking samples in real-time and applying them hence creating a bar that intends to exceed the constraints of conventional techniques and better performance in the field of yoga.

**1.3 Motivation**

The motivation behind selecting Yoga pose detection lies my curiosity in about machine learning. Unlike many web development projects, this project features testing and training data from sources and then implementing it in real-life making. So many yoga practitioners, and yoga enthusiasts can rely on this tool for their daily use.

**1.4 Scope of the Project**

In today’s modern day, Machine learning and deep learning techniques have proven their worth for all around the world. We can now effectively use machine learning models to detect different structures images and videos. Thus I want to develop a machine-learning model that can accurately detect and recognize different yoga. This model is expected to significantly reduce the error in wrong yoga poses, thus providing a reliable model for daily use. This project involves data collection, preprocessing training and testing of the model, and evaluation of its performance.

**Chapter 2**

**Literature Survey**

**2.1 Overview of Pose Detection Techniques**

Pose estimation is very much interesting area in the field of computer vision. Most recent and popular work onpose estimation is by Deva Ramanan and Yi Yang

**2.1.1 Articulated pose Estimation With Flexible Mixtures of parts (By - Yi Yang, Deva Ramanan)[1] :**

This paper describes a method for pose estimation in stationary images based on part models. In this method they have used a spring model as a human model and calculated a contextual correlation between the model parts. One way to visualize the model is a configuration of body parts interconnected by springs. The spring like connections allow for the variations in relative positions of parts with respect to each other. The amount of deformation in the springs acts as penalty (Cost of deformation).

Most of the work done on action recognition from video requires RGB as well as Depth data to recognize the action.

**2.1.2 An Approach to Pose based Action Recognition (Chunyu Wang, Yizhou Wang and Alan L. Yuille) [2] :**

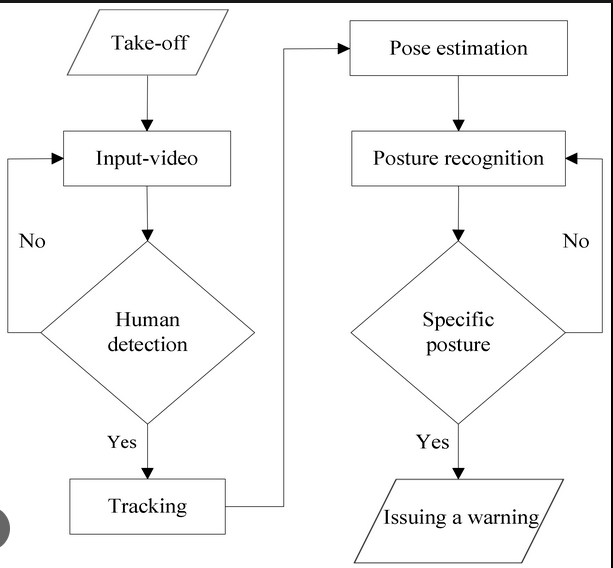
For representing human actions, it first group the estimated joints into five body parts namely Head, L/R Arm, L/R Leg. A dictionary of possible pose templates for each body parts is formed by clustering the poses of training data. For every Action class we distinguish some part sets ( Temporal and Spatial ) for representing the given action and then find the maximum intersection out of it..

**2.1.3** A significant area of research has also focused on detecting the poses of multiple people in one shot. Papandreou etal. [5] detect multiple poses through a two-stage process, first identifying possible bounding boxes for people, then detecting pose keypoints in each bounding box. In contrast, Cao et al. [3] use Part Affinity Fields to estimate poses of multiple people in a scene in real time without the need to identify individual persons first. Cao et al. [3] have open-sourced their work as a project called OpenPose, which we utilize for Pose Trainer.

**Chapter 3**

**Methodology**

Pose detection is a machine learning technique that involves the detection of yoga postures through the image and video entered by the user and detecting its poses through a variety of trained datasets and then categorizing the user posture according to the trained dataset through the use of different mediapipe landmarks and opencv and keras. This process involves data collection, data training, data preprocessing and model implementation over user inputted datasets. This model doesnot work upon pre built dataset instead it is built upon user build dataset, thus user can collect and train their individual data according to themself.

**3.1 Flowchart of Steps Involved:** ****

**3.2 Dataset Selection:**

The Dataset is created manually and is chosen for training and testing purposes. The dataset can be custom-created using “data\_training.py” from where different user can input different data for themselves according to their future preferences. User can offer a diverse set of images for their dataset and can store them in their folders.

**3.3 Data Preprocessing:** .

The first step in preprocessing in the data is to use openpose library to extract the key points of the poses in the video frame. Pose extraction in also done the online by user in real time and the key points in yoga poses are also done in real time through the user. OpenPose funs frame by frame of the video and the corresponding output for each frame is saved in .npy format.

The project includes .npy data that is retrieved and used by the model, thus when the user has to use that pose, the model can easily detect that pose through the entered data.

**3.4 Dataset Implementation:**

To facilitate efficient data loading, processing, and augmentation, “data\_collection.py” is created that collects the name for the user-inputted data and opens the screen to collect the data. Then the user can run “data\_training.py” to train that data and record its accuracy.

**3.5 Evaluation Metrics:**

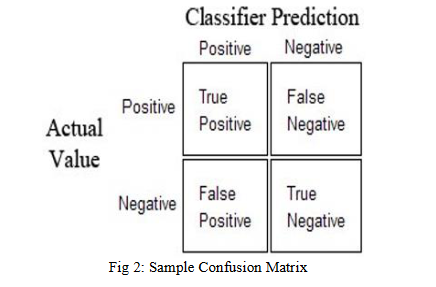
**3.5.1 classification Score-** The classification score refers to what is normally considered to be the accuracy of the model. This can be explained as the percentage of the total number of correct predictions in the total input samples.

****

In the case of multiclass classification, this metric gives good results when the number of samples in each class is almost the same

**3.6 Confusion Matrix**

The Confusion matrix represents a matrix that perfectly describes the accuracy of the model. There are four important terms when measuring the performance of a model.



1. True positive: Both the predicted value and the actual output are 1.

2. True Negative: Both the predicted value and the actual output are 0.

3. False positives: The predicted value is 1, but the actual output is 0.

4. False Negative: The predicted value is 0, but the actual output is 1.

The basic confusion matrix for binary classification. The diagonals of the matrix must always contain the maximum, as the diagonals represent a well-classified sample. In multiclass classification, each class represents a matrix row and column.

**Chapter 4**

**Result and Discussion**

This Yoga pose detection model is trained on user-inputted datasets according to ones individual hence one can inputter a varied number of data for their better accuracy in future preference classification. This model has achieved an accuracy of over 98% and precision around 99%. This high accuracy helps us to detect the posture correctly with over 99% precision.

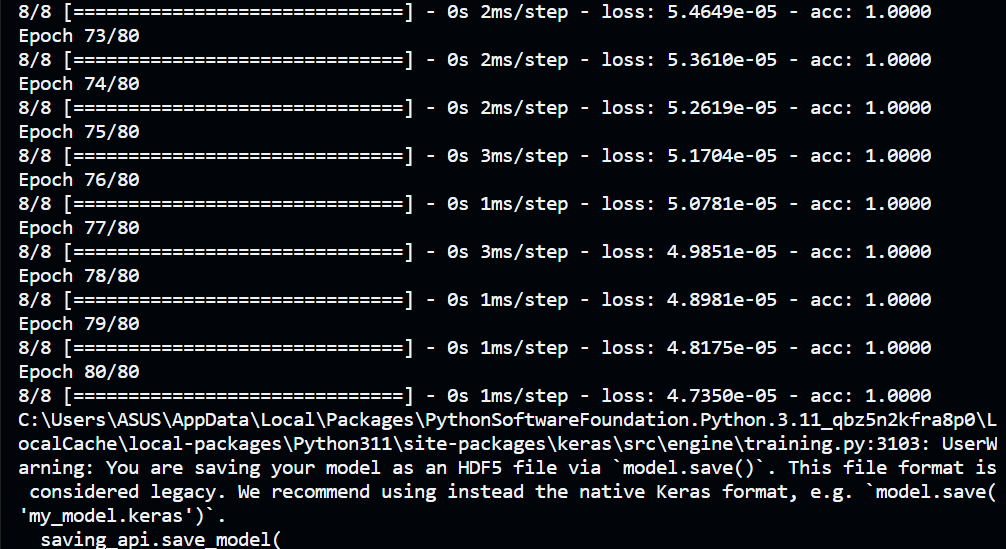
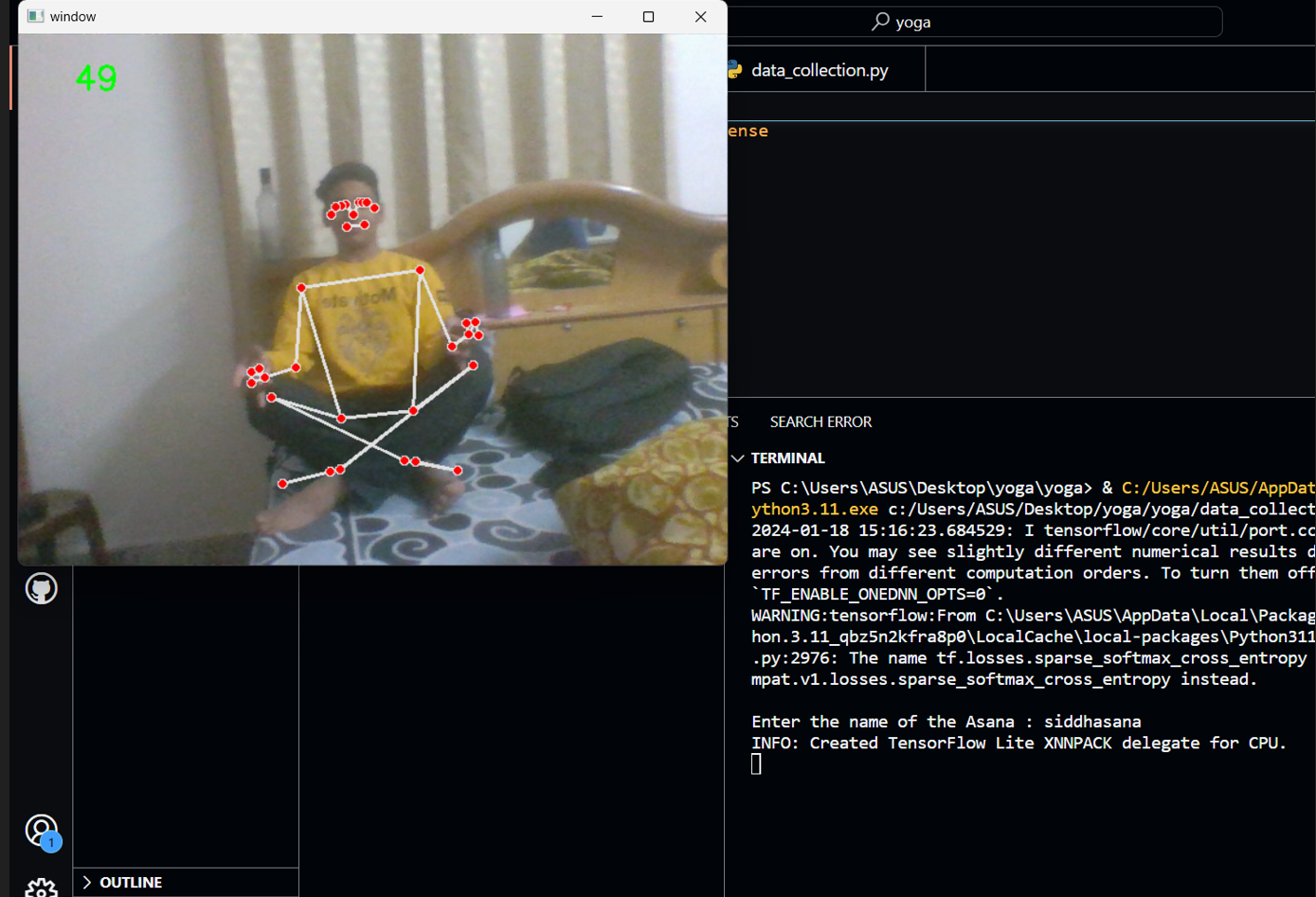


Fig 4.1

Fig4.1 depicts the accuracy of siddhasana over 80 sets of samples that’s being calculated through a machine learning algorithm.

**4.2 input:**

.



**4.3 Output:**

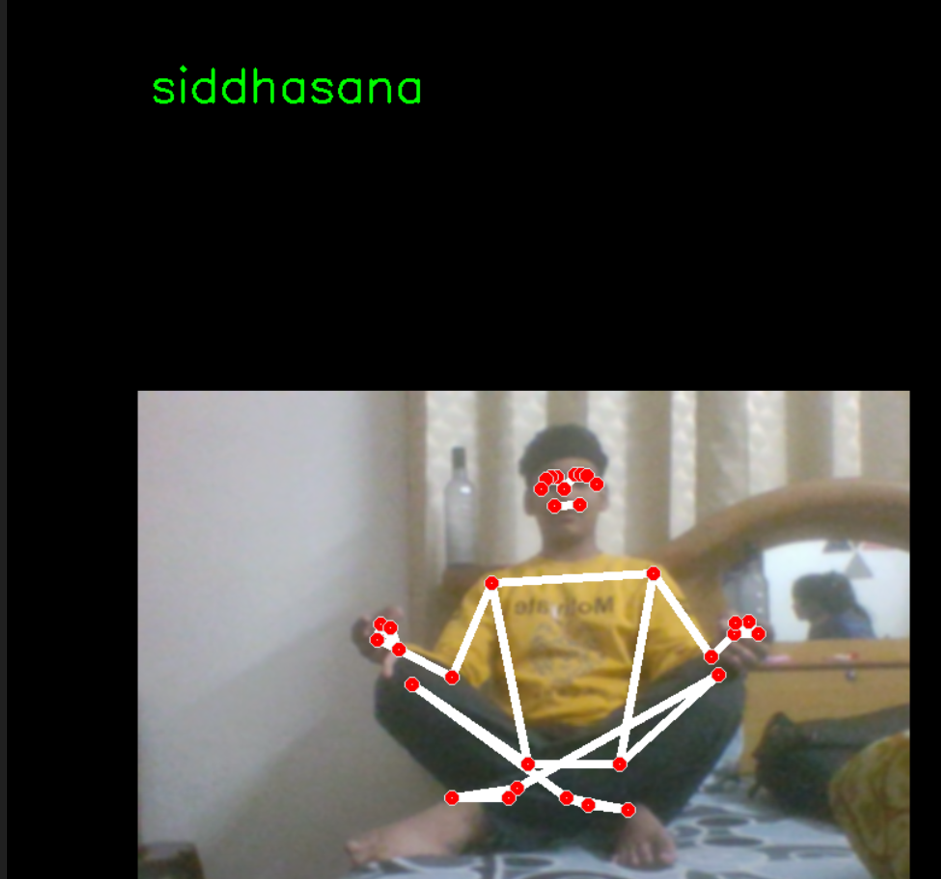
****

Fig 4.3 depicts the output of the video image performed by the user and the name above depicts the name of the asa

**Chapter 5**

**Conclusion and Future Work**

**5.1 Conclusion**

In conclusion, this project successfully utilizes the detection of yoga poses through different machine learning techniques with different classifications of poses to train the model Overall, this project significantly detects the pose and increases understanding and analysis of pose data for different yoga asanas. While building this model I learned a lot of things about how to create an ML model and how to deploy it that can be reliable for daily use.

**5.2 Future Work**

The proposed model presently classifies the most effective 6 yoga asanas. There are some yoga asanas, and therefore growing a pose estimation version that may be a hit for all of the asanas is difficult to trouble. The dataset may be improved by including extra yoga poses carried out through people who are no longer most effective in indoor placing but additionally outdoors. The overall performance of the fashions relies upon the exceptional OpenPose pose estimation which might not carry out nicely in instances of overlap among human beings or overlap among frame parts

The dataset can be improved significantly through a varied range of data inputted by the user.

**References**

[1] L. Sigal. “Human pose estimation”, Ency. of Computer Vision, Springer 2011

[2] S. Yadav A. Singh, A. Guptas , and J. Raheja, “”Real time yoga recognition using deep learning”.

[3] W. Wu, W. Yin, F. Guo, “Learning and self-instruction expert system for yoga”, Proc. Intl. Work

Intelligent Syst. Appl, 2010

[4] <https://www.youtube.com/bleedAI>

[5] <https://www.google.com/programminghut/machine-Learning-blogs/>

[6] M. Chen, M. Low, “Recurrent human poseestimation”, [Online]. Available:

https://web.stanford.edu/class/cs231a/prevprojects2016/final

[7] https://keras-.org/stable/

[8] https://www.youtube.com/programminghut

[9] https://www.geeksforgeeks.org/machine-learning/

[10] https://www.geeksforgeeks.org/machine-learning/open-pose