ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/360950640

# Yoga Pose Detection and Correction using Posenet and KNN

**Conference Paper** · May 2022

CITATIONS
3

READS
3,491

**4 authors**, including:

Varsha Bhosale
Vidyalankar Institute of Technology
**13** PUBLICATIONS **15** CITATIONS
SEE PROFILE

Abhishek Bale
Indian Statistical Institute
**2** PUBLICATIONS **3** CITATIONS
SEE PROFILE

**Some of the authors of this publication are also working on these related projects:**



Data Warehousing and Mining for Gene Expression Data View project



Detection and Classification View project

All content following this page was uploaded by Varsha Bhosale on 30 May 2022.

The user has requested enhancement of the downloaded file.

International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 09 Issue: 04 | Apr 2022 www.irjet.net p-ISSN: 2395-0072

# Yoga Pose Detection and Correction using Posenet and KNN

**Varsha Bhosale[1], Pranjal Nandeshwar[2], Abhishek Bale[3], Janmesh Sankhe[4]**

*[1]Associate Professor, [2,3,4] Under-Graduate Student*
*[1,2,3,4] Department of Information Technology, Vidyalankar Institute of Technology, Mumbai, India.*

---

**Abstract -** *The fundamental goal of Yoga pose detection and correction is to provide standard and correct yoga postures using computer vision. If the yoga posture is not done properly, it can result in serious injuries and long-term issues. Analyzing human poses to detect and correct yoga poses can benefit humans living a healthier life in their homely environment. This project focuses on exploring the different approaches for yoga pose classification, so we are using PoseNet and KNN classifier. Using such deep learning algorithms, an individual can get the correct/ideal way/method to perform that specific yoga asana that he/she is trying to do. Using computer vision techniques and Open Pose (an open-source library), human pose estimation is used to estimate an individual's Yoga posture. The suggested system recognises the difference between the actual and target positions and corrects the user with high accuracy by offering real-time visual output and necessary instructions to correct the identified pose.*

***Key Words*: KNN, Open Pose, Posenet, Pose Estimation, Crucial Angles.**

## 1. INTRODUCTION

Yoga, a centuries-old practice that is originally from India but is globally famous for its numerous spiritual, corporeal, and mental benefits, is a type of exercise with complex postures. The problem with yoga is that, like any other exercise, it is critical to practice it correctly because any incorrect position during a yoga session can be ineffectual and potentially inconvenient. This necessitates the presence of a trainer to supervise the meeting and correct the individual's stance. Since not every client approaches or has access to a trainer, a computerized reasoning-based application might be used to detect yoga poses and provide customized feedback to help people improve their structure. As a result, we've proposed an artificial intelligence-based application that can recognise yoga stances and respond with a customised response to help users improve their postures. The purpose of this study is to gain a better knowledge of the many techniques used to classify yoga poses, including pose estimation, KNN classification [1], Posenet [2], and the KNN classifier model [3]. Lack of physical fitness increases the risk of adverse health conditions including coronary heart diseases, high blood pressure, stroke, metabolic syndrome, type 2 diabetes which leads to a decrease in the life expectancy of humans. In this work, pose estimation for the correction of yoga poses is implemented. Many pose detection systems are open source which requires system setup and camera. This is a basic yoga model that allows people to execute various yoga poses and adjust their yoga postures by looking at a reflected image.

## 2. LITERATURE SURVEY

Deep learning is a promising area of research that allows us to examine large amounts of data in a scalable manner. Deep learning, unlike traditional machine learning models, does not require feature extraction or engineering since it understands complex patterns in data and extracts features on its own. Earlier work emphasized on the visual look of structures, which depict nature of human creation as a solid arrangement. Humans with skeletons connecting their bodily parts are employed to acquire knowledge into the human body. To assume that each body part is independent of the others, a preset kinematic body model is typically utilized. Extensive extensions have lately been created, such as mixed, hierarchical, multimodal, and powerful appearance models, such as falling/sequential prediction. Each limb is represented by a layout that includes position, orientation, shape arrangement, and layout model. Patil et al. [4] suggested a technique for determining yoga posture discrepancies between an expert and a practitioner using picture contour information and speeded up robust features (SURF). However, characterizing and comparing postures nearly entirely on the basis of contour data is insufficient.

A Yoga identification method utilizing a typical RGB camera was proposed by Santosh Kumar Yadav et al. [5]. They used a high-definition webcam to acquire the data. Using OpenPose, they scanned the user and discovered key points. The time-distributed CNN layer was used to recognize patterns between key points in a single frame, and the LSTM was used to memorize the patterns discovered in recent frames. Their method eliminated the requirement for Kinect or any other specific technology to identify Yoga postures.

Guo et al. [6] presented a picture and text-based intelligent systems for yoga training that includes inertial measuring units (IMUs) and tactors. However, this could be inconvenient for the user, as well as disrupting the natural yoga stance, and they failed to consider the practitioner's posture.

[7] has described another technique for Kinect-based yoga pose correction that takes into account three yoga poses: warrior III, downward dog, and tree pose. However, their accuracy score is only 82.84 percent, which is not very impressive. Deep learning-based technologies have now supplanted the old skeletonization process.

Shruti Kothari [8] described a computational method for classifying Yoga poses from photos that uses deep learning, specifically CNN. For the classification model, they used a dataset of 1000 photos divided into six classes. This work was completed with nearly 85% correctness.

Chaudhari et al. [9] built a system that provides clear feedback to the practitioner so that they can properly practise yoga postures using domain knowledge of five yoga positions. A CNN model was used to identify yoga positions, and a human-joint localization model was used to find defects in the pose.

## 3. PROPOSED SYSTEM

The dataset, which consists of a collection of nearly 400 distinct Images of each Yoga Pose. These Images are optimised to similar pixels and size, then blended to a single video. The Proposed Model will get trained using the output video which consist images of different poses where KNN classifier is used to detect the pose by detecting the key-points of the limbs of human and the model is saved. We are using webcam as input feed. Using webcam, we can get real-time Yoga Pose of the user and KNN classifier will identify significant points and will classify it into one of the Yoga Pose then we can see an instructor video performing the detected Yoga Pose. We can see the architecture diagram below

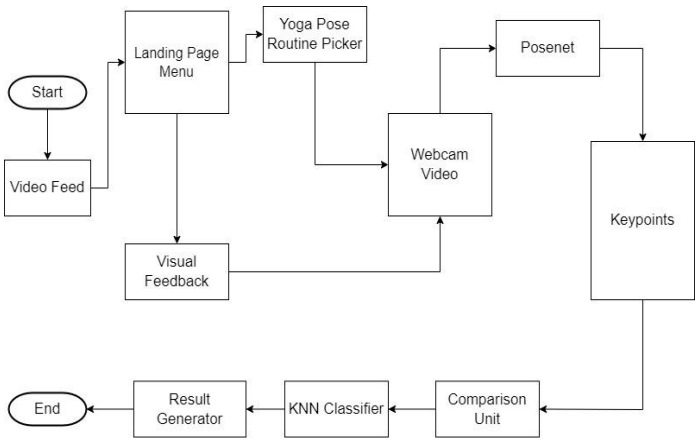


**Figure 1**: Architecture diagram

### 3.1 KNN Classifier

K-Nearest Neighbors is a Machine Learning algorithm that stores all different cases and classifies based on similarities. It is mostly used in Pattern Recognition technique. The KNN algorithm predicts the values of new data points based on 'feature similarity,' which implies that the new data point will be assigned a value depending on how closely it matches the points in the training set. Its operation is demonstrated using the procedures below:

1. A dataset is required for the implementation of any algorithm. As a result, the training and test data must be loaded during the first step of KNN.

2. The value of K, i.e. the data points closest to it. K can be any integer that is required to be selected.

3. Do the following for each point in the test data:

   i) Calculate the distance between each row of training data and the test data using one of the following methods: Euclidean, Manhattan, or Hamming distance. The Euclidean method is the most widely used method for calculating distance.

   ii) Sort them in ascending order depending on the distance value.

   iii) It then selects the top K rows from the sorted array.

   iv) The most frequent category of these rows will now be used to assign a category to the test point.

### 3.2 Dataset

Dataset used for this are collected from Kaggle [11] and other open sources. It consists of images of 5 different yoga poses namely - Utkata Konasana (goddess), Tadasana (mountain), Phalakasana (plank), Vrksasana (tree), Virabhadrasana III (warrior 3). A total no. of 1578 Images are collected.

### 3.3 POSENET

It is deep learning framework used to identify human poses in images and video sequences by identifying the critical points in human body and Training the Model based on these key points.

### 3.4 Training Model

An optimised video with all distinct yoga poses is created using python libraries such as TesnorFlow - Keras, OpenPose, NumPy. Now, The Deep learning framework will extract the key points from video sequence so that we can classify and train the model using a Frontend UI. Training Accuracy: 93.56%

### 3.5 Model Performance

Using webcam, Model will extract the key points from the Yoga Pose performed by user. Depending upon the feed it will classify it to one of the trained Yoga Poses and then we can see a video of a Yoga Instructor performing the detected pose accurately so that user can learn and correct the posture accordingly. The results were promising with an accuracy rate of 98.51%.

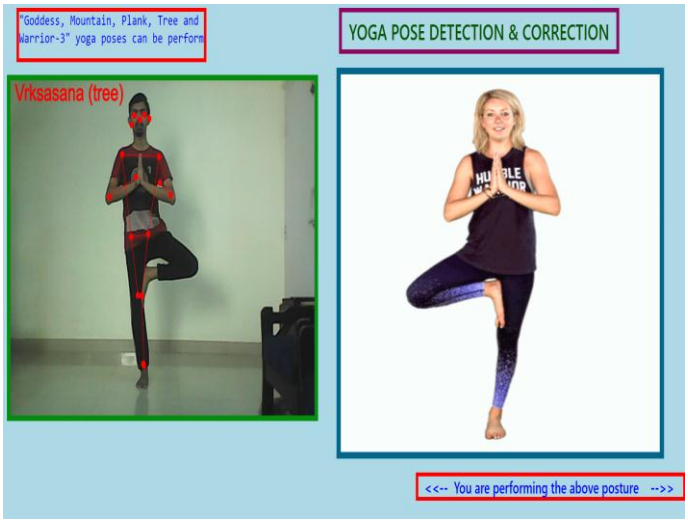


**Figure 2**: Vrksasana (tree) Pose

In the above picture, as the user is performing Vrksasana (tree), the keypoints of the user like eyes, ears, nose, limbs are compared with the trained model. The yoga asana which will have the highest confidence will be displayed on the screen. As the user is performing Vrksasana (tree), its correct posture alongwith the steps will be displayed so that it could assist the user in performing the yoga asana.


**Figure 3**: Tadasana (Mountain) Pose

In the above picture we can see that user is performing Tadasana(Mountain) Pose, the keypoints of limbs are detected and KNN classfier trained model will classify it into one of the saved poses, as we can see an instructor is performing the similar pose so that user can follow the steps and correct his/her posture.


**Figure 4**: Phalakasana (Plank) Pose

The user is performing Phalaskasana(Plank) Pose, On the right side we can see an instructor video performing the same pose is displayed.

## 4. CONCLUSION

Over the last few years, a lot of research has gone into estimating human position. Human posture estimate differs from other computer vision challenges in that it must locate and assemble human body parts based on an existing human body structure. The use of yoga posture estimation in fitness can help people avoid injuries and increase their effectiveness. State-of-the-art picture categorization algorithms like KNN are put to the test in this system along with Posenet. The results were quite impressive with an accuracy of 98.51%.The movement of the yoga Asanas can be analysed using video and image analysis to check their correctness. Model designs such as Posenet, Ml5, and KNN Classifier are appropriate for video-based analysis.

## 5. FUTURE WORK

Currently, the proposed model classifies yoga asanas into 5 categories. As there are so many yoga asanas, developing a posture estimate model that works for all of them is a tedious task. More yoga positions performed by individuals in both indoor and outdoor settings could be added to the dataset. The display of the model is determined by the nature of Open Pose's current evaluation, which may or may not work well in situations when there is overlap between persons or across body parts. This framework can be implemented on a mobile device for self-preparation and constant predictions. For plausible applications, this work demonstrates movement acknowledgment. We hope to include more AI-assisted complex yoga asanas in the future.

## 6. REFERENCES

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

[2] Posenet - OpenPose [online], Available: https://www.geeksforgeeks.org/posenetpose-estimation/

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

[4] S. Patil, A. Pawar, A. Peshave, "Yoga tutor: visualization and analysis using SURF algorithm". In: Proceedings of 2011 IEEE control system graduate research colloquium, ICSGRC 2011.

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

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

[7] H. -T. Chen, Yu-Zhen He, C. -L. Chou, S. -Y. Lee, B. -S. P. Lin and J. -Y. Yu, "Computer-assisted self-training system for sports exercise using kinects," 2013 IEEE International Conference on Multimedia and Expo Workshops (ICMEW), 2013, pp. 1-4, doi: 10.1109/ICMEW.2013.6618307.

[8] Shruti Kothari 2020 Yoga Pose Classification Using Deep Learning Ph.D. thesis SAN JOSE STATE UNIVERSITY.

[9] A. Chaudhari, O. Dalvi, O. Ramade and D. Ambawade, "Yog-Guru: Real-Time Yoga Pose Correction System Using Deep Learning Methods," 2021 International Conference on Communication information and Computing Technology (ICCICT), 2021, pp. 1-6, doi: 10.1109/ICCICT50803.2021.9509937.

[10] https://janmesh18.github.io/Yoga.github.io/ - Visit this website for reference and Project implementation



View publication stats