ZyoFiSik - Your Personal AI Trainer

Angel Francis[#1], Bairavi Balaguru[#2], Jey Sneha Arul Joseph[#3],Dr. L. Jaba Sheela[#4]

[#]*Department of Computer Science and Engineering,*
Panimalar Engineering College, Chennai, india

[1]angelfrancis1111@gmail.com

[2]tbalaguru61@gmail.com

[3]jeysneha1@gmail.com

[4]peccse2@gmail.com

***Abstract*—While ensuring proper practice, Yoga learning and Self-instruction systems can popularize and disseminate Yoga. These computerized Self-training systems for sports and fitness can improve participant performance and thwart injuries. Concocting an interactive web application that uses the webcam to recognize the user's yoga and exercise poses, and estimates each pose to assist the user practicing those postures and tracks successful shots at various stances. Our approach seeks to recognize the yoga asanas based on the data attained by Data Collection from an open-source dataset. The detected critical points are passed to our model where neural networks find patterns and Sequential model-CNN analyze their evolution over time. Mediapipe framework is used in detecting the landmarks of the human body to retrieve the stick figure of the user for estimating their yoga poses and predict its accuracy by passing it to the CNN model. In contrast , in the curl counter, the angles made by the lines of the stick figure are used to calculate the degree or the angle made between the arm and the forearm which indeed helps in assisting the user as he or she can alter his or her posture accordingly. Finally, the system contains a meditation coach which is programmed to give commands at standard intervals and subsequently, the user is ushered by a voice that instructs them to maintain their breathing pace thereby creating a soothing environment.**

Keywords—CNN, Yoga assistant, Mediapipe, Artificial intelligence

I. Introduction (*Heading 1*)

The main goal of our project is to develop an interactive web application that uses a webcam to recognize the user's yoga and exercise poses and estimates each pose to assist the user practicing those postures to infer whether or not practiced effectively.

In the Yoga Assistant module, the CNN algorithm is deployed by using the Keras- Sequential Model that uses categorical cross entropy to calculate the loss function and adam optimizer to reduce the loss.

The Exercise Trainer module takes the frame from a real-time webcam stream at specific intervals and gets all the keypoints using pose estimation algorithms from the BlazePose Model in MediaPipe.

In the Meditation Module the user specifies the duration of the meditation.Consequently, when the client seeks the server every half a second and checks for a change in the file name to switch its instruction, the server provides the audio file depending on the command at standard intervals such as breathe in, breathe out.

II. Literature Survey

Yash Agrawal, et al in “Implementation of Machine Learning Technique for Identification of Yoga Poses” [1] tested six classification models of machine learning namely Logistic Regression, Random Forest, SVM, Decision Tree, Naive Bayes and KNN .But by using CNN and LSTM an accuracy of 99.04% was attained on a single frame when the Key Points were Open Pose was used to acquire the Keypoints. MediaPipe is faster because of its use of GPU. Dataset- 10 yoga poses, each class containing around 400 to 900 images, and the compiled Color Image dataset consists of 5459 images. Overall 94.28% accuracy was attained of all machine learning models, with Random Forest producing the most accuracy of approx 99%.

Chhaihuoy Long, et al. in “Development of a yoga posture coaching system using an interactive display based on transfer learning”[2] uses TL-MobileNet-DA through which they have achieved 98.43% accuracy. The Model is trained on 14 poses with a total of 120 images. By Performing data augmentation including sheer range of 0.2, zoom at 0.2, width shift and height shift in the range of 0.12, brightness at range (1, 1), and horizontal flip. Transfer learning (TL) is performed on six different pre-trained models, including VGG16, VGG19, MobileNet, MobileNetV2, InceptionV3, and DenseNet201, using the ImageNet dataset. This model uses the categorical cross-entropy loss function and Adam optimizer, with 100 epochs, in the training process and min delta value of 0.001 at patience 10 on EarlyStopping.

Nagalakshmi Vallabhaneni and Dr. P. Prabhavathy, discusses in “The Analysis of the Impact of Yoga on Healthcare and Conventional Strategies for Human Pose Recognition”[3] that yoga is perhaps the most straightforward physical, mental, and Spiritual work during this isolation period. Yoga Practice deals with these psychological issues, and along with breathing, meditation is the best general practice that will be careful with our body, brain, and soul. Yoga's impact on the people in the lockdown period is analyzed in this paper. The data was collected from 109 people, including 64 people who are male and 45 are female. Five parameters are considered: stress level, peace of mind, consciousness, physical health, and the respondent's mental health. The result of the survey is given in Table 1 and Table 2

TABLE I. SURVEY RESULT - I

Health Issue	Stage during lockdown (%)		
	Good	*Average*	*Bad*
Stress	50	40	10
Peace Of mind	60	30	12
Consciousness	60	33	5
Physical health	75	20	2
Mental health	60	30	10

TABLE II. SURVEY RESULT - II

Percentage	Technique[a]
68%	Yoga
14%	Walking
7%	Relaxation
6%	Exercise
2%	Meditation
3%	Others

a. Health Management technique used by different respondents

Wu, Y., Lin, Q, et al. in "A Computer Vision-Based Yoga Pose Grading Approach Using Contrastive Skeleton Feature Representations."[4] proposed a grading approach to input two yoga pose images from the learner and the coach, respectively, and then extract the human skeleton key points and calculate the feature similarity between them. This system used both the coarse triplet example and the fine triplet example. Mediapipe framework was used to extract the key points. Yoga pose classification image dataset adopted from Kaggle with 45 categories and 1931 images selected gave accuracy of 83.21%. Yoga pose grading image dataset constructed by the authors containing 3000 triplet examples with each triplet example consisting of three pose images that belong to the same yoga pose category produced 63.58% accuracy.

Deepak Kumar, and Anurag Sinha, proposed a system in "Yoga Pose Detection and Classification Using Deep Learning"[5] where keypoint Extraction of the human joint areas is done utilizing OpenPose which gives 18 keypoints. The system works at a rate of 3 FPS(frames per second). Dataset used comprises recordings of 6 yoga asanas performed by 15 distinct people (5 females and 10 guys) containing a total of 88 recordings (rate=30 FPS). This system uses the Support Vector Machine(SVM) algorithm which is best suited for classification giving a Train precision of 0.9953 and Validation exactness of 0.9762 and Test precision of 0.9319. But the CNN outperforms the SVM in terms of testing accuracy with Train exactness of 0.9878 Validation precision of 0.9921 and Test precision of 0.9858.

Girija Gireesh Chiddarwar, et al. in "AI-Based Yoga Pose Estimation for Android Application"[6] surveys the various technologies that can be used for pose estimation and concludes the best method based on the usability for an android application. This paper ponders Pose Estimation models like OpenPose, DeepPose, and PoseNet.

OpenPose uses Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields. The issue with OpenPose is that it necessitates specialized hardware and does not scale well on mobile devices.

DeepPose makes use of deep learning models, yet it performs poorly due to weak generalization.

PoseNet endorses the PoseNet model as it is an open-sourced technology that allows extracting the 17 essential points natively and a skeleton of the human pose is drawn with the help of these points, which is then used to derive angles between these points thus enabling us to effectively correct the user's yoga poses. It is deployed using Tensorflow. But The Pose net model only returns 17 key points, none of which includes fingers, which could lead to hand position modifications being limited.

M. Verma, S. Kumawat, Y. Nakashima and S. Raman, proposed in "Yoga-82: A New Dataset for Fine-grained Classification of Human Poses"[7], Yoga82 consisting of 82 complex yoga pose images downloaded from the web using the Bing search engine. It includes both Sanskrit and English names of yoga poses that were used to search for images and the downloaded images were cleaned and annotated manually. Every image contains one or more people doing the same yoga pose. Furthermore, images have poses captured from different camera view angles. This dataset has a varying number of images in each class from 64 (min.) to 1133 (max.) with an average of 347 images per class. Some of the images are downloaded from a specific yoga website. Hence, they contain only yoga poses with a clean background. However, there are many images with random backgrounds.

Nuruldelmia Idris, et al., discuss in "A Generic Review of Web Technology: DJango and Flask"[8] about Django and Flask python web frameworks. Django comes with a comprehensive MVC Framework that takes care of everything. While Django may be used to create a RESTful API on its own, it is one of the frameworks that sows fantastical creations. It is a feature-rich extension of the Django framework. On the other hand, Flask is a micro-framework that adheres to a set of rules: complete one task at a time effectively. It provided very little upfront, but it has a significant number of extensions that match Django's feature set.

Versioning – Django's flexibility makes the task less critical and multiple URL formats are supported and can be sent in as a request parameter.

Browsable API – Django Generates HTML pages to browse and execute all the endpoints where the users or developers can operate swiftly and simply.

Regular releases - the news version is always released twice a year to keep users up to date with the newest edition.

Speed – Flask is generally fast in performance compared to Django which might be due to Flask's simplistic design as it can handle hundreds of queries per second without slowing down the process.

NoSQL support –Flask is compatible with NoSQL databases such as MongoDB and DynamoDB.

Camillo Lugaresi, et al. discuss in “MediaPipe: A Framework for Perceiving and Processing Reality”[9], about various pose estimation algorithms. Mediapipe is a cross-platform library developed by Google that provides ready-to-use customizable ML solutions for live and streaming media,for computer vision tasks. Alternative of mediapipe include ARKit3,Microsoft Mixed Reality Toolkit,OpenPose

ARKit 3 Body Tracking produces several false positives similar to the ones in the picture, e.g.shadows on the walls, painting decorations being mistakenly identified as humans.

Microsoft Mixed Reality Toolkit SDK was an abstraction for the capabilities of the different platforms not suitable for all platforms.

OpenPose is based on DNN which requires a high-end computer Slight tradeoff between speed and accuracy. Current human pose performance metrics are based on keypoint accuracy. Failure cases still exist including foot and leg occluded and rare joint position.

Google MediaPipe’s accuracy of the solution was very high. MediaPipe is Fast is able to achieve its speed by the use of GPU acceleration and multi-threading. Mediapipe is Modular and Reusable. Its use of graphs, subgraphs, and calculators means that the work of one project can easily translate to the work of another. Mediapipe is Deployment Platform Friendly allowing to deploy the application not only to desktops but to mobile devices as well.

T. K. K. Maddala, et al. proposed a process"YogaNet: 3-D Yoga Asana Recognition Using Joint Angular Displacement Map s With ConvNets"[10] performed by integrating joint angular movements along with the joint distances in a spatiotemporal color-coded image called a joint angular displacement map(JADM). JADMs are trained with a single-stream CNN. The Steps include computing JADMs from 3D data then Color coding the JADMs then to train and test the CNN. A single 3D skeleton with J joints, there will be J×(J−1) / 2 unique joint pairs. Dataset prepared contains a total of 4200 3D yoga videos recorded for 42 poses and 100 variations per pose. Model consisting eight convolutional layers followed by two fully connected layers produced a accuracy of 90.01%

Chen, H., He, Y., & Hsu, C., in , “Computer-assisted yoga training system”[11], proposed a computer vision-based yoga training system, termed Y-system, to analyze the postures of a practitioner and assist in rectifying incorrect postures. This system analyzes up to twelve yoga poses. Topological skeletons are generated from the practitioner’s body maps of front and side views. Then, the Y-system extracts postural features including dominant axes, skeleton-based feature points, and contour based feature points. A distance map is produced by applying distance transform to the body map. Explicit posture description models for the 12 yoga poses based on their respective training is implemented. The accuracy of Y-system for each pose ranges from 76.22% to 99.87%

III. Proposed Methodology

Zyofisik contains three subsystems. Meditation coach, Exercise Trainer and Yoga assistant. All the three subsystems are integrated with the Flask web app. Mediapipe is used for retrieving landmarks of human pose which contains 33 keypoints. The 33 points are shown in Fig. 1.

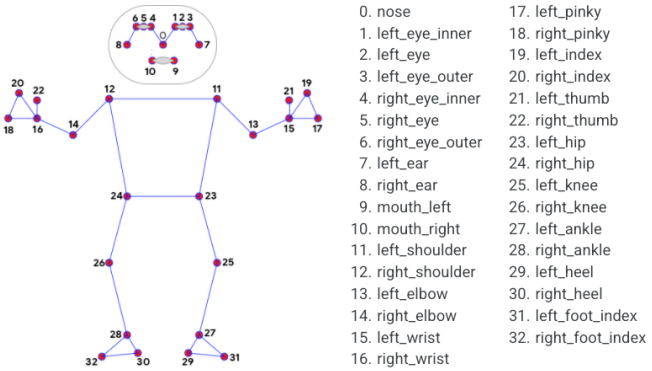

Fig. 1 Mediapipe Pose landmarks

A. *Meditation coach*

Meditation module gets the duration as input from the user on the client side. On receiving the input a thread is created which runs for the specified duration. For every 3 seconds the audio file changes between breathe in and breathe out file. The client checks for the current audio file every second and if there is a change it plays the new audio file.After the timer ends the thread stops and clears the audio file.

B. *Exercise Trainer*

The user must select the exercise he/she is going to perform. Based on the selected exercise the web page is rendered on the client side with the steps to perform and a graphical representation for reference. On clicking the start button the frames captured on the clients webcam are sent to the server side. The landmarks of the input image are retrieved using the Mediapipe Blazepose model and the figure is drawn on the image. According to the selected exercise the angle and distance made by the joints are calculated. The calculated angle is checked with the coded limits and if it passes the conditions the count is increased and stage is changed. The changed values and modified image is returned to the Client side. The angle made by three points is found using numpy’s arctan2(x1,x2) in radians. Later the angle in radians is converted to degrees.

radians= numpy.arctan2(c[1] - b[1], c[0] - b[0]) - numpy.arctan2(a[1] - b[1], a[0] - b[0])

degree= numpy.abs(radians * 180.0 / numpy.pi)

For bicep curl exercise the angle made by LEFT_SHOULDER, LEFT_ELBOW, LEFT_WRIST is calculated and when the angle increase above 160° then the stage is changed to “Down”. When the angle drops below 30° and the previous stage was “Down” then the count is increased and the stage is changed to “Up”. Similarly the waiter curl exercise adopts the same process with different angles and additionally the distance between the wrists are calculated using the distance formula because the exercise necessitates both the hands to be cupped together.

For Squats exercise the angle made by RIGHT_HIP, RIGHT_KNEE AND RIGHT_ANKLE is calculated and when the angle increase above 170° BUT BELOW 190° then the stage is changed to “Stand”. When the angle drops below 150° but above 90° or below 270° and above 210°

with the previous stage was “Stand” then the count is increased and the stage is changed to “Sit”.

C. *Yoga Assistant*

Yoga assistant uses a CNN model to predict the asana the user is performing in real-time. Yoga-82[7] is used for training the model.

The images downloaded from the Yoga-82 URL dataset are read and the landmarks are written into a CSV file. The CSV file is read and the landmarks and appended which is reshaped into (33,2,1) array. The labels for each image is an array containing 81 0’s and single 1. The index of 1 is the number of the asana. This array is appended for all images. The appended arrays are split into training and testing data at 8:2 ratio.

A sequential model is built with two 2D Convolution layers with kernel size of (1,1) and Relu activation function. The input size of the first Convolution layer is (None,33,2,1) . The model contains 3 Dense layers and a Dropout of 50%. The final Dense layer uses Softmax activation function. The model is compiled using Adam optimizer and categorical_crossentropy loss function and 0.1 loss weight. The model is trained with the training data with 50 epochs and 10 steps per epoch. The built CNN model gives an accuracy of 98.9% on test data. The trained model is converted to a tflite model and loaded in the web app.

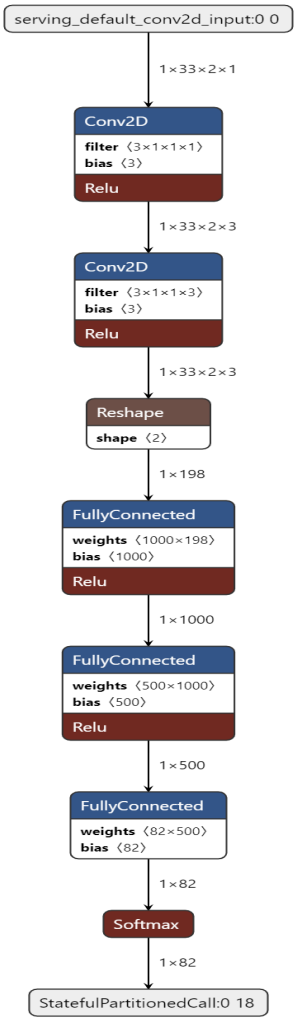

Fig. 2 CNN model layers

The user can select the asana he/she needs assistance for or allow the system to predict. The frames from the client side are sent to the server side .The pre-built CNN model is loaded on receiving the first request. The landmarks of the input image are retrieved using the Mediapipe Blazepose model and the figure is drawn on the image. The landmarks are reshaped into a 33x2 shaped array. The reshaped array is passed to the CNN model which returns the result containing 82 values. The accuracy of the selected asana is returned or if no asana is selected the asana with highest accuracy is returned.

IV. System Architecture

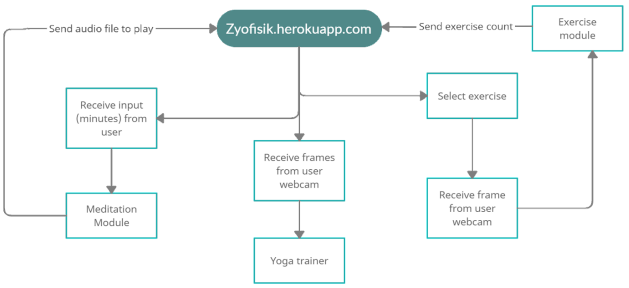

Fig. 3 UI System Architecture

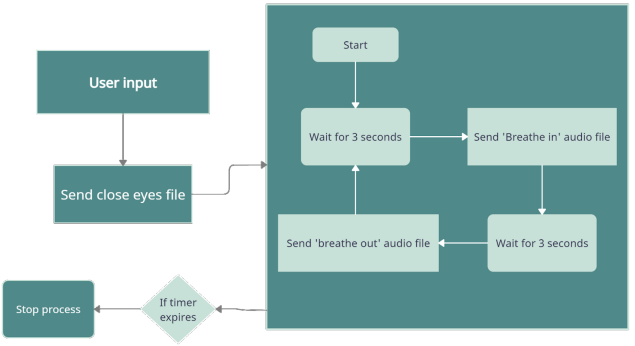

Fig. 4 Meditation Coach System Architecture

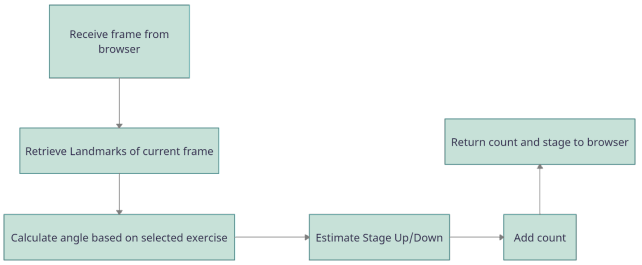

Fig. 5 Exercise Trainer System Architecture

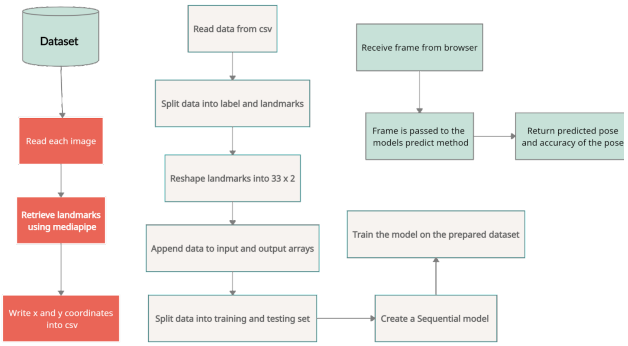

Fig. 6 Yoga Assistant System Architecture

V. Results and Discussions

Zyofisik is a Flask web app containing a Meditation Coach, Exercise Trainer and Yoga assistant. The system provides accurate scores for the yoga asanas with 100 as maximum score. The exercise counter takes into account only if the exercise is done correctly which helps the user to do the exercise properly. The exercise module currently contains 5 exercises counter. The number of exercises will be increased in future advancement. The web app is deployed in heroku cloud. The client sends the frame to the server for every 500 milliseconds in exercise trainer and for every second in yoga assistant which will also be reduced in future advancement.

VI. Conclusion

Transitioning to Modern fitness, this paper suggests a system that can classify 82 yoga poses with an accuracy of 98.98%, accurately assesses the exercises performed to aid the user performing the workouts as they can revise their posture accordingly, and instructs the user to maintain their breathing pace using a programmed meditation coach.

References

[1] Yash Agrawal, Yash Shah, and Abhishek Sharma, (2020), “Implementation of Machine Learning Technique for Identification of Yoga Poses”, 9th IEEE International Conference on Communication Systems and Network Technologies, pp. 40-43

[2] Chhaihuoy Long, Eunhye Jo, and Yunyoung Nam, (2022), “Development of a yoga posture coaching system using an interactive display based on transfer learning” - J Supercomput 78, 5269–5284

[3] Nagalakshmi Vallabhaneni and Dr. P. Prabhavathy, (2021), “The Analysis of the Impact of Yoga on Healthcare and Conventional Strategies for Human Pose Recognition”, Turkish Journal of Computer and Mathematics Education (TURCOMAT) .,vol. 12 no. 6, pp. 1772–1783

[4] Wu, Y., Lin, Q., Yang, M., Liu, J., Tian, J., Kapil, D.P., & Vanderbloemen, L. (2021). “A Computer Vision-Based Yoga Pose Grading Approach Using Contrastive Skeleton Feature Representations.” Healthcare (Basel), 10(1):36

[5] Deepak Kumar, and Anurag Sinha, (2020), “Yoga Pose Detection and Classification Using Deep Learning” International Journal of Scientific Research in Computer Science, Engineering and Information Technology, Volume 6 Issue 6 Page: 160-184.

[6] Girija Gireesh Chiddarwar, Abhishek Ranjane, Mugdha Chindhe, Rachana Deodhar, and Palash Gangamwar, (2020), “AI-Based Yoga Pose Estimation for Android Application” International Journal of Innovative Science and Research Technology, Volume 5 - Issue 9.

[7] M. Verma, S. Kumawat, Y. Nakashima and S. Raman, (2020), "Yoga-82: A New Dataset for Fine-grained Classification of Human Poses", IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) pp. 4472-4479

[8] Nuruldelmia Idris, Cik Feresa Mohd Foozy, and Palaniappan Shamala, (2020), “A Generic Review of Web Technology: DJango and Flask”, International Journal of Advanced Computing Science and Engineering ISSN 2714-7533 Vol. 2, No. 1

[9] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays,Fan Zhang, Chuo-Ling Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua,Manfred Georg and Matthias Grundmann, (2019), “MediaPipe: A Framework for Perceiving and Processing Reality”, Computer Vision for AR/VR at IEEE Computer Vision and Pattern Recognition (CVPR).

[10] T. K. K. Maddala, P. V. V. Kishore, K. K. Eepuri and A. K. Dande, (2019), "YogaNet: 3-D Yoga Asana Recognition Using Joint Angular Displacement Maps With ConvNets," IEEE Transactions on Multimedia, vol. 21, no. 10, pp. 2492-2503, Oct. 2019

[11] Chen, H., He, Y., & Hsu, C., (2018), “Computer-assisted yoga training system”, Multimedia Tools and Applications, 77, 23969-23991.