International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 08 Issue: 05 | May - 2024 SJIF Rating: 8.448 ISSN: 2582-3930

YOGA POSE VALIDATION USINGDEEP LEARNING

MS ROSHINI S.N(AP/AI&DS**)**

MR ARASU PANDIAN N, MS KRITHIKA R.S , MR KISHORE S

BACHELOR OF TECHNOLOGY -2ND YEAR

DEPARTMENT OF ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

SRI SHAKTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY(AUTONOMOUS)

COIMBATORE - 641062

ABSTRACT:

The yoga practice was developed 5000 years ago in ancient India by the Indus- Sarasvati culture. The term yoga describes a deep union of the body and the psyche. It is used to maintain body and mental balance through all of life's ups and downs through asana, meditation, and other disciplines. Yoga has recently gained worldwide interest in the modern way of life due to increased stress levels, and there are numerous resources and methods available for studying it. Yoga can be performed in studios, with private instructors, and independently with the use of books, videos, the Internet, and other resources. Many people prefer self-learning because of their hectic schedules and the likelihood that the aforementioned materials won't always be available. However, with self-learning, a bad posture might not occur. Poor posture can have a negative impact onone's health, causing both short-term, acute pain and long-term, chronic issues. This research develops deep learning-based methods to identify bad yoga posture. Users can choose their preferred yoga pose by using this strategy. In order to train models that produce the anomalous angles found between the user's pose and the actual position, the user's pose is given. By highlighting the areas where the yoga stance is incorrect, the system uses these outputs to give the user advice on how to correct it.

1. INTRODUCTION:

Yoga has been incredibly popular all over the world in recent years because of all of its many mental, emotional, and physical advantages. As yoga continues to develop, people are realizing how important it is to maintain good alignment and posture when performing yoga poses. Proper alignment during yoga practice not only increases the power of the poses but also reduces the chance of injury. Accurate alignment can be difficult to achieve, though, particularly forinexperienced practitioners or thoseworking without an instructor.

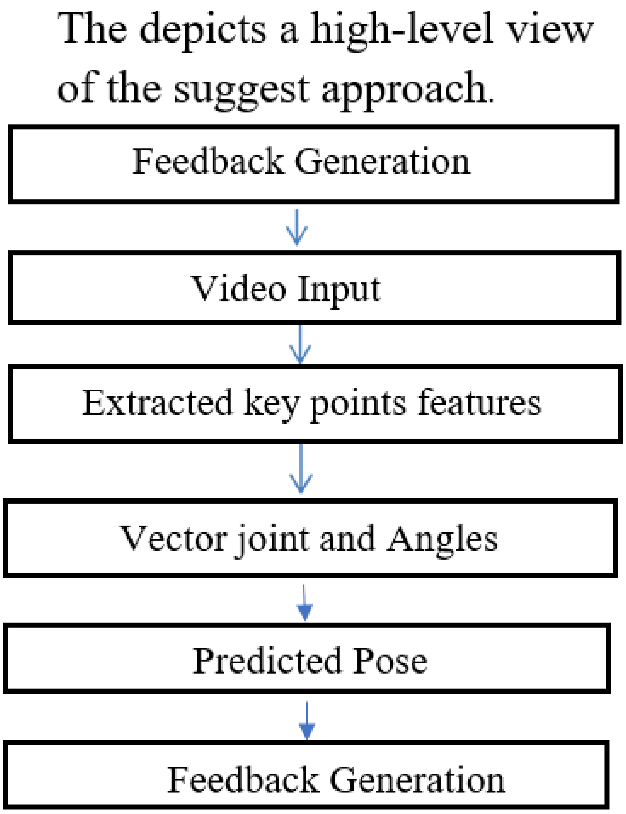
Technological developments in deep learning present a viable way to address this issue by creating tools that can help yoga practitioners refine their poses. Deep learning approaches are being incorporated into the field of yoga posture validation, which creates chances for automated systems to analyze and evaluate the accuracy of yoga poses in real time.

With the use of deep learning, yoga position validation uses artificial intelligence to give practitioners immediate feedback on the alignment of their poses. Key anatomical markers in yoga positions may be detected and evaluated using this novel approach, which makes use of computer vision and machine learning algorithms. These landmarks may consist ofthe body's general equilibrium, joint angles,and limb positions.

The main goal of this research is to create a reliable and precise system that can precisely evaluate yoga positions.

A system like this may work as a virtual yoga instructor, providing on-the-spot coaching and corrections, improving the quality of the practice overall, and encouraging safer and more productive yoga sessions. Furthermore, this technology might be included into internet resources, smartphone apps, or even smart yoga mats, enabling a diverse variety of practitioners toutilize it regardless of their expertise level or location.

The goal of this research project is to close the gap between traditional yoga instruction and state-of-the-art technology by exploring the fields of computer vision, deep learning, and biomechanics. We hope to transform how people approach their yoga practice, creating a deeper understanding of proper alignment and ultimately improving the general well- being of yoga enthusiasts worldwide by fusing the age-old wisdom of yoga with the cutting-edge capabilities of deep learning.



2. RELATED WORKS:

1. Yoga Pose Estimation and CorrectionviaDeep Learning:

Prior research has investigated the automatic recognition and correction of yoga poses through the use of deep learning techniques, such as convolutional neural networks (CNNs) and pose estimation models. These methods concentrate on identifying important joint points and angles and provide practitioners with immediate alignment feedback.

2. Human Pose Estimation in ExerciseVideos:

Exercise films, particularly yoga postures, have been the subject of research in the field of human position estimation. Accurate tracking and analysis of body keypoints has been achieved through the application of several deep learningarchitectures and techniques, supporting

Alignment enhancement and poseassessment.

3. Biomechanics-Informed PoseValidation:

Some works build more precise pose validation systems by integrating biomechanical concepts and restrictions into deep learning models. Through the consideration of joint dynamics and anatomical constraints, these methods offermore accurate input regarding posturecorrectness.

4. Multi-Modal Data Fusion for PoseEvaluation:

Researchers have investigated the integration of data from several sources, including depth sensors, RGB cameras, and wearable devices, to improve the durability and accuracy of pose validation. The model's capacity to collect fine-grained pose details is enhanced by the integration of many modalities.

5. Pose Similarity and Alignment Metrics:

Determining relevant similarity metrics between user-generated postures and reference poses is the focus of several works. Pose embeddings that quantify alignment discrepancies are learned through deep learning algorithms, allowingpractitioners to receive tailored feedback.

6. Yoga Pose Classification and Feedback:

Deep learning algorithms have been used to categorize various yoga poses and give practitioners individualised feedback. These technologies assist users in distinguishing between different poses, comprehending how they are executed, andadjusting as needed to ensure correctalignment.

7. User-Centric Pose Analysis:

Some studies stress how crucial it is to take unique user attributes—like flexibility and body proportions—into account when validating poses. Based on these particular characteristics, deep learning models can adjust and offer customized feedback.

8. Real-time Feedback Systems forYoga:

Deep learning-based real-time pose validation systems have been incorporated into interactive platforms like virtual reality settings, smart mirrors, and smartphone apps. During yoga sessions, these tools allow for instant feedback and correction.

9. Transfer Learning for Yoga Pose Recognition:

In order to adapt deep learning models trained on general human posture datasets to the domain of yoga poses, transfer learning approaches have been investigated. Pose validation models that are precise and efficient can be developed more quicklywith this method.

10. Dataset Creation and Annotation for Yoga Poses:

A vast variety of yoga postures and variants are included in datasets that have been specially selected and annotated by researchers for the purpose of pose analysis. These datasets make it easier to build and assess deep learning models for validating yoga poses.Together, these related works advance the field of yoga pose validation using deep learning by providing methods, insights, and techniques that can direct the creation of a precise and intuitive system forimproving alignment and real-time pose assessment in yoga practice.

3. METHODOLOGY:

This work proposes a deep learning- based methodology for estimating yoga postures, as described in algorithm 1, to identify proper positions and offer feedback to enhance the posture. Using deep learning to validate yoga positions may be a fun and difficult undertaking. Here is a high-level approach that you could use deep learning to validate yoga poses:

1. Data Collection and Preparation:

Assemble a varied collection of pictures or movies showing different yoga positions. The collection ought to comprise people with various bodykinds and skill levels, as well as backgrounds and lighting situations.

Add ground-truth labels to the dataset that show the appropriate posture for every picture or video frame. To ensure precision, this can be a labor-intensive process that calls for seasoned yoga practitioners.

2. Pose Estimation:

To extract important points or joint locations from the pictures or frames, use a posture estimation model that has already been trained. OpenPose, PoseNet, andHRNet are a few of the well-liked poseestimation methods.

In order to feed the key point locations into your deep learning model, convert them into a common format. This could entail encoding joint angles or normalizing coordinates.

3. Model Architecture:

Create a deep learning model that can anticipate the matching yoga position label by using the standardized pose data as input. Convolutional neural networks (CNNs), recurrent neural networks (RNNs), and even more sophisticated models like transformers are examples of designs that can be used.

To determine which model is ideal for your task, experiment with various architectures, layer combinations, and hyperparameters.

4. Data Augmentation:

Use data augmentation methods to make your training dataset more diverse. Various methods such as rotation, scaling, flipping, and noise addition can enhance thegeneralization of the model.

5. Training:

Divide your dataset into sets for testing, validation, and training. Your deep learning model should be trained using the training set, and its performance should bemonitored and overfitting avoided using thevalidation set.

Use a suitable loss function, like categoricalcross-entropy, to train the model for your multi-class classification task.

6. Evaluation:

To determine your trained model's performance precisely, use the test set to evaluate it. Measures like as recall, accuracy, precision, and F1-score can shed light on how well the model is categorizingyoga postures.

7. Fine-tuning and Optimization:

Examine the model's performance and think about adjusting its hyperparameters or architecture to increase generalization or accuracy.

Investigate methods such as transfer learning, which begin with a model that has already been trained and refine it for your particular task.

8. Deployment:

After you're happy with the model's performance, use it in an actual situation. This can entail incorporating it into a website, smartphone app, or other platforms where users can upload videos or pictures of their yoga poses for approval.

9. Continuous Improvement:

Gather data and user input over time to hone and enhance the model's functionality. This article's estimation allows you to retrain the model on a regularbasis. updated information to reflect changes and variances in yoga poses.

Recall that creating an effective system for validating yoga poses involves a thorough comprehension of both deep learning and yoga. Working together with yoga instructors, practitioners, and subject specialists can improve your system's accuracy and efficacy significantly.

3.1. Collections of Data. The suggested methodology is tested using an online, open-source dataset that is accessible to the public [30]. Six yoga positions are included in this dataset: Triangle (Trik), Lotus (Padam), Mountain (Tada), Cobra (Bhuj), Tree (Vriksh), and Corpse (Shav). There are 70 films overall of the six poses, and 350 instances total of the six poses combined. The camera is used to record these films in a room at a distance of 4 meters, with a 30 frames per second frame rate. In order to create strong trained models, participants executed these positions with minimal deviations. The dataset's statistics include the number of videos, the time in seconds for each exercise class, and the number of participants for each yoga pose individually.

3.2. Multiperson Pose Estimation in Real Time. Generally speaking, pose estimation Because they are not important features, the nose, ears, and eyes' features are not included when determining body points. Furthermore, in order to properly identify joints, characteristics with confidence scores less than 0.3 are not considered. As a consequence, 13 vectors are visible. There are 12 joints in the feature set overall, but none of them have an ear, eye, or nose.Algorithms identify body spots, connect those points, and produce their essential points. The 12 joints are the neck to the right shoulder, the right shoulder to the right elbow, the right elbow to the right wrist, and the neck to the left shoulder. The posture estimation extracts 18 body important points, and each point is composed of a body point's x and y coordinates.When several values are discovered for a key in the dictionary, all of this data, together with the corresponding confidence levels, are shown in an array. Keys are bodily components in the dictionary, and values are their coordinates. With differing degrees of certainty, two body locations are identified for the right wrist in the Bhuj position.

3.3. Feature Extraction. Important points for pose estimation are extracted using Keras real-time multiperson pose estimation [7, 8].While identifying poses for a person, a number of significant points are being found with differing degrees of confidence. Because of the way Keras posture estimation works, it can incorporate the first significant point that is found without accounting for confidence intervals. A few changes were made to the Keras pose to account for significant regions of highest confidence levels.

The left hip to the left knee, the right knee to the right ankle, the neck to the left hip, the left hip to the left knee, and the left shoulder to the left elbow, left elbow to the left wrist. For models to be trained to achieve high accuracy, key points must be scaled in several positions at different distances from the camera.

On the other hand, no feature needs to be scaled because of distance variations when angles are used as features. For example, the angle formed by the joint where the left ankle and left knee touch the ground at a 90-degree angle remains constant regardless of the distance from the camera. As stated by [29], the hip angles won't change if the knees, shoulders, and hips all rotate at the same angle. If two key points are swapped, the angle formed in respect to the reference point in reference [28] stays the same. As a result, whereas the angles employed in the system are rotation- variant, the angles used in references [28, 29] are rotation-invariant.

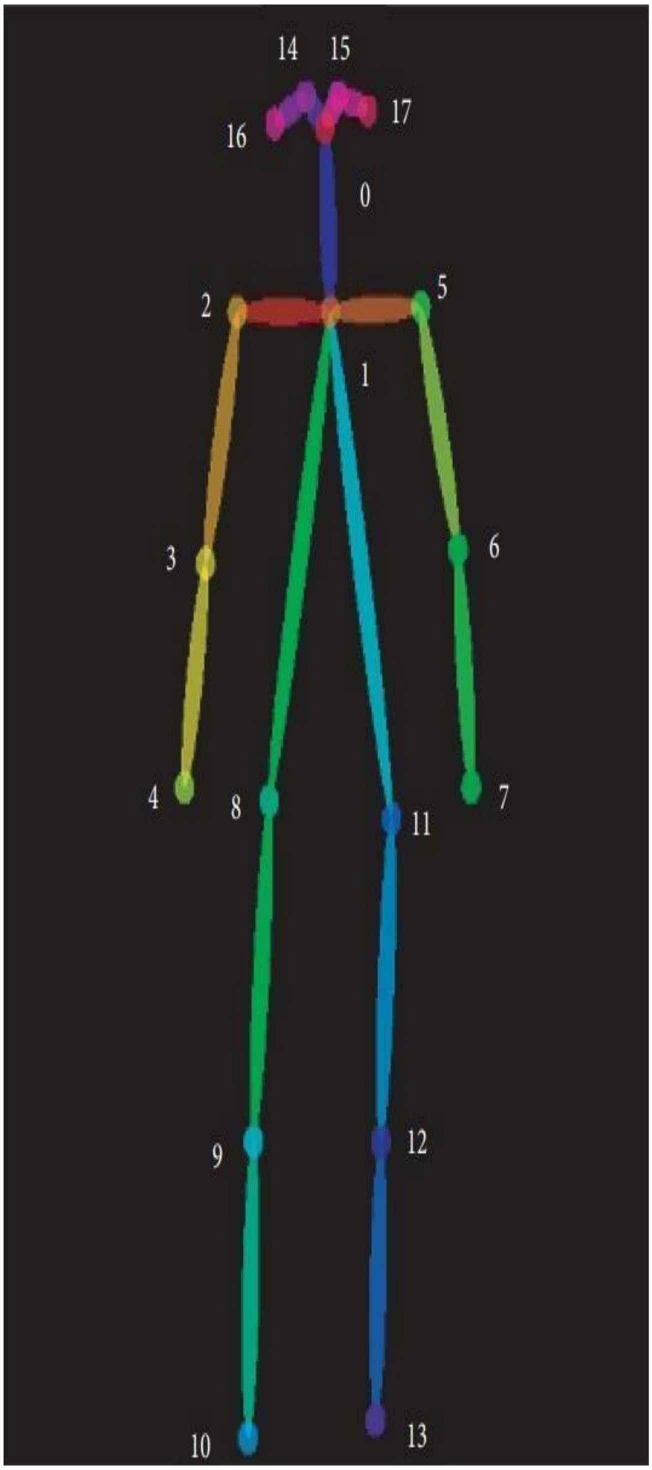
3.4. Generation of Feedback. Average values or angles are calculated for each posture in the dataset by accounting for all postures that are performed by all participants. The calculated average values are compared with the angles that were retrieved from the image. Each of the 12 values, or disparities between these angles, is calculated. The significant disparities between these 12 different values demonstrate how much one length is 12 and has 6 classes, meaning the output layer length is 6. There are 350 instances in total; 320 of them are utilized for training, and 5 instances are used for validation for each position. Ten thousand epochs are employed in training, with a batch size of twenty. The accuracy and loss graphs for the training and validation datasets are displayed in the figure. The accuracy of the training and validation datasets had numerous ups and downs until the 6900 epoch, when they finally reached an accuracy of 0.9958. Training and validation loss gradually dropped from 6900 to 10000 epochs, allowing the training model to classify with high confidence. The loss of training and validation datasets significantly decreased from epoch 0 to 10000.Since the study is categorizing input features into one of the six labels, categorical cross-entropy is the loss function that is employed. The researchers employed the sign of positive or negative to signify whether to rotate the joints when using an Ada Delta optimizer to determine whether the adaptable must modify his posture either counterclockwise or clockwise. In this way, suggestions are given to users for every joint.

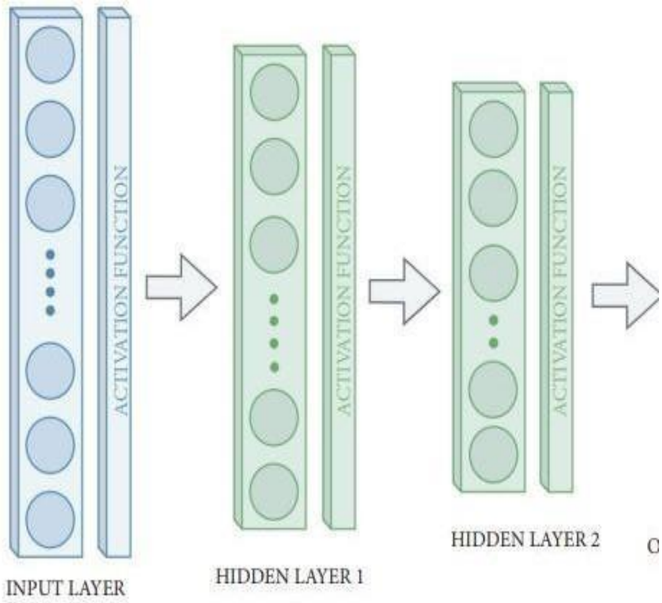
4. RESULTS:

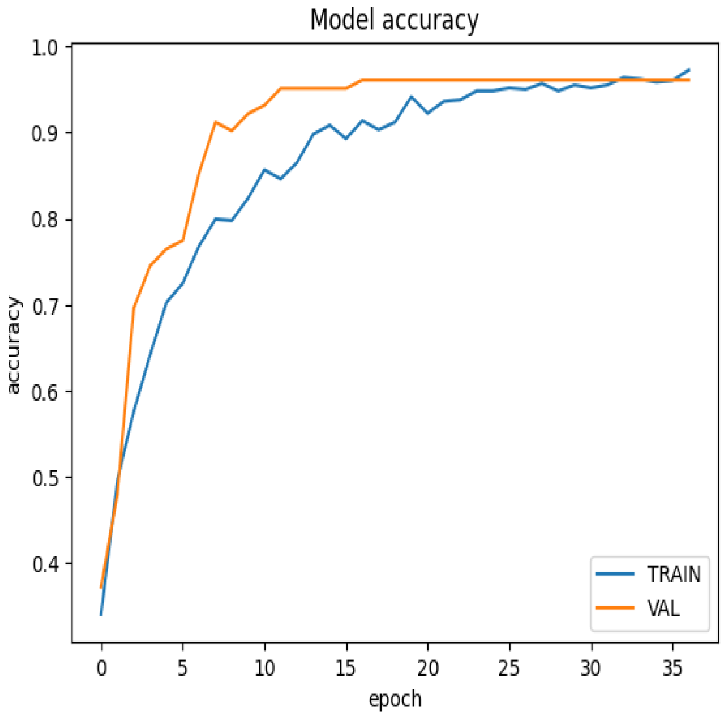
Three different kinds of layers are used to construct neural networks (MLP). Input layer, hidden layer, and output layer are their names. The number of hidden layers is determined by how complicated the training set is. The model may underfit training data if there are few hidden layers, and overfit training data if there are many. Every node in an MLP is connected to every other node in successive layers, making it a type of connected neural network. These are employed in supervised training scenarios where an output label exists for each input data set.

The classification of human stance is done using multilayer perception (MLP). Angles between important locations have been calculated in this case and supplied to the MLP as input.

The learning rate of the supplied data to overcome two shortcomings. Softmax is the activation function that was utilized for the last layer because it produces confidence levels for each label. The anticipated label is the one with the highest level of confidence. CNN obtained results of 0.9858, SVM obtained results of 0.9319, and CNN + LSTM obtained results of 0.9938 for accuracy. Despite using updated features, the system's MLP power is significantly less than that of CNN and CNN + LSTM, yet it still achieved an accuracy of 0.9958. A confusion matrix, which describes the accuracy of the classification model performance for all cases as per equation, is used to assess the efficacy of the methodology. Classification score, which is the ratio of correct classifications to total cases, is another name for classification accuracy. The result evaluation has a 6 × 6 confusion matrix since the study's confusion matrix includes 6 labels. The total number of instances in the confusion matrix for the training, validation, and training datasets is 30, 30, and 320, respectively. It is able to see thatevery instance.







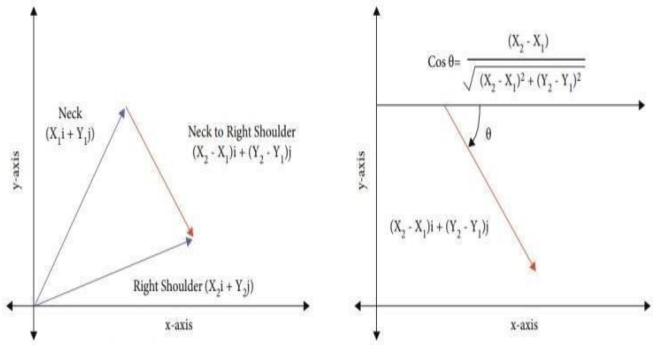


Figure 7: The coordinate vector and angle calculation representation of neck and right shoulder vectors.

RUNTIME ANALYSIS:

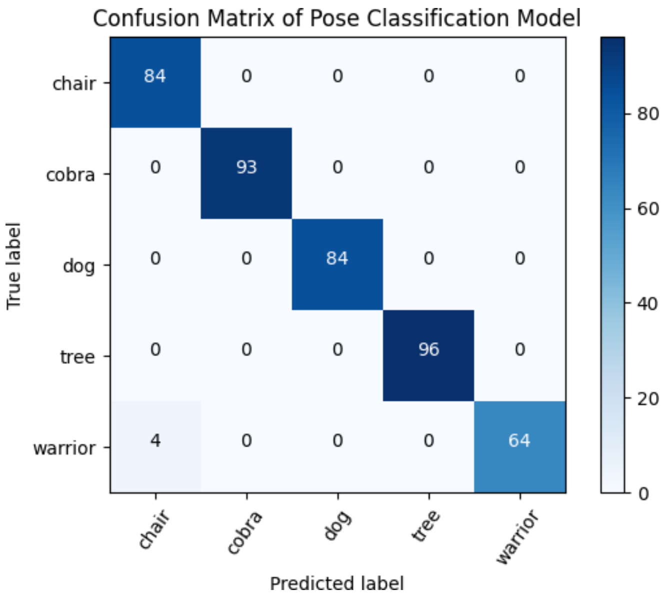
The methods described in this study use deep learning to identify bad yoga posture and provide advice on how to repair it. This study computes vectors for every joint and classifies the angle between the vectors for neighboring joints as a feature. Key points are extracted using a pose estimationtechnique. Subsequently, thesecharacteristics were incorporated into the classification algorithms, which subsequently produced feedback regarding the accuracy of the yoga stance. For every procedure, the runtime for feature extraction and computation stays the same. average runtime per frame and the experimental methodology's standard deviation. Milliseconds are used to display the time. It includes the amount of time needed for feature extraction and categorization with each frame and feedback development.

```
  14/14 [==============================]
- 0s 1ms/step
  Confusion matrix, without
normalization

  Classification Report:
              precision    recall  f1-
score   support

      chair       0.95      1.00
0.98      84
      cobra       1.00      1.00
1.00      93
        dog       1.00      1.00
1.00      84
       tree       1.00      1.00
1.00      96
    warrior       1.00      0.94
0.97      68

   accuracy
0.99     425
  macro avg       0.99      0.99
0.99     425
weighted avg      0.99      0.99
0.99     425
```



5. CONCLUSION:

To sum up, human posture validation is essential to guaranteeing the precision, dependability, and efficiency of different systems and applications that use human movement analysis. Human position validation uses cutting-edge technology like computer vision, machine learning, and deep learning to properly track and evaluate people's complex movements and postures.

The pose estimation algorithms' limitations, flaws, and errors can be found with the use of this validation procedure, which also yields insightful information that can beused to improve and refine the algorithms.

Furthermore, human pose validation encourages the creation of novel technologies that make society safer, healthier, and more productive. It makes it possible for academics, professionals, and developers to produce apps that enable people to improve their general well-being, avoid injuries, and maximize their movements. The field of human posturevalidation will continue to be essential in guaranteeing the greatest levels of precision

and accuracy in human movement analysisas long as technology keeps developing.

REFERENCES:

1. Yoga tutor visualisation and analysis usingSURF algorithm," Proceedings of the 2011 IEEE Control and System GraduateConference, S. Patil, A. Pawar,
A. Peshave, A.
N. Ansari, and A. Navada Research Colloquium, IEEE, Shah Alam, Malaysia, pp.43–46 June 2011

2. "Learning and self-instruction expertsystem for yoga," by W. Wu, W. Yin, and
F. Guo, appeared in the proceedings of the 20102nd International Workshop on Intelligent Systems and Applications, published by IEEEin Mumbai, India, in May 2010.

3. Robust pose detection using deep learning,
A. Mohanty, A. Ahmed, T. Goswami, A.Das,
P. Vaishnavi, and R. R. Sahay, Proceedings of the International Conference on ComputerVision and ImageProcessing, Springer, Singapore, December 2017, pp. 93-105

4. Deeppose: human position estimate with deep neural networks,
A. Toshev and C. Szegedy, Proceedings of the IEEE conference on computer vision and pattern recognition, Columbus, OH, USA, August 2014, pp. 1653–1660 Posenet is a convolutional network for real-time 6-dof camera relocalization, as described by A. Kendall, M. Grimes, and R. Cipolla in Proceedings of the IEEE international conference on computer vision, Santiago, Chile,December 2015, pp. 2938-2946.

5. Real-time yoga recognition using deep learning, S. K. Yadav, A. Singh, A. Gupta, and J. L. Raheja, Neural Computing & Applications,vol. 31, no. 12, pp. 9349-9361, 201