ISSN No:-2456-2165

AI Human Pose Estimation: Yoga Pose Detection and Correction

Rutuja Gajbhiye, Snehal Jarag, Pooja Gaikwad, Shweta Koparde
aStudent Professor, Department of Computer Engineering, Faculty of Computer Engineering, Pimpri Chinchwad College of Engineering and Research, 412101, Ravet, Pune, Maharashtra.

Abstract:- The most important of yoga poses is known around the world and proves the health benefits preached by ancient sages. As yoga becomes more important, yoga faces the following important challenges: Computer vision technology provides a promising solution for assessing human posture. However, these techniques are rarely used in the areas of health and exercise, and there are no specific references or projects. Named after yoga. This white paper describes the different technologies that can be used for pose estimation and summarize the best ways to use them based on the ease of use of your Android app application. The following describes the methodology used to provide yoga pose estimation in Android applications, how the app is modelled, and how each component works. Pose estimation is a branch of computer vision that deals with the recognition of the individual parts that make up the body (usually the human body). There are several ways to achieve this, The approach I use starts with passing the incoming image through a CNN classifier trained to look for people. When the human body poses are recognized, the pose estimation network searches for trained joints and limbs. The computer can then display the image to the user using markers that identify parts of the body.

***Keywords**:- Deep Learning, Machine Learning, Pose Estimation, OpenPose, PostNet, YogaPoses, CNN.*

I. INTRODUCTION

Human pose estimation is a computer vision technique used to predict the position/pose or joint position of a part of the human body. To do this pose, the joints which are also known as key points of the human body, such as the wrists, elbows, knees, and ankles, are defined in the image or video. When the image is input to the pose estimation model, the coordinates of these recognized body parts are identified as output with a confidence value that indicates the certainty of the human body.

In today's modern-day, Machine learning and deep learning techniques have been proven to be important for object discovery tasks. We can effectively use the model to recognize different important body parts and estimate the user pose in real-time. There are various methods we can use for pose estimation. This article describes the evolution of human pose estimation over the years and how Postnet is best suited for our project (Yoga pose estimation on Android). This project provides real-time pose estimation for client-side yoga pose estimation and correction. The real-time estimation can be done with Tensorflow. These points are used to draw the skeleton of a human pose, from which the angle between these points is derived. This allows you to effectively modify your yoga poses. This methodology is used in Android applications along with Google's Text-to-Speech and Speech-to-Text modules to enable users to practice yoga very effectively. Anyway, yoga problems, like some other exercises, are most important to practice carefully, as the wrong posture during a yoga meeting is ineffective and can probably be uncomfortable to everyone.

Human estimation is a hard-to-solve problem in the field of computer vision. It deals with locating human joints in an image or video to form a skeletal representation. Automatically detecting the pose of a person in an image is a difficult task because it depends on many aspects, including B. Image scale and resolution, lighting changes, background noise, clothing changes, the environment, and people-to-environment interactions. The application of human pose estimation that has fascinated many researchers in this field is exercise and fitness. This is an ancient practice that began in India but is now world-famous for its many mental, physical, and spiritual benefits. However, the problem with yoga is that, like any other exercise, incorrect posture during a yoga session can be unproductive and potentially harmful. Therefore, you need an instructor to supervise the session and correct your personal posture. All users don't have access to or resources to the instructor, for this, we can use an artificial intelligence-based application to identify yoga poses and provide personalized feedback to improve your personal changes. In recent years, human pose estimation has benefited from deep learning, with significant performance improvements. The machine learning approach provides an easier way to map structures than to handle dependencies between structures. Using machine learning and deep learning manually, we identified five exercise poses pull-ups, Swiss ball hamstring curls, push-ups, cycling, and walking.

One of the reviews that attracted numerous analysts in this area, Utilization, is Practice and Wellness. The various type of activity is Yoga, which is a deeply rooted practice started in India but now 444 is celebrating as a whole Benefits. Pose estimation from video plays an important role in overlaying digital content and information about the real world in augmented reality, enabling sign language recognition and whole-body gesture control. Pose estimation in fitness applications for yoga, dance, and fitness treatments includes different poses (eg hundreds of yoga asanas), numerous degrees of freedom, occlusions (eg body and other

ISSN No:-2456-2165

objects obscuring the limbs from the camera's view), and special for a number of reasons.

II. RELATED WORK

We identify several state-of-the-art methods for pose estimation that accurately estimate human poses under a variety of sensor configurations, shots, and counts of individuals per shot. Toshev et al. [7] were the first to use a deep neural network to improve pose detection, finding the location of each body joint using regression on CNN features. Newell et al. [4] introduce a stacked hourglass neu- ral network architecture that uses repeated bottom-up and top-down processing to achieve accurate single pose predictions. Wei et al. [8] propose a different architecture, using multiple convolutional networks to refine joint esti- mates over sequential passes. Instead of RGB camera data, Shotton et al. [6] use single depth maps captured by the Mi- crosoft Kinect to predict 3D positions of joints through an object recognition approach. Bogo et al. [2] estimate 3D pose, as well as 3D mesh shape, using just single RGB im- ages.

A significant area of research has also focused on detecting the poses of multiple people in one shot. Papandreou et al. [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.

Pose estimation allows us to analyze the static posture of humans, which will provide valuable information regarding posture correctness. Zell et al. [9] use an interesting approach for analysis of physical movements, where the body is represented as a mass-spring system and used to find the forces and torques that travel through the joints of the body. We have found that, by using exercise spec- ifications and feedback from professionals, we can take a simpler approach to physical analysis, analyzing the angles and distances between joint keypoints to provide important feedback to users without needing a full physical simula- tion.

We identify several state-of-the-art methods for pose estimation that accurately estimate human poses under a variety of sensor configurations, shots, and counts of individuals per shot. Toshev et al. [7] were the first to use a deep neural network to improve pose detection, finding the location of each body joint using regression on CNN features. Newell et al. [4] introduce a stacked hourglass neu- ral network architecture that uses repeated bottom-up and top-down processing to achieve accurate single pose predictions. Wei et al. [8] propose a different architecture, using multiple convolutional networks to refine joint esti- mates over sequential passes. Instead of RGB camera data, Shotton et al. [6] use single depth maps captured by the Mi- crosoft Kinect to predict 3D positions of joints through an object recognition approach. Bogo et al. [2] estimate 3D pose, as well as 3D mesh shape, using just single RGB im- ages.

A significant area of research has also focused on detecting the poses of multiple people in one shot. Papandreou et al. [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.

Pose estimation allows us to analyze the static posture of humans, which will provide valuable information regarding posture correctness. Zell et al. [9] use an interesting approach for analysis of physical movements, where the body is represented as a mass-spring system and used to find the forces and torques that travel through the joints of the body. We have found that, by using exercise spec- ifications and feedback from professionals, we can take a simpler approach to physical analysis, analyzing the angles and distances between joint keypoints to provide important feedback to users without needing a full physical simula- tion.

We identify several state-of-the-art methods for pose estimation that accurately estimate human poses under a variety of sensor configurations, shots, and counts of individuals per shot. Toshev et al. [7] were the first to use a deep neural network to improve pose detection, finding the location of each body joint using regression on CNN features. Newell et al. [4] introduce a stacked hourglass neu- ral network architecture that uses repeated bottom-up and top-down processing to achieve accurate single pose predictions. Wei et al. [8] propose a different architecture, using multiple convolutional networks to refine joint esti- mates over sequential passes. Instead of RGB camera data, Shotton et al. [6] use single depth maps captured by the Mi- crosoft Kinect to predict 3D positions of joints through an object recognition approach. Bogo et al. [2] estimate 3D pose, as well as 3D mesh shape, using just single RGB im- ages.

A significant area of research has also focused on detecting the poses of multiple people in one shot. Papandreou et al. [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.

Pose estimation allows us to analyze the static posture of humans, which will provide valuable information regarding posture correctness. Zell et al. [9] use an interesting approach for analysis of physical movements, where the body is represented as a mass-spring system and used to find the forces and torques that travel through the joints of the body. We have found that, by using exercise spec- ifications and feedback from professionals, we can take a simpler approach to physical analysis, analyzing the angles and distances

between joint keypoints to provide important feedback to users without needing a full physical simula- tion.

We identify several state-of-the-art methods for pose es- timation that accurately estimate human poses under a va- riety of sensor configurations, shots, and counts of indi- viduals per shot. Toshev et al. [7] were the first to use a deep neural network to improve pose detection, finding the location of each body joint using regression on CNN fea- tures. Newell et al. [4] introduce a stacked hourglass neu- ral network architecture that uses repeated bottom-up and top-down processing to achieve accurate single pose pre- dictions. Wei et al. [8] propose a different architecture, using multiple convolutional networks to refine joint esti- mates over sequential passes. Instead of RGB camera data, Shotton et al. [6] use single depth maps captured by the Mi- crosoft Kinect to predict 3D positions of joints through an object recognition approach. Bogo et al. [2] estimate 3D pose, as well as 3D mesh shape, using just single RGB im- ages.

A significant area of research has also focused on detect- ing the poses of multiple people in one shot. Papandreou et al. [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.

Pose estimation allows us to analyze the static posture of humans, which will provide valuable information regard- ing posture correctness. Zell et al. [9] use an interesting approach for analysis of physical movements, where the body is represented as a mass-spring system and used to find the forces and torques that travel through the joints of the body. We have found that, by using exercise spec- ifications and feedback from professionals, we can take a simpler approach to physical analysis, analyzing the angles and distances between joint keypoints to provide important feedback to users without needing a full physical simula- tion.

We have identified several state-of-the-art human pose estimation methods that accurately estimate human poses under different sensor configurations, exposures, and the number of subjects per exposure. They were the first to use deep neural networks to improve pose detection and use regressions of CNN function to locate joints in each body. They introduced a neural network architecture that uses bottom-up and top-down iterations to achieve accurate single-pose predictions. They propose an architecture that uses multiple convolutional networks to improve joint estimation in sequential paths. Instead of using RGB camera data, a single depth map captured by Microsoft Kinect is used to predict the 3D position of the joint through an object detection approach. Estimate 3D poses with RGB image.

A widespread place of studies has additionally centered on detecting the poses of a couple of human beings in a single shot. They locate a couple of poses via a two-level process, first figuring out feasible bounding bins for human beings, then detecting pose key points in every bounding box. In contrast, use Part Affinity Fields to estimate the poses of a couple of human beings in a scene in actual time without the want to perceive man or woman individuals first. They have open-sourced their paintings as a task referred to as OpenPose, which we make use of for Pose Trainer. Human Pose estimation research the static posture of humans, with the intention to offer treasured facts concerning posture correctness. We use an exciting technique for the evaluation of bodily movements, wherein the frame is represented as a mass-spring device and used to locate the forces and torques. They have observed that with the use of exercising from professionals, we will take a less difficult technique to bodily evaluation, reading the angles and distances among joint key points to offer crucial remarks to customers while not having a complete bodily simulation.

III. RESEARCH METHODOLOGY

This section details the methodology used to identify the model (Figure 1). The user interface allows users to learn about the yoga poses the system offers and their benefits. The users can perform or practice each yoga pose one at a time. Both textual and verbal instructions are given to the user to correct the pose. Live video feeds from webcams are used to capture user movements. PostNet is used to detect and correct yoga poses. PostNet is a machine learning model that provides a system with constant human posture detection. This system allows the user to first fix their position in front of the camera. The user can take a yoga pose as shown in the picture. This is where the user's yoga poses are captured. The key points are identified which are drawn on the video canvas. These key points are used to compare the user`s pose with the target yoga pose to see if there is any correction required. If the two poses have a high similarity status, then the pose of a user is treated as perfect.

If the user's yoga pose does not match the coordinates of the target yoga pose, the system will generate instructions for the user to modify their pose. The user can follow the instructions given by the instructor and correct the mistake. Text instructions are provided as input to the JavaScript speech synthesis API, which provides the user with verbal instructions for pose correction. After practicing yoga, the user can continue the session or end the practice session.

ISSN No:-2456-2165

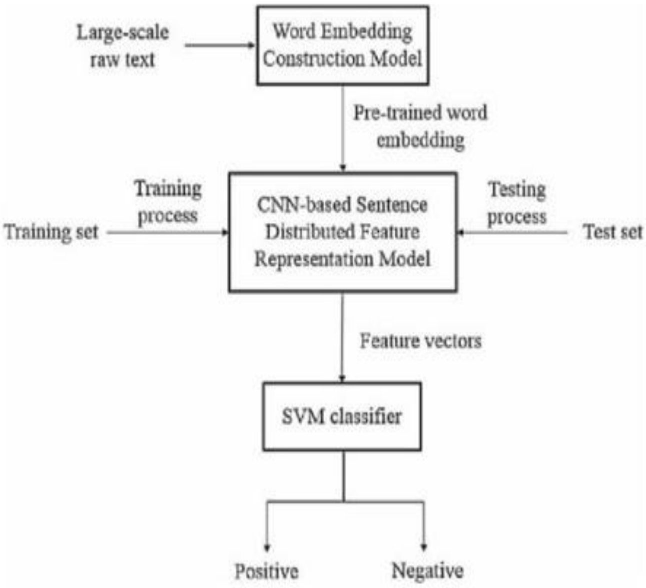

Fig 1: Detailed Research Methodology

A. Dataset Collection:

The dataset used for this project is part of an open-source collection and is publicly available. Yoga poses used in the project are Padmasana (Lotus Pose), Shavasana (Corpse Pose), Tadasana (Mountain Pose), Trikonasana (Triangle pose), and Vrikshasana (Tree pose) are different yoga poses. The rate at which the video was recorded is 30 FPS (frames/second). The videos should be recorded indoors at 4 meters from the camera. We have created a dataset that all subjects can perform yoga poses and use to build a robust yoga pose recognition system. The average length of the videos should be 45-60 seconds. Different yoga poses performed in Video frames are displayed in videos of different themes used for training, testing, and validation sets.

B. Data Preprocessing:

The first step in preprocessing the data is to use the OpenPose library to extract the key points of the pose in the video frame. Pose extraction is done offline, and online in real-time, and the key points identified from the input to the camera are sent to the model. OpenPose runs frame by frame of the video and the corresponding output for each frame is saved in JSON format. This JSON data contains the position of each person's body part identified in the video image. We used default settings to extract pose key points for ideal performance.

The project includes JSON data that is retrieved and stored in a NumPy array in a 45-frame sequence. 60% of the dataset was used for training, 20 was used for testing and 20 was used for validation. The training data contains 45 frames of 7989 sequences, each sequence containing the 2D coordinates of the 18 key points captured by OpenPose. The validation data consists of 84 such sequences and the test data contains 84 sequences.

IV. EVALUATION METRICS

A. 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 sample.

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

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

B. 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.

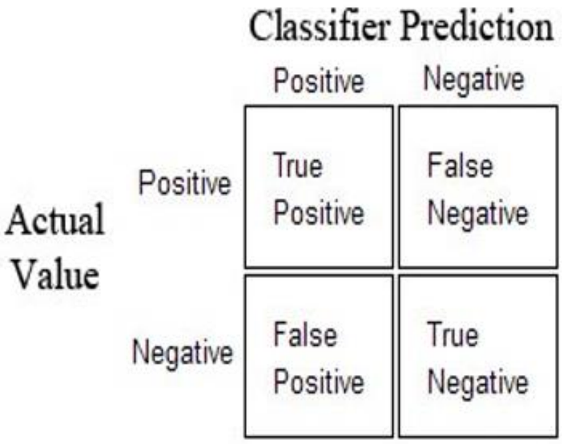

Fig 2: Sample Confusion Matrix

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. Four. 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.

C. Model Accuracy and model loss curve:

These curves, also known as learning curves, are most commonly used for models that learn step by step over time, such as neural networks. These neural networks represent the training and evaluation of validation data, and you can see how well the model can be trained and generalized. The model loss curve represents the minimization (loss) score. In other words, the lower the score, the better the performance of the model. The accuracy curve used in machine learning represents the maximized score. In these cases, we can say that the higher the score, the better the performance of the model. A good model loss curve is one in which training and validation losses are reduced to reach a stable point and there is a minimum gap between the final loss values. The curve that is more accurate and stable in training and validation, with minimal gaps between the final accuracy value is the best model.

V. PROPOSED SYSTEM

AI-based fitness yoga trackers are typically intended for use through devices with cameras that can record angular coordinates and capture more images during exercise performance. The usual algorithm for a tracker based on human posture estimation is: When users start using the fitness yoga tracker, the camera captures their movements during exercise. The captured photo and angular coordinates are split into individual frames and processed using a human pose estimation model. This model detects key points in the user's body and forms a virtual "skeleton" in 2D or 3D dimensions. The virtual skeleton is analyzed by geometry-based rules or other means to identify errors in the exercise method (if any). The user receives a description of the errors made and recommendations for eliminating them.

The proposed system is implemented in python using the OpenCV library and runs on a Lenovo Intel Core i3 CPU, 4 GB RAM, and Windows 10 64-bit operating system. A dataset containing 84 yoga asanas sets in a typical yoga posture is selected by the system using a regular webcam and made publicly available. A novel hybrid approach based on machine learning and deep learning classifiers. Step one contains, a support vector machine (SVM) is planned. This classifier uses machine learning prediction to improve the performance of ML algorithms. The second step is a convolutional neural network, which captures the human skeleton of poses and the user's target poses, and compares the two poses to obtain similarities. I imported libraries such as OS, Time, Keyboard, Array, and Mediapipe. Our hypothesis is that the coordinates of the various body parts of the human body from the images contain information to determine if the pose is being performed correctly or not.

VI. PROBLEM STATEMENT

Recently to stumble on and pick out Human motion popularity is so tough and with our fast paced lives in recent times human beings regularly select to workout at domestic however experience the want for an instructor to test their workout form. As those sources aren't constantly without problems available, human popularity may be used to create a self-paced workout software that allows human beings to research and exercise and workout properly. The Proposed device is used with Convolutional Neural Network for Recognizing Human Action Based on Yoga Pose Classification Using Image Processing and Deep Learning.

A. Objectives

The Objective of pose estimation is for monitoring the movement of human topics for distinct exercises. To look at the makes use of a pre-skilled model, with a purpose to use geometric evaluation to become aware of the exercising achieved inside a video, which may be in real-time. To look at the component which offer area vectors possibilities to calculate angles among frame parts. To degree the biomechanics of the human in an exercising or schooling video, and offer remarks on their routine.

B. Algorithm

A novel hybrid method primarily based totally on system gaining knowledge of classifiers became proposed. The first is the guide vector system (SVM), which employs system gaining knowledge of (ML) prediction to enhance the overall performance of ML algorithms. The 2d technique is to apply a convolutional neural network (CNN) to get the human skeletons of the consumer and goal pose and examine them to provide a similarity score.

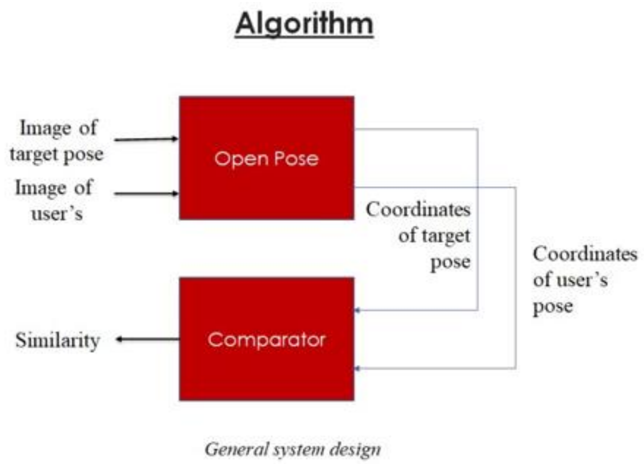

Fig 3: General System Design

C. Support Vector Machine (SVM):

SVM is a supervised tool learning model that is inherently a -elegance classifier. However, as most problems comprise multiple training, a multiclass SVM is often used. A multiclass SVM paperwork multiple elegance classifiers and differentiates the classifiers on the idea of the superb label vs. the rest (one-vs- rest or one-vs-all) or amongst each pair of training (one-vs-one). SVM performs the class thru growing a hyperplane in this kind of way that separation amongst training is as tremendous as possible. A default SVM has been knowledgeable on the training statistics with the radial basis function (rbf) kernel. Rbf is the default and most well-known kernel that could be a gaussian radial basis function. It offers more flexibility in contrast to exceptional kernels, linear and polynomial. The rate of the easy margin parameter C is 1 and the choice function is one-vs-rest. The keypoints captured using OpenPose are used as talents to SVM. These 18 keypoints are represented thru X and Y coordinates which makes the general extensive type of talents as 36 (18 * 2). The statistics is reshaped to make the extensive type of samples equal.

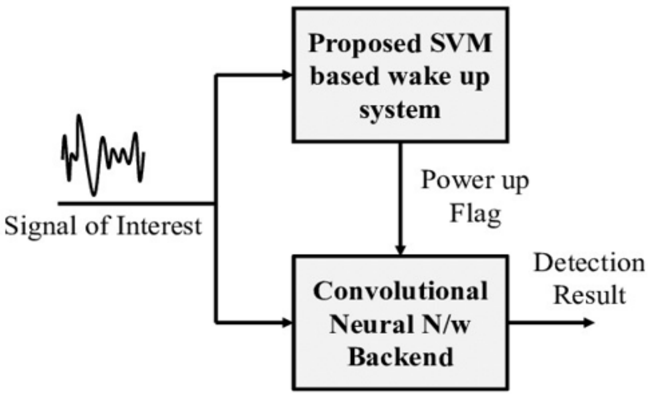

Fig 4: SVM Architecture

D. Convolutional Neural Network (CNN):

The convolutional neural community (CNN) is a segment of deep gaining knowledge of community. CNN represents a primary development in picture recognition. It works as Input - We have stay picture shooting of the user. Convolution - In this step, we are able to paintings on function detectors, which act as neural community filters. Then paintings on function maps. Pooling - In this layer, we lessen the picture stack to a smaller size. Pooling is finished after placing it via the release layer. Flattening – We convert information right into a 1-dimensional array to insert it into the subsequent layer. Fully Connected - We have completed education the community and might start to are expecting and examine the overall performance of the partition.

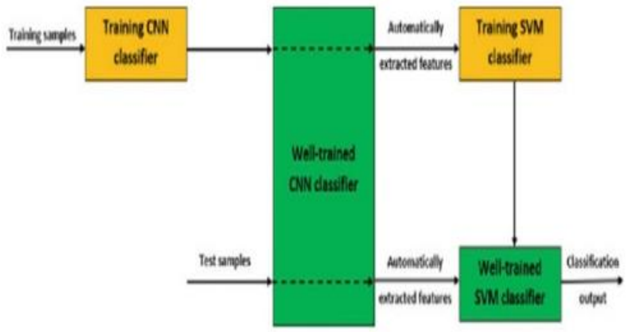

Fig 5: CNN Architecture.

VII. SYSTEM ARCHITECTURE

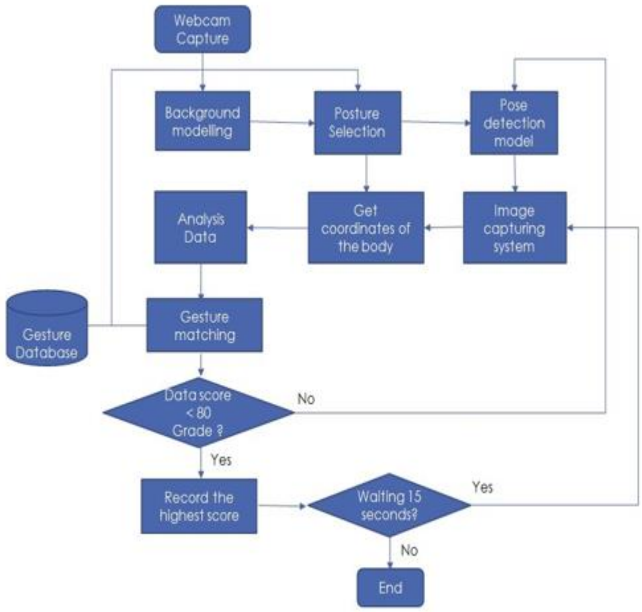

Fig 6: System architecture

Pose – Overall, PoseNet will move returned a pose object containing a list of key elements and an instance-diploma self warranty score for a detected person. Pose self warranty score - Determines the overall self warranty withinside the estimation of a pose, ranging amongst 0.0 to 1.0. It can be used to cowl poses which are not deemed strong enough. Keypoint - A part of a person's pose wherein the crucial aspect body issue is predicted, which consist of the nose, right ear, left knee, right foot, etc. Keypoint Confidence Score - Determines the self warranty of an predicted keypoint feature is accurate, ranging amongst 0.0 and 1.0. It can be used to cowl key elements are not deemed strong enough. Keypoint Position - 2D x and y coordinates withinside the proper input picture wherein a key aspect has been detected.

ISSN No:-2456-2165

VIII. LITERATURE SURVEY

[1] "Yoga Pose Assessment Method Using Pose Detection for Self-Learning" By M. C. Thar, K. Z. N. This paper recommends a Yoga pose evaluation approach to the usage of pose detection to assist the self-mastering of Yoga. This paper proposed a Performance Evaluation System as Yoga Pose Training System to assist the self-mastering of Yoga. This paper gives a way to discover yoga poses and the usage of pose discovery to assist in self-examine of yoga.
[2] "Real-time Yoga reputation the usage of deep mastering" By S. K. Yadav, A. Singh. An in-intensity hybrid mastering version has proposed the usage of CNN and LSTM to display yoga in real-time movies in which the CNN layer is used to extract capabilities from the important thing of everybody determined in open pose and observed through LSTM to offer transient predictions. This paper indicates a cell assistant yoga app primarily based totally on human key acquisition fashions for video chat.
[3] "Yoga Mobile Application Based on Yoga Detection" By Sylvie. The authors show a yoga assistant's mobile app based on a personal model where instructors guide and supervise their students to practice yoga with video chat. An in-depth learning model was proposed using CNN and LSTM to recognize yoga in real-time videos.
[4] "ML Learning Yoga pose in Video Sequences" By Jozsef. The issue approach is by studying the production model of normal motion patterns using multiple sources with limited control. With unencrypted performance, we suggest two autoencoders so that they can operate without minimal guidance.
[5] "Real-Time Detection in Crowded Area" By Ammar Ladjailia. A fully unsupervised dynamic coding approach for detecting unusual events in videos based on online constructability of query signals from learned event dictionaries.

IX. MODEL PERFORMANCE AND RESULTS

A. Support Vector System (SVM):

SVM is a supervised gadget getting to know version this is inherently a -elegance classifier. However, for maximum issues contain in training, a multiclass SVM is frequently used. A multiclass SVM paperwork a couple of elegant classifiers and differentiates the classifiers on the idea of the awesome label. The relaxation (one-vs- relaxation or one-vs-all) or among every pair of training (one-vs-one). SVM plays the type with the aid of using developing a hyperplane in any such manner that separation among training is as huge as possible.

A default SVM has been educated at the education records with the radial foundation characteristic (RBF) kernel. Rbf is the default and maximum famous kernel that's a Gaussian radial foundation characteristic. It offers extra flexibility compared to different kernels, linear and polynomial. The key points captured in the usage of OpenPose are used as capabilities to SVM. These 18 key points are represented with the aid of using X and Y coordinates which makes the full wide variety of capabilities 36 (18 * 2). The records are reshaped to make the wide variety of samples equal.

➢ *Results:*
- Train accuracy: 0.9953
- Validation accuracy: 0.9762
- Test accuracy: 0.9319

➢ *Analysis:*

The coaching accuracy of the model is pretty high at 0.99. There's a small decrease within the validation and check the accuracy, however, the results are still good. We will see in the confusion matrix that almost all categories are classified properly apart from tadasana (mountain cause). Out of 17,685 frames for tadasana, 6992 are misclassified as trikonasana (tree pose) and similarly, there is some incorrect classification for vrikshasana. This might be owing to the similarity in the poses as each of them need a standing position and conjointly the initial pose formation is similar.

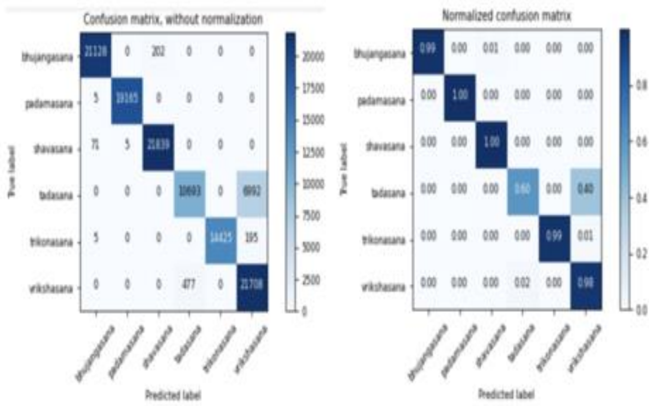

Fig 7: SVM

B. Convolutional Neural Network (CNN):

The input form is a pair that signifies the 32 key points having X and Y coordinates. Batch standardization is applied to the output of the CNN layer in order that the model converges faster. We tend to even have a dropout layer that forestalls over-fitting by indiscriminately dropping some fraction of the weights. The performance used is a corrected linear measure which is applied for feature extraction on key points of every frame. The ultimate output is planar before being passed to the dense layer with soft-max activation Associate units wherever each unit represents the chance of yoga created in cross-entropy terms for all categories.

This can be used because it permits measuring the performance of the output of the densely connected layer with soft-max activation. This loss function is employed for multi-category classification, and as we've multiple yoga pose classes, it is sensible to use categorical cross-entropy. Finally, to manage the educational rate, an optimizer with an initial learning rate of 0.0001 is used. The overall variety of epochs that the model is trained in is 100.

ISSN No:-2456-2165

➢ *Results:*

- Train accuracy: 0.9841
- Validation accuracy: 0.9910
- Test accuracy: 0.9868

➢ *Analysis:*

The coaching, validation, and take a look at the accuracy of the model are nearly the same, just about 0.99. The confusion matrix additional shows that the model will a superb job of classifying all samples correctly, aside from some samples in trikonasana that are misclassified as tadasana, resulting in 93misclassifications in CNN. However, the model loss curve higher than shows a rise within the validation loss and a decrease in the training loss which shows that there's some overfitting.

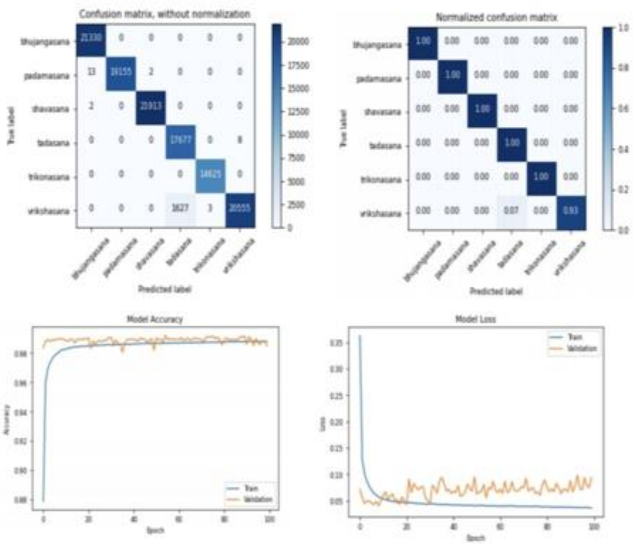

Fig 8: CNN

X. RESULTS AND ANALYSIS

The coaching, validation, and check accuracy of the model are nearly the same, close to 0.99. The confusion matrix shows that the model will a wonderful job of classifying all samples correctly, aside from some samples in trikonasana that are misclassified as tadasana, resulting in ninety-three accuracies for vrikshasana. There are fewer misclassifications in CNN. However, the model loss curve on top shows a rise in the validation loss and a decrease in the training loss which shows that there's some overfitting.

The coaching accuracy of the model is high at 0.98. There's a small loss in the validation and check of the accuracy, however, the results are still good. We are able to see in the confusion matrix that almost all categories are classified properly apart from tadasana (mountain create). Out of 17,695 frames for tadasana, 6982 are misclassified as trikonasana and equally, there is some incorrect classification for vrikshasana. This might be thanks to the similarity in the poses as each of them needs a standing position and additionally, the initial pose formation is similar.

A. *Home Page:*

Description: On visiting the home page, verbal greetings are provided to the user. The user can click on the GET STARTED button to proceed with the practice session, which navigates to the PRACTICE NOW page.

B. *Practice Now Page:*

Description: The practice now page will show all the poses offered by the system and the guidelines to be followed by the user. Here, the user can choose a particular pose or they can click on the START button to perform all the yoga poses in a sequence.

C. *Fixing of Position Page:*

Description: shows the fixing of position page where the instructions to the user are given to align the eye and ankle key points, so the user's position is fixed in front of the camera. It will be done to ensure the user's entire body (head to toe) is visible in the camera.

D. *Practicing a Pose*

Description: While practicing a pose, first the user's pose is identified and the generated key points and the skeleton is drawn on the video canvas. Instructions are provided for the user to do the yoga pose correctly. Upon successfully performing the yoga pose, the system tells the user to hold the pose and breathe deeply.

XI. FUTURE SCOPE

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 now no longer most effective in indoor placing but additionally outdoor. The overall performance of the fashions relies upon the exceptional of OpenPose pose estimation which might not carry out nicely in instances of overlap among human beings or overlap among frame parts. A transportable tool for self-schooling and real-time predictions may be carried out for this system. This painting demonstrates the pastime's reputation for realistic applications. A method similar to this could be applied for pose reputation in responsibilities inclusive of sports, surveillance, healthcare, etc. Multi-character pose estimation is entirely new trouble in itself and has plenty of scope for research. There are plenty of eventualities in which unmarried character pose estimation might now no longer suffice, for instance, pose estimation in crowded eventualities might have more than one folk as a way to contain monitoring and figuring out the pose of every individual. A lot of things inclusive of background, lighting, overlapping figures, etc. that have been mentioned in advance on this survey might in addition make multi-character pose estimation difficult.

ISSN No:-2456-2165

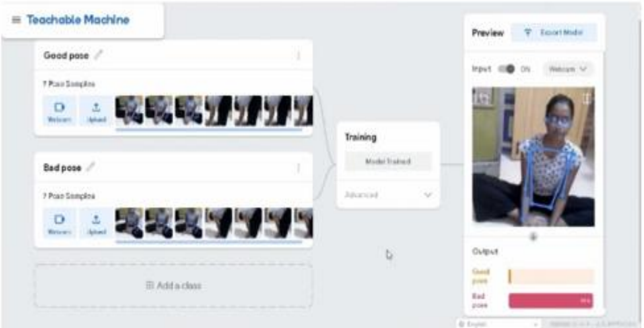

Fig 9: Yoga Pose 1

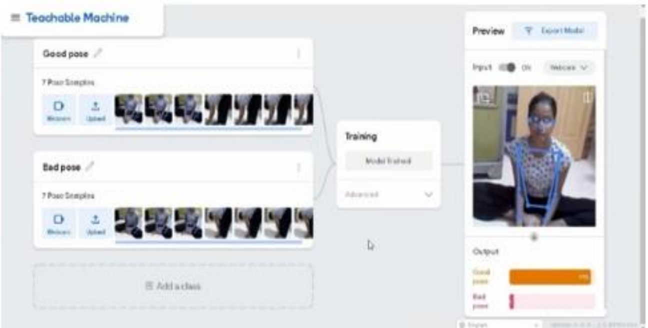

Fig 9: Yoga Pose 2

Fig 9: Yoga Pose 3

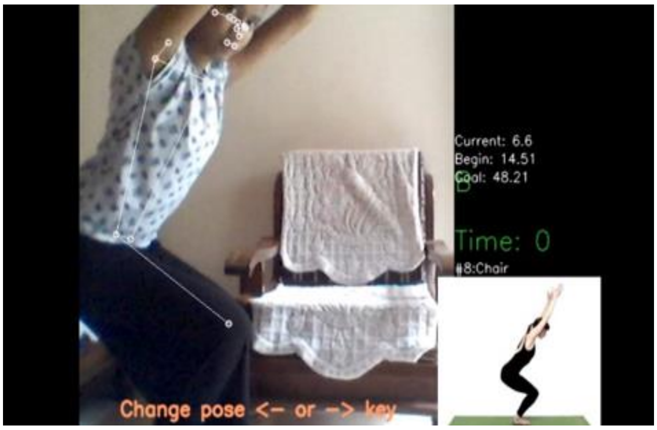

Fig 9: Yoga Pose 4

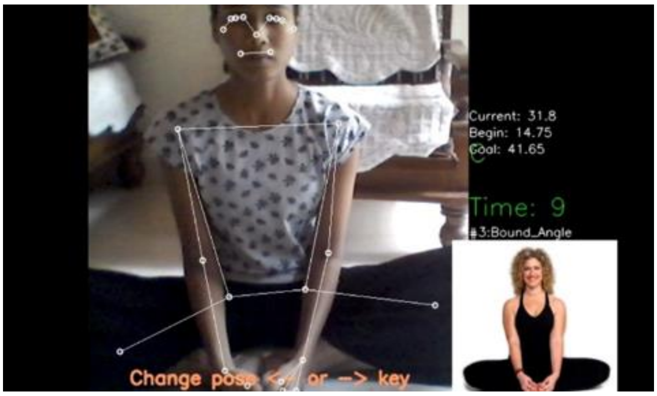

Fig 9: Yoga Pose 5

XII. CONCLUSION

The time-distributed CNN layer discovers patterns between key points in one frame and also the SVM reviews in the latest frames for the memory of previous frames, the results build the system even a lot of strong by minimizing the error because of false keypoint detection. Because the frames of Yoga pictures are sequential. We tend to plan a Yoga identification system using a conventional RGB camera. The dataset is collected using an HD 1080p Logitech digital camera for fifteen people (ten males and 5 females) and created publicly available. The machine learning-based framework eliminates the options giving the addition of the latest asanas by simply preparing the model with new data. We tend to apply the time-distributed CNN layer to detect patterns between key points during a single frame and the LSTM to study the patterns found within the recent frames. Using LSTM for the memory of previous frames and polling for denoising, the results build the system even more strong by minimizing the error because of false keypoint detection. Since the frames of a Yoga video are sequential. The same approach can be used for posture recognition in varied tasks likes, sports, healthcare, and image classification.

REFERENCES

[1]. L. Sigal. “Human pose estimation”, Ency. of Comput. Vision, Springer 2011.
[2]. S. Yadav, A. Singh, A. Gupta, and J. Raheja, “Real-time yoga recognition using deep learning”, Neural Comput. and Appl., May 2019. [Online]. Available: https://doi.org/10.1007/s00521-019-04232-7
[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 Patter 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”, IEEE Region 10 Humanit. Tech. Conf., pp. 668-67, 2017.

[6]. S. Patil, A. Pawar, and A. Peshave, “Yoga tutor: visualization and analysis using SURF algorithm”,Proc. IEEE Control Syst. Graduate Research Colloq.,pp. 43-46, 2011.

[7]. W. Gong, X. Zhang, J. Gonza`lez, A. Sobral, T. Bouwmans, C. Tu, and H. Zahzah, “Human pose estimation from monocular images: a comprehensive survey”, Sensors, Basel, Switzerland, vol. 16, 2016.

[8]. G. Ning, P. Liu, X. Fan and C. Zhan, “A top-down approach to articulated human pose estimation and tracking”, ECCV Workshops, 2018.

[9]. A. Gupta, T. Chen, F. Chen, and D. Kimber, “Systems and methods for human body pose estimation”, U.S. patent, 7,925,081 B2, 2011.

[10]. H. Sidenbladh, M. Black, and D. Fleet, “Stochastic tracking of 3D human figures using 2D image motion”, Proc 6th European Conf. Computer Vision, 2000.

[11]. A. Agarwal and B. Triggs, “3D human pose from silhouettes by relevance vector regression”, Intl Conf. on Computer Vision Pattern

[12]. M. Li, Z. Zhou, J. Li and X. Liu, “Bottom-up pose estimation of multiple person with bounding box constraint”, 24th Intl. Conf. Pattern Recogn.,2018.

[13]. Z. Cao, T. Simon, S. Wei, and Y. Sheikh, “OpenPose: realtime multiperson 2D pose estimation using part affinity fields”, Proc. 30th IEEE Conf. Computer Vision and Pattern Recogn,2017.

[14]. A. Kendall, M. Grimes, R. Cipolla, “PoseNet: a convolutional network for real-time 6- DOF camera relocalization”, IEEE Intl. Conf. Computer Vision, 2015.

[15]. S. Kreiss, L. Bertoni, and A. Alahi, “PifPaf: composite fields for human pose estimation”, IEEE Conf. Computer Vision and Pattern Recogn, 2019.

[16]. P. Dar, “AI guardman – a machine learning application that uses pose estimation to detect shoplifters”. [Online]. Available2018

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

[18]. A. Lai, B. Reddy and B. Vlijmen, “Yog.ai: deep learning for yoga”. [Online]. Available: http://cs230.stanford.edu/projects$_{w}$*inter*$_{2}$019

[19]. M. Dantone, J. Gall, C. Leistner, “Human pose estimation using body parts dependent joint regressors”, Proc. IEEE Conf. Computer Vision Pattern Recogn., 2013.

[20]. A. Mohanty, A. Ahmed, T. Goswami, “Robust pose recognition using deep learning”, Adv. in Intelligent Syst. and Comput, Singapore, pp 93-105, 2017.

[21]. P. Szczuko, “Deep neural networks for human pose estimation from a very low resolution depth image”, Multimedia Tools and Appl, 2019.

[22]. M. Chen, M. Low, “Recurrent human pose estimation”, [Online]. Available: https://web.stanford.edu/class/cs231a/prev$_{p}$*rojects*$_{2}$016/*final*

[23]. K. Pothanaicker, “Human action recognition using CNN and LSTMRNN with attention model”, Intl Journal of Innovative Tech. and Exploring Eng, 2019.

[24]. N. Nordsborg, H. Espinosa, “Estimating energy expenditure during front crawl swimming using accelerometrics”, Procedia Eng., 2014.

[25]. P. Pai, L. Changliao, K. Lin, “Analyzing basketball games by support vector machines with decision tree model”, Neural Comput. Appl., 2017. [26] S. Patil, A. Pawar, A. Peshave, “Yoga tutor: visualization and analysis using SURF algorithm”, Proc. IEEE Control Syst. Grad. Research Colloquium, 2011.

[26]. W. Wu, W. Yin, F. Guo, “Learning and self-instruction expert system for yoga”, Proc. Intl. Work Intelligent Syst. Appl, 2010.

[27]. E. Trejo, P. Yuan, “Recognition of yoga poses through an interactive system with kinect device”, Intl. Conf. Robotics and Automation Science, 2018.