Grenze International Journal of Engineering and Technology, Jan Issue

Detection and Estimation of Yoga Pose using Artificial Intelligence

Ranjana Battur[1], Pavan Kunchur[2], Sadhana Bangarashetti[3], Manjunath Managuli[4], and Gururaj L Kulkarni[5]
[1]Assistant Professor, Department of Computer Science and Engineering, KLS, Gogte Institute of Technology(Autonomous), Visvesaraya Technological University, Belagavi
[2]Associate Professor, Department of Computer Science and Engineering, KLS, Gogte Institute of Technology(Autonomous), Visvesaraya Technological University, Belagavi
[3]Professor, Department of Information Science and Engineering, Basaweshwar Engineering College (Autonomous), Visvesaraya Technological University, Belagavi
[4]Associate Professor, Department of Electronics and Communication Engineering, KLS, Gogte Institute of Technology(Autonomous), Visvesaraya Technological University, Belagavi
[5]Associate Professor, Department of Computer Science and Engineering, Vardhaman College of Engineering, Hyderabad

***Abstract*— Yoga offers a fresh perspective on our daily disciplines, including our physical, mental, and spiritual routines. Yoga has been around for a while. Medical experts and a large number of celebrities advise practicing yoga because of its various health benefits. A straightforward definition of yoga is to take care of your body, mind, and breath. However, depending on the COVID-19 condition, it is crucial to keep your health in check while everything takes place at home. You must work out every day. You need a home teacher to perform the exercise properly, something you cannot do in the COVID-19 environment. Lack of instruction when performing yoga poses is harmful to our health. So, in order to identify and detect the yoga posture form, we are going to offer a proposed model for pose recognition using machine learning. Our method is based on 8 yoga postures. Using the Tkinter library, we created a desktop application with a GUI. The media pipe and OpenCV library are used to preprocess the input as an image, recognize the item, and identify the human body's central organs. We use CSV files that are downloaded from Kaggle for the training and testing data. We train and test using logistic regression models. The system receives a perfect accuracy rating.**

I. INTRODUCTION

Yoga is a form of exercise with intricate postures that has been practiced for thousands of years and has its roots in India but has gained worldwide renowned for its many spiritual, physical, and mental advantages. Yoga's drawback is that, like any exercise, it must be performed correctly to be effective. Additionally, performing yoga in the wrong position can be both uncomfortable and useless. This calls for the presence of a trainer to oversee the exercise and correct the participant's posture. Since not every customer has access to a trainer, it may be possible to identify yoga poses using a computerized reasoning-based application and offer personalized feedback to assist clients correct their posture. This work aims to improve our understanding of the numerous posture classification methods, such as pose estimation, KNN classification, Posenet, and the KNN classifier model. The risk of undesirable health disorders, such as coronary heart disease, high blood pressure, stroke, metabolic syndrome, and type 2 diabetes, increases with a lack of physical fitness, which lowers people's life expectancies. In this study, position estimate for yoga pose correction is put into practice. There are numerous open source pose detection

Grenze ID: 01.GIJET.10.1.504_2

systems that need a camera and system setup. This is a fundamental yoga model that enables practitioners to perform different yoga postures and modify them by gazing at a mirrored image.

II. Literature Survey

An interesting field of research is deep learning, which enables help scale-up the analysis of massive amounts of data. Since deep learning recognizes intricate patterns in data and extracts features on its own, it does not require feature engineering or extraction like typical machine learning models. The visual appeal of structures that portray human production as a solid arrangement was emphasized in earlier work. Humans with connected skeletons between various body sections are used to study the human body. A predetermined kinematic body model is commonly used, with the assumption that each body part is independent of the others.

Recent years have seen the development of numerous extensions, including mixed, hierarchical, multimodal, and potent appearance models, such falling/sequential prediction. A layout that incorporates position, orientation, shape arrangement, and layout model is used to depict each limb. Using picture contour information and sped-up robust features (SURF), Patil et al.'s method for identifying yoga posture differences between a practitioner and an expert was proposed in their paper (Patil et al., 2004). However, it is insufficient to describe and contrast postures almost solely on the basis of contour data. Santosh Kumar Yadav et al. [5] suggested a Yoga detection approach using a regular RGB camera. They collected the information using a high-definition webcam. They scanned the user with Open Pose and identified important details. The LSTM was used to remember the patterns seen in recent frames, while the time-distributed CNN layer was utilized to identify patterns between important points in a single frame. Their approach did away with the need for Kinect or any other specialized equipment to recognize yoga postures.

An intelligent system for yoga training that uses tactors and inertial measurement units (IMUs) was introduced by Guo et al. [6]. They neglected to take the user's posture into account, which could make it uncomfortable for them and disrupt the natural yoga stance.

Another method for Kinect-based yoga pose correction [7] has been described, which takes into account the stances warrior III, downward dog, and tree pose. Their accuracy rating, however, is only 82.84 percent, which is not particularly noteworthy. The outdated skeletonization procedure has since been replaced with deep learning-based technology.

III. Methodology

The suggested system attempts to observe users as they practice yoga in order to recognize proper positions and offer visual cues for correction. An interactive experience requires real-time processing. The system takes input in the form of points data, which is then processed using machine learning methods including Gradient Boosting Classifier, Logistic Regression, Ridge Classifier, and Random Forest Classifiers for feature extraction, classification, and preprocessing. Based on the geometric properties of the yoga dataset, the system extracts feature from the dataset and applies the SVM algorithm for classification and prediction. The technology then recognizes yoga poses and makes any necessary modifications.

IV. Suggestive System

The dataset is made up of a grouping of about 400 different images for each yoga pose. These Images are optimized to have comparable pixels and sizes before being combined into a single video. The KNN classifier is used to recognize the pose by recognizing the key points of the human limbs, and the model is saved. The proposed model will be trained using the output video, which contains images of various poses. As an input source, we are employing a webcam. We can capture a user's actual yoga pose using a camera, and a KNN classifier will identify critical points and categories it into one of the yoga poses. Then, we can watch a video of a yoga instructor doing the identified yoga pose. We can see the building.

K-Nearest Neighbors Classifier is a machine learning technique that stores all cases and groups them according to commonalities.

It is mostly utilized in the process of pattern recognition. According to "feature similarity," which denotes how closely a new data point resembles existing data points in the training set, the KNN algorithm predicts the values of new data points. These steps are used to demonstrate how it works:

1. Every algorithm needs a dataset to be implemented. As a result, during the first stage of KNN, the training and test data must be loaded.
2. K's value, or the data points that are closest to it. Any integer that must be chosen as K is acceptable.

3. Execute the following for every test data point:

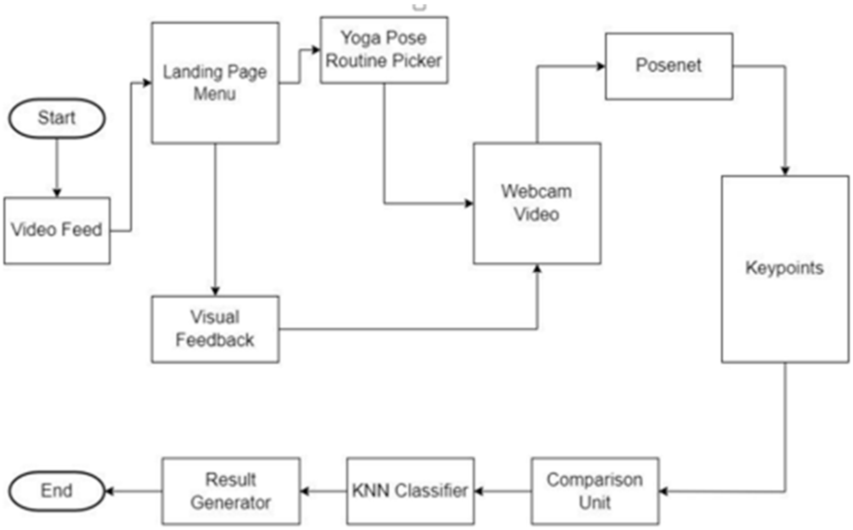

Figure 1: Architecture diagram

i. Determine the distance in terms of Euclidean, Manhattan, or Hamming angles between each row of training data and test data. The Euclidean approach is the one that is most frequently used to calculate distance.
ii. Depending on the distance, rank them ascendingly. The top K rows of the sorted array are then chosen
iii. The test point will now be given a category based on the most frequent category of these rows.

V. DATASET

This dataset was compiled using information from Kaggle and other public sources. Images of five various yoga postures.

VI. POSENET

It is a deep learning framework that uses key spots on the human body to train the model in order to recognize human poses in image and video sequences.

VII. MODEL FOR TRAINING

Python tools like TesnorFlow - Keras, OpenPose, and NumPy are used to build an optimal movie that contains every single unique yoga stance. In order to categories and train the model using a front-end user interface, the deep learning framework will now extract the essential elements from the video sequence. Accuracy in training: 93.56%

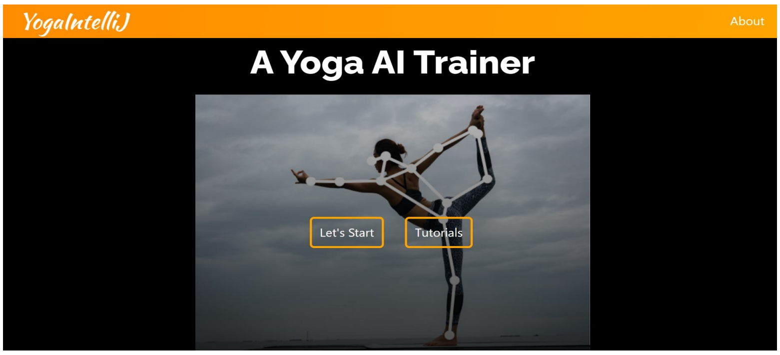

Figure 2. AI Trainer GUI

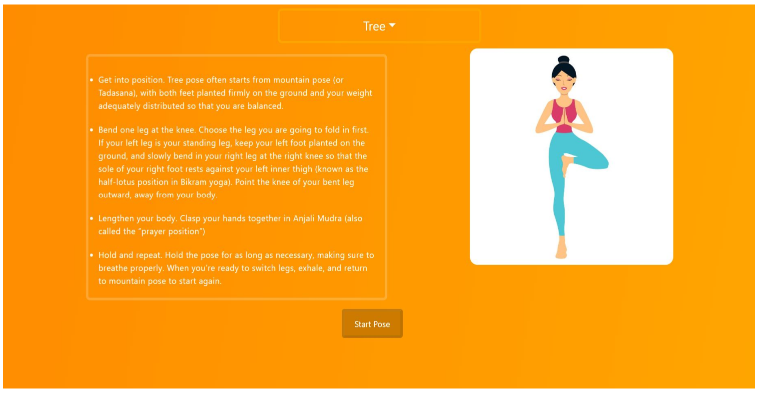

Figure 3. Example caption

The important features of the user, including as their eyes, ears, nose, and limbs, are compared to the trained model in the above image as they are executing Vrksasana (tree pose). The yoga pose that inspires the most confidence will be shown on the screen. The user will see the proper alignment and instructions for Vrksasana (tree pose) as they are doing it, helping them to complete the yoga pose.

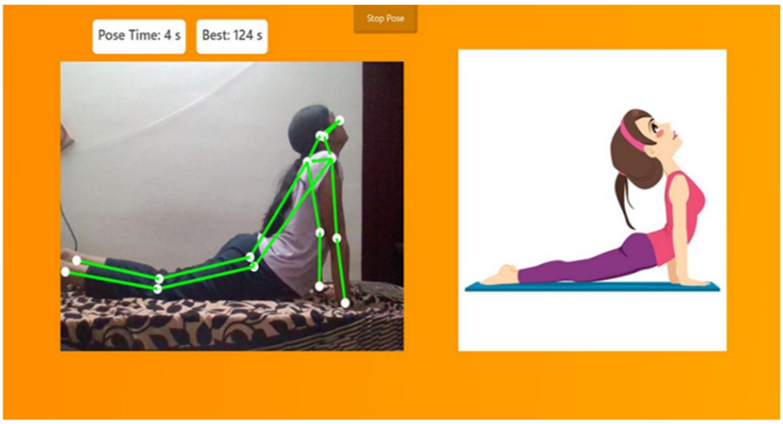

Figure 4. Different Yoga pose

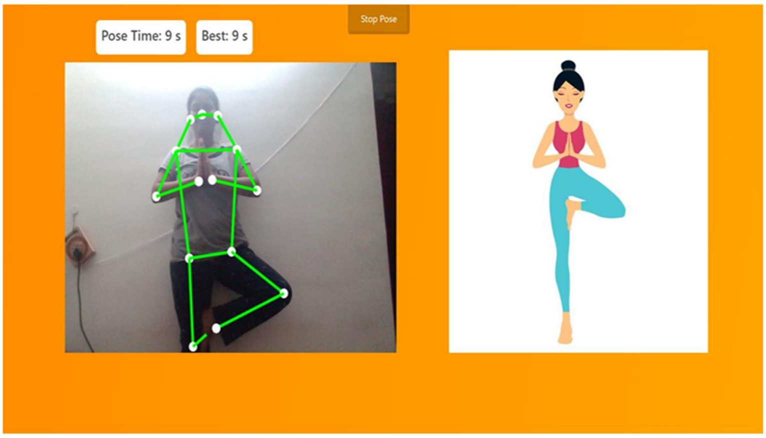

Figure 5. Different yoga pose

VIII. Proposed Model Performance

Model will identify the important elements of the user's yoga pose using a camera. According to the feed, it will categories it as one of the learned yoga poses, and we can then see a video of a yoga instructor precisely executing the recognized pose so that the user can learn and adjust their posture appropriately. With an accuracy percentage of 98.51%, the results were encouraging.

IX. CONCLUSION

A lot of study has been done on assessing human position in the recent years. In contrast to other computer vision problems, estimating human posture requires locating and assembling human body parts based on an existing human body structure. Using yoga posture estimation for fitness can help people prevent injuries and be more productive. In this system, Pose net and cutting-edge picture categorization algorithms like KNN are tested. The accuracy of 98.51% was extremely good for the outcomes. Yoga Asana' movements can be examined using video and image analysis to ensure that they are performed correctly. For video-based analysis, models like Pose net, Ml5, and KNN Classifier are suitable.

FUTURE SCOPE

The proposed paradigm currently divides yoga asana into five groups. It is difficult to create a posture estimate model that is applicable to all yoga asana because there are so many of them. The dataset could be expanded to include more yoga poses practiced by individuals in both indoor and outdoor settings. The evaluation method used by Open Pose now determines how the model is displayed, which may or may not be effective when there is overlap between people or between different body sections. This framework can be used on a mobile device for continuous forecasting and self-preparation. This work demonstrates movement acknowledgment for reasonable applications. In the future, we intend to add more challenging yoga asana that are helped by AI.

REFERENCES

[1] KNN classification Available at: www.saedsayad.com/k_nearest_neighbors.htm.

[2] Pose net - Open Pose Available at: https://www.geeksforgeeks.org/posenetposeestimation/.

[3] Available at https://www.Pyimagesearch.com/2016/08/08/k-nnclassifier-for-image-classification/. KNN classifier model.

[4] "Yoga tutor: visualisation and analysis using SURF algorithm," S. Patel, A. PA war, and A. Pushover. In: ICSGRC 2011: Proceedings of the 2011 IEEE Control System Graduate Research Colloquium.

[5] "Real-time Yoga recognition using deep learning" by Santos Kumar Yama, Amitojdeep Singh, Abhishek Gupta, and Jagdish Lal Raheja (2019).

[6] "Learning and self-instruction expert system for yoga," Proc. International Work Intelligent System Application, 2010, by W. Wu, W. Yin, and F. Guo.

[7] "Computer-assisted self-training system for sports exercise using kinects," 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), Yu-Zhen He, C. L. Chou, S. Y. Lee, B. S. P. Lin, and J. Y. Yu.

[8] H. T. Chen, Y. Z. He and C. C. Hsu, "Computer-assisted yoga training system", Multimedia Tools and Applications, vol. 77, no. 18, pp. 23969-23991, 2018.

[9] S. K. Yadav, A. Singh, A. Gupta and J. L. Raheja, "Real-time Yoga recognition using deep learning", Neural Computing and Applications, vol. 31, no. 12, pp. 9349-9361, 2019.

[10] Y. Chena, Y. Tianb and M. Hea, "Monocular Human Pose Estimation: A Survey of Deep Learning-based Methods", arXiv, 2020.

[11] D. Osokin, "Real-time 2D Multi-Person Pose Estimation on CPU: Lightweight OpenPose", CPRAM 2019 - Proceedings of the 8th International Conference on Pattern Recognition Applications and Methods, pp. 744-748, 2019.

[12] "yogajournal", Pocket Outdoor Media Inc, [online] Available: https://www.yogajournal.com/.