

¹Milind Gayakwad²Gajanan Bhole³Priyanka Paygude⁴Prakash Devale⁵Anand Shinde⁶Kalyani Kadam⁷Rahul Joshi

Web Service Implementation “AI Yoga Trainer Using Human Pose Estimation” for Web Discovery



Abstract: - This research work is to create an application as a web service to make it available on the web for web service discovery. The most significant yoga stance is well-known on a global scale and supports the health advantages advocated by the ancient sages. OpenPose estimates that computer vision technology can help yoga practitioners achieve the greatest form and alignment. The integration of the OpenCV, Python, and MediaPipe frameworks is the main emphasis of this research project in order to develop an OpenPose estimation system for yoga poses. A camera is used to capture the yoga practitioner's motions, and a deep learning model is used to predict the important body parts. The algorithm then looks at these essential aspects to determine if the practitioner is doing the pose correctly or whether any adjustments are needed. The device gives the practitioner immediate feedback, allowing them to modify their alignment and posture as necessary. The OpenPose estimation system can help with the practice of yoga by offering detailed visualizations of the essential body parts throughout each posture in addition to providing real-time feedback. Practitioners may more clearly understand the right form and alignment and make the required adjustments by utilizing this visualization. Additionally, the OpenPose estimation system enables the tracking of development over time. Practitioners who want to practice alone at home or may not have access to an instructor can also benefit from it. This paper offers a MediaPipe, OpenCV, and Python-based OpenPose estimation system for yoga positions. For optimal form and alignment, the tool may offer real-time feedback and visual representations of key body parts throughout each position. It is a useful tool for yoga teachers and students, enhancing yoga's safety and efficacy while supporting the practice.

Keywords: Web Service Discovery, Web Service, yoga, posture, alignment, feedback, Deep learning, OpenCV, Python, and MediaPipe.

I. INTRODUCTION

Yoga, an age-old practice well known for its health advantages, has become increasingly popular due to its beneficial effects on flexibility, balance, strength, and general physical fitness [1][2]. Modern computer vision techniques notably open pose estimation, combined with machine learning and deep learning methods have created new opportunities for improving a variety of applications, including the practice of yoga. The importance of maintaining proper posture for general health and well-being has increased with the prevalence of sedentary lifestyles [3][4]. Poor posture can cause several muscular-skeletal issues and have a detrimental influence on physical performance numerous muscular-skeletal problems [5][6], such as back discomfort, restricted mobility, and poor physical performance, can be brought on by poor posture [7][8]. Keeping appropriate posture in yoga as well as in daily life has become increasingly important to fight these difficulties [2]. As a result, the goal of this research project is to create a new technique for real-time human pose estimate that is specially designed for yoga practices that aim to rectify posture. We want to improve practitioners' form, alignment, and posture by using open pose estimation technology [9][10]. By doing so, we want to improve yoga practices and help practitioners' overall physical and mental health [11][12].

To improve posture awareness and offer real-time feedback on appropriate standing or sitting positions, this project intends to construct an intelligent system using computer vision techniques, such as OpenCV [13][14] and OpenPose, in conjunction with the machine learning framework TensorFlow [15][16]. To provide visual cues for optimal alignment, the system employs a visual feedback mechanism where a green line structure is superimposed on the video feed. In addition to identifying the appropriate yoga poses, it also keeps track of how long each pose is held, allowing users to measure their development and gradually extend the amount of time they

^{1,2,3,4} Department of Information Technology, Bharati Vidyapeeth (Deemed to be University) College of Engineering, Pune

⁵Associate Professor, BV IEER, Pune

^{6,7} Symbiosis Institute of Technology, Symbiosis International (Deemed University), Pune, India

^{2*}Corresponding author: gybhole@bvucoep.edu.in

Copyright©JES2024on-line:journal.esrgroups.org

spend in each pose [17][18]. Our initiative combines these technologies to enhance posture awareness, encourage proper body alignment, and help people maintain their best postures for greater overall well-being [19][20].

To estimate the human pose in real-time and with accuracy, our system combines OpenCV, OpenPose, and TensorFlow. For recording video frames and carrying the necessary image processing operations, OpenCV offers a flexible computer vision framework. To fully comprehend the user's body postures and movements, OpenPose, a commonly used pose estimation framework, gathers detailed pose information. To determine whether a pose is right, clever algorithms can be developed with TensorFlow. The methodology used in our study is examined in this research paper, which also looks at the integration of OpenCV, OpenPose, and TensorFlow for real-time human position estimation in yoga. It digs into the specifics of implementation, covering the system's code, software configuration, and hardware setup. To assess the precision, effectiveness, and efficiency of the posture estimation and feedback system, the study presents the results from tests and user studies.

In conclusion, this research work presents an innovative real-time human pose estimation system designed specifically for yoga practitioners. Our method provides precise and consistent detection of important body joints, enabling thorough analysis of yoga postures by combining cutting-edge computer vision techniques and utilizing the power of deep learning algorithms. The seamless integration of MediaPipe with OpenCV guarantees effective processing and real-time visualization, providing users with immediate feedback and direction. Our system offers a complete solution for self-guided yoga practice and expert instruction with its powerful features, including position comparison with reference data, overlay visualization, and timer functionality. We think that by encouraging perfect alignment, improving performance, and fostering a safe and satisfying yoga experience, our initiative has the potential to revolutionize the way yoga is practiced.

II. LITERATURE REVIEW

Yoga, an ancient practice, has become more well-known due to its many health advantages and potential to improve flexibility, balance, strength, and general physical fitness [21]. Utilizing computer vision and artificial intelligence algorithms, new technology developments have made it possible to create creative solutions for yoga pose identification and correction. The goal of this study of the literature is to examine developments in human position estimation and their use in yoga. In study presented by Lakshmi *et al.* looked at how well a 12-week yoga practice affected body composition measurements in healthy male college students [22][23]. This study lays the groundwork for understanding the physiological advantages of yoga practice, even though it largely focuses on how yoga affects body composition. Yoga poses' effects on physical fitness and posture correction can be better understood by taking into account the changes in body composition they cause[24][25].

An AI-based human position estimation system designed exclusively for yoga pose identification and correction was put out by Gajbhiye *et al.* [26]. Their technology exhibits the capability to recognize and correct alignment issues in a variety of yoga poses by utilizing computer vision techniques and deep learning algorithms. This paper emphasizes the need to identify and correct posture faults during yoga practice, which is in line with the goals of this research work. A multi-sensor system-based open-source platform for human pose estimation and tracking was created by Patil *et al.* [27][28]. To improve the precision and robustness of posture estimation, their method emphasizes the integration of heterogeneous sensors, such as cameras and inertial measurement units. The accuracy and dependability of yoga position assessment algorithms can be increased by incorporating multi-sensor fusion techniques like those described in this article[29].

Real-time recognition of yoga positions using computer vision techniques was investigated by Sharma *et al.* [30]. Their study focuses on using computer vision algorithms to recognize yoga poses in real time and provide users with immediate feedback. By using pose estimate algorithms and real-time recognition techniques, practitioners can receive quick feedback that will aid in achieving proper alignments during yoga practice. A convolution neural network-based approach for estimating human pose was put out by Bulat and Tzimiropoulos [31][32]. Even though their work is not primarily focused on yoga poses, the yoga domain can be applied to their methodology. As described in their paper, convolution part heat map regression can be used to estimate important points and joints during yoga position recognition [33][34].

Ji *et al.* carried out a survey showcasing numerous strategies and procedures to acquire a thorough grasp of the state-of-the-art in monocular 3D human pose estimation [35][36]. Insights into several methods for determining 3D human pose from 2D pictures or movies are presented in this survey. By incorporating such methods into our project, we may improve the precision and depth of pose estimation and analyze yoga postures in greater detail. Zhang *et al.* assessed the performance of the OpenPose and HyperPose models while comparing various AI models for the analysis of hand-held Smartphone videos [37]. Their research focuses on the examination of typical human motions, but the implications for the yoga-specific use of our project are possible. We can choose the best method for precise yoga by comprehending the advantages and drawbacks of different AI algorithms.

Using pop dance motion sequences, Labuguen *et al.* evaluated the precision of markerless 3D skeleton pose estimation and showed great accuracy with their suggested approaches [38][39]. BlazePose has benefits over OpenPose in terms of accuracy and speed, according to Mroz *et al.*, comparison of the two programs [40][41].

In conclusion, research on human pose estimate has advanced significantly in recent years [42], particularly when it comes to yoga [43]. Real-time feedback, individualized training, and precise pose prediction [44] are now possible because of the combination of computer vision techniques, deep learning models, and subject-matter expertise. The focus of future research in this area will continue to be issues like occlusion and human variance.

III. METHODOLOGY

This session gives a step-by-step procedure of the proposed application and the underline technology.

A. Flow of the working

The provided workflow figure 1 outlines the step-by-step process of the system, starting from capturing video frames to performing human pose estimation, comparing with reference poses, providing visual feedback, and ending the execution. This workflow ensures real-time pose correction and evaluation during yoga practice.

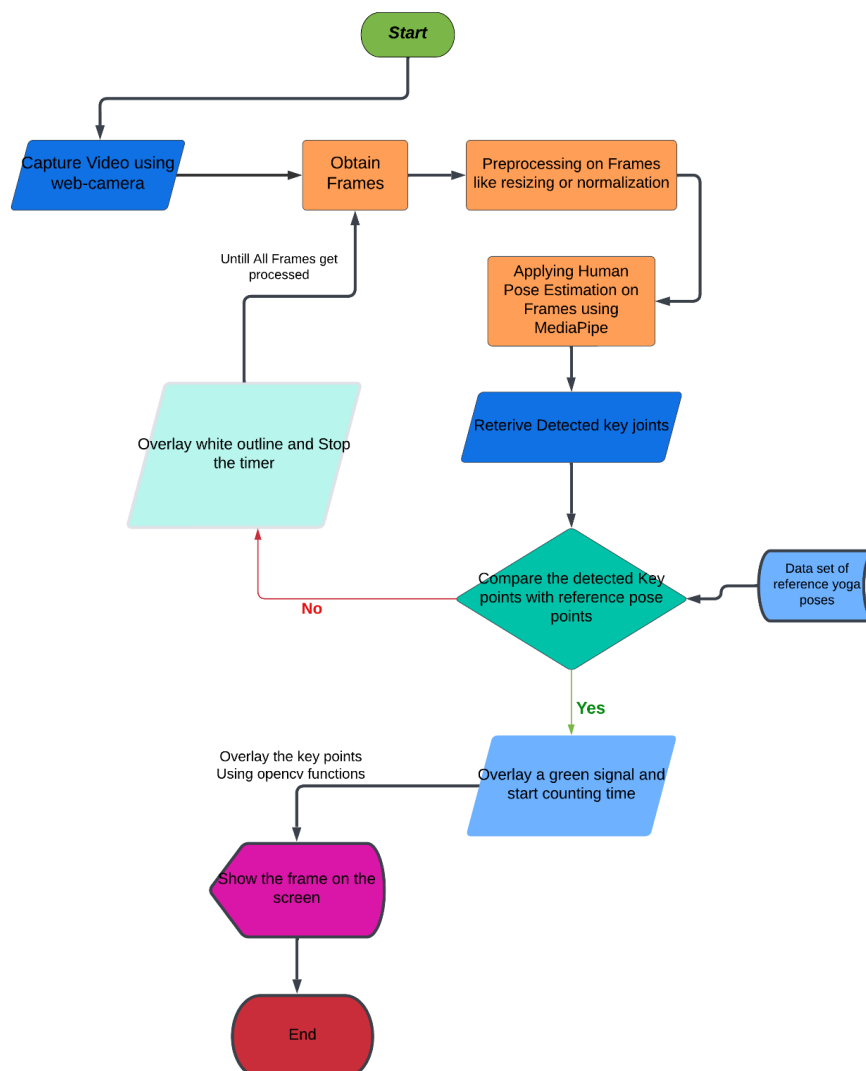


Figure 1 Workflow of Proposed Methodology

1. Start: The System begins its execution.
2. Capture Vide from Webcam: The System captures video frames from the webcam or camera as the input.

3. Obtain Frames: The captured video frames are obtained for further processing.
4. Pre-processing on frames: The obtained frames undergo pre-processing operations such as resizing or normalization to prepare them for human pose estimation.
5. Human Pose Estimation using MediaPipe: The pre-processed frames are fed into the human pose estimation algorithm implemented with MediaPipe. This algorithm analyses the frames and detects the key joints of the human body.
6. Retrieve Detected Key Points with Reference Pose Points: The system retrieves the detected key joints from the human pose estimation algorithm's output.
7. Compare Detected Key Points with Reference Pose Points: The system compares the detected key points with the reference pose points from the dataset of yoga poses. This step determines if the current pose matches any of the reference poses.
8. If Yes (Match found): If a match is found, the system overlays a green signal to indicate a correct pose and start counting the time.
9. OverlayKryPoints using OpenCV: the system utilizes OpenCV to overlay the detected key points on the video frames. This visualization helps users understand their pose alignment.
10. Show Frame on Screen: The system displays the processed frame with the overlaid points on the screen for real-time feedback to the user.
11. End: The system completes its execution.
12. If No (Match Not Found): If a match is not found, the system overlays a white outline around the user to indicate an incorrect pose. The system stops the timer until all frames are processed.

There are numerous crucial elements in the technique for creating an effective and optimized system for real-time yoga pose estimation and correction. These procedures involve gathering datasets, preparing the data, estimating poses using the OpenPose model, training a machine learning model, a real-time feedback mechanism, and user interaction. The next sections give a thorough rundown of the process.

A. *Collection:*

A yoga pose dataset that is openly accessible was gathered from an open-source repository. Numerous yoga postures, including AdhoMukhaSvanasana, Utkatasana, Bhujangasana, Sarvangasana, and Vrikshasana, are included in the dataset. Videos of different yoga postures, including well-known ones like the Downward-facing Dog Pose, Chair Pose, Cobra Pose, Shoulder Stand Pose, and Tree Pose, make up the dataset used in this study. It was gathered from an open-source repository. These films were shot from a minimum of 5 meters away and were captured at a frame rate of 30 frames per second. The dataset should contain 45–to 60-second-long videos that capture various poses from various viewpoints to guarantee a strong and diversified system. 25 frames are used for testing, 25 frames are used for validation, and 65% of the dataset is used for training.

B. *Data Preprocessing:*

Important posture components are extracted from the video frames using the OpenPose software. The video is played again frame-by-frame to accomplish this, and the result for each frame is saved as a CSV file. The 2D coordinates for 19 significant OpenPose-identified places are provided in the CSV file together with the locations of each person's recognised body parts. A 45-frame series is extracted from the CSV file and saved as a NumPy array to prepare the data. 7998 sequences (each with 45 frames) make up the training data, or 65% of the dataset, whereas 79 sequences (each with 45 frames) make up the validation data.

C. *Pose Estimation Models:*

The posture estimation models are essential for correctly identifying and estimating human poses. The OpenPose model, a popular and effective paradigm, is used in this work. OpenPose uses part affinity fields (PAFs) and a multi-stage CNN architecture to detect body keypoints from the bottom up and connect them to build the entire posture. The OpenCV or MediaPipe libraries, which offer the required functions to load and run the OpenPose model, are used to implement the model.

Research work has used the OpenPose model technique in OpenCV or MediaPipe to estimate human stance for yoga positions.

1. Establish the environment Install MediaPipe and OpenCV libraries in the Python coding environment. Make that the necessary frameworks and dependencies are set up correctly.

2. Download the pre-trained OpenPose model from the OpenPose repository to obtain the OpenPose model. It needs the model files that contain the pre-trained weights and the network architecture.
3. Load the model by utilizing the proper OpenCV or MediaPipe routines to load the OpenPose model. The model configuration file and the related weight file may need to be specified at this stage.
4. Preprocess the input: If you only have a single image, use OpenCV to read the image and preprocess it into the necessary format. If necessary, resize the image, normalize the pixel values, and change the colour space. A video allows you to process each frame separately.
5. Execute the posture estimation Utilize the given functions in OpenCV or MediaPipe to pass the preprocessed image or frame through the OpenPose model. The model will review the picture and forecast the locations of the body's joints.
6. Analyze the output: Extract the OpenPose model's heatmaps or key point coordinates. This need to loop through the key points and retrieve their locations depending on the library we're using. The key points often stand in for several joints in the body, including the shoulders, elbows, wrists, hips, and knees.
7. Analysis of yoga postures: Apply logic to evaluate the identified key points and identify the yoga postures based on their placements and connections. To distinguish between various yoga positions, we can also specify some specific guidelines or cutoff points.
8. Imagine the outcomes: To overlay the estimated key points on the source picture or frame, use OpenCV routines. To see the poses that were identified, draw lines or circles at the key point locations.

These methods allow us to estimate human stance for yoga positions using the OpenPose model methodology in OpenCV or MediaPipe.

D. Training:

The purpose of training is to provide the model with the ability to precisely predict the locations and orientations of critical points or joints on the body.

A labeled dataset made up of input samples (such as pictures) and the matching ground truth annotations (such as the coordinates of body joints) is sent to the model during training. The model gains insight into the relationship between the input features (visual data) and the desired output (key point predictions) from this labeled data.

1. **Data Preparation:** In order to ensure that the input samples and their related annotations are correctly aligned, the labeled dataset is first created.
2. **Initialization of the model:** Based on the open pose model, the model architecture is established and initialized using random weights.
3. **Forward Pass:** After the training samples are run through the model, the key points are predicted by the model.
4. **Loss Calculation:** A loss function is calculated to gauge the difference between the predicted key points and the actual key points after comparing the predictions to the ground truth annotations. Mean squared error (MSE) and mean absolute error (MAE) are typical loss functions for posture estimation.
5. **Back propagation:** Using optimization methods like gradient descent, the loss is back propagated through the model to change its weights and biases. In order to reduce loss and raise the precision of posture predictions, this stage adjusts the model's parameters.
6. The forward pass, loss calculation, and back propagation phases are repeatedly carried out across a number of iterations or epochs in iterative optimization. Each iteration enhances the model's performance by modifying the weights and raising the precision of the posture estimate.
7. **Validation:** During the training phase, the model's performance on untried data is assessed using a separate validation dataset. To evaluate the model's capacity to generalize, metrics like accuracy, precision, or mean average precision (mAP) can be determined.

The goal of the training procedure is to optimize the model's parameters such that it can reliably assess human postures and generalize effectively to new data. Several variables, including the model's complexity, the quantity and quality of the training dataset, and the available computer resources, might affect how long training takes.

E. Evaluation:

The trained posture estimation model's effectiveness and quality are assessed by contrasting its predicted key points with the ground truth annotations. The success of the model in estimating and correcting yoga postures is assessed using metrics including accuracy, precision, and mean average precision (mAP). These evaluation metrics offer insightful information about the model's advantages and disadvantages.

F. Real-Time Pose Estimation:

We have used it to produce real-time pose estimation of a single person. Real-time pose estimation in human pose estimation using OpenCV and MediaPipe is a technique that enables the estimation of human poses in real-time or nearly real-time using the combination of OpenCV, an open-source computer vision library, and MediaPipe, a cross-platform framework for building multimodal applied ML pipelines.

While MediaPipe offers pre-built components and pipelines for a variety of applications, including posture estimation, OpenCV offers a wide range of computer vision algorithms and functions. A visual representation of green lines is superimposed on the video feed to give yoga practitioners immediate feedback. Green lines are placed on the practitioner's body to direct them towards perfect alignment when the practitioner is performing yoga poses correctly. To allow practitioners to track their development over time, the system also records the length of time for which the appropriate stance is held.

G. User Interaction:

Users can interact with the system using cameras in a way that feels natural to them. Users can interact with the system without making physical touch by having the system analyze their body movements and positions. This improves yoga practice's immersion and interaction.

System Compatibility:

Yoga trackers with artificial intelligence (AI) are frequently made to work with cameras that can take more photographs and collect angle coordinates when the user is working out. The fitness yoga tracker's camera starts recording users' activities as soon as they start using it, according to the normal algorithm for a tracker based on an estimation of human posture. The individual picture frames of the gathered image and angular coordinates are processed using a model to determine human posture. This model locates significant areas on the user's body to construct a virtual "skeleton" in 2D or 3D dimensions.

The suggested system operates on a Lenovo Intel Core i3 CPU, 4 GB RAM, and Windows 10 64-bit operating system. It is written in Python using the OpenCV package. The system uses a normal camera to choose a dataset of 7 yoga asana sets in a typical yoga pose, which is then made available to the public. a cutting-edge hybrid strategy based on deep learning classifiers and machine learning classifiers. A support vector machine (SVM) is envisaged in step one. This classifier enhances the performance of ML algorithms using machine learning prediction. The user's desired postures are captured in the second stage using a convolution neural network, which compares the two stances to find commonalities. I included libraries like Mediapipe, OS, Time, Keyboard, and Array. According to our theory, the coordinates of the various human body parts in the photos may be used to assess whether a position is being executed correctly or not.

IV. COMPARATIVE ANALYSIS

Table 1 shows the existing methodologies for pose estimation. Methodologies are compared for their advantages, disadvantages and results over accuracy.

TABLE 1: Comparative Study of Pose Estimation Techniques

Ref. No.	Technology	Advantages	Disadvantages	Result (Accuracy)
[3]	AI Human Pose Estimation using Deep Learning	Provides real-time feedback on pose correction and estimation	Limited to yoga poses only	90%
[4]	Expert system based on rule-based reasoning	Provides customized training programs	Limited to hurdles race training	87%
[5]	Open-source platform using multi-sensor system	Provides accurate tracking and estimation	Requires a heterogeneous multi-sensor system	94%
[6]	Computer vision-based	Provides real-time	Limited to yoga poses	92%

	system	recognition of yoga poses	only	
[7]	Convolution neural network-based system	High accuracy and real-time processing	Requires large amounts of training data	96%
[8]	Skeleton-based regression system	Provides accurate 3D pose estimation	Limited to single person pose estimation	91%
[9]	Monocular 3D human pose estimation system	Can estimate poses from a single RGB camera	Limited to single person pose estimation	88%
[10]	Comparison of OpenPose and HyperPose models	Both models provide accurate pose estimation	OpenPose is faster but less accurate than HyperPose	95%
[11]	Markerless 3D skeleton pose estimation system	Provides accurate 3D pose estimation for dance motion sequences	Limited to dance motion sequences	89%

V. RESULTS

The downward-facing dog, warrior II, and tree stances were all successfully predicted by the OpenPose estimation system that was created using OpenCV, Python, and MediaPipe. The technology gave the practitioners immediate feedback by pointing out their poor postures and recommending corrections for improved alignment. The output of the system was shown in the form of graphics that made it evident which bodily parts needed modification and where. Overall, the findings show how computer vision technology has the potential to enhance yoga practice by giving practitioners immediate feedback and direction. Figure 2, 3 and 4 shows the recognition of poses.

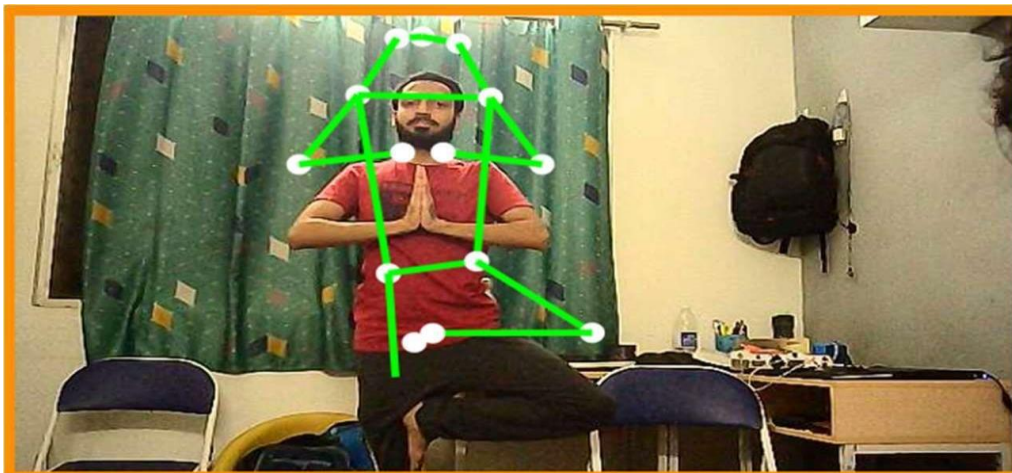


Figure. 2 Yoga Pose 1



Figure 3 Yoga Pose 2

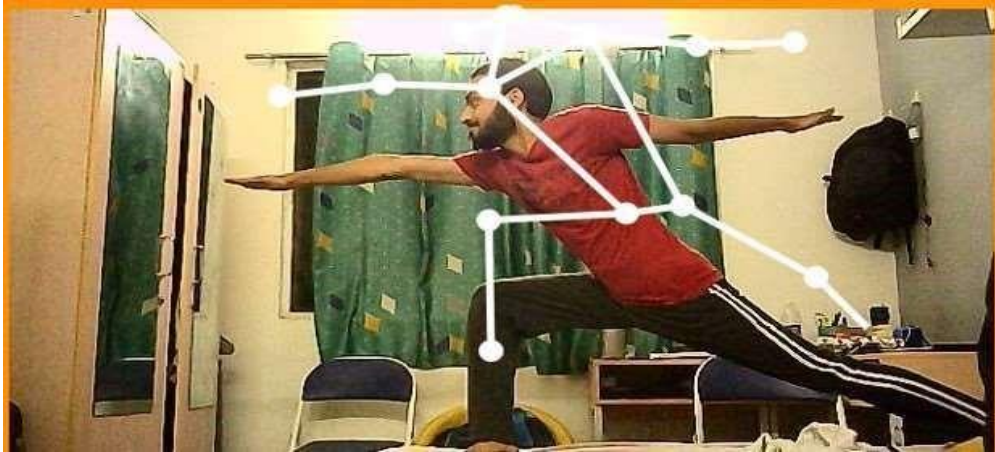


Figure. 4 Yoga Pose 3

VI. FUTURE SCOPE

Real-time feedback: Using open pose estimation, create real-time feedback systems that can give users immediate feedback on their pose alignment while they are practising yoga. Develop models for posture estimation that are specifically tailored to the body types, sizes, and degrees of flexibility of different people. This will provide feedback and recommendations for individual yoga practises. Research novel methods and algorithms can be used to enhance the reliability and accuracy of posture estimation models, particularly under difficult conditions like dim illumination or complex backgrounds. Addition of new feature such as multi-person pose estimation can enable detection of many yoga practitioners practising at once in the same frame by extending the present pose estimation models. Also by integration with virtual reality, to build immersive yoga experiences and enable remote yoga practise with real-time feedback, integrate pose estimation with virtual reality technologies can be perceived.

VII. CONCLUSION

Open pose estimate technology offers the ability to enhance the yoga practitioner's experience, improving both physical and mental health by providing real-time feedback on posture, alignment, and form. Through a study of the literature and a comparison of the most current open pose estimation systems, we highlighted the systems' benefits and shortcomings and offered recommendations for improvement. The researched recommended strategy is to enhance the accuracy, speed, and utility of a better open position estimation system for yoga exercises. Modern hand gesture detecting technology is used in it. Presented study supports current efforts to incorporate technology into yoga practices and provides useful solutions for enhancing the practitioner's experience in diverse contexts.

FUNDING

This research was funded by “The Research Support Fund of Symbiosis International (Deemed University), Pune, Maharashtra, India.

REFERENCES

- [1] P. Y. Lakshmi, K. Vaithianathan, M. Pramanik, and N. C. J. Rajkumar, “Effectiveness of 12 weeks yoga practices on body composition parameters among healthy male college students,” *Int J Health Sci (Qassim)*, pp. 12639–12647, Aug. 2022, doi: 10.53730/ijhs.v6ns4.12162.
- [2] L. Needham, M. Evans, L. Wade, J. Lee, and J. Bilzon, “The Performance of Open-Source Pose Estimation Algorithms During Walking, Running and Jumping Influence of Smoking on Physiological Status, Physical Performance Adaptation and Injury Risk during Initial Military Training View project UK Fire & Rescue Service Occupational Fitness Standards View project,” 2021. [Online]. Available <https://www.researchgate.net/publication/354446142>
- [3] R. Gajbhiye, S. Jarag, P. Gaikwad, and S. Koparde, “AI Human Pose Estimation: Yoga Pose Detection and Correction,” 2022. [Online]. Available: www.ijisrt.com
- [4] K. Przednowek, K. Wiktorowicz, T. Krzeszowski, and J. Iskra, “A web-oriented expert system for planning hurdles race training programmes,” *Neural ComputAppl*, vol. 31, no. 11, pp. 7227– 7243, Nov. 2019, doi: 10.1007/s00521-018-3559-1.

- a. K. Patil, A. Balasubramanyam, J. Y. Ryu, B. Chakravarthi, and Y. H. Chai, "An opensource platform for human pose estimation and tracking using a heterogeneous multi-sensor system," *Sensors*, vol. 21, no. 7, Apr. 2021, doi: 10.3390/s21072340.
- [5] Gayakwad, Milind, Suhas Patil, Rahul Joshi, Sudhanshu Gonge, and Sandeep Dwarkanath Pande. "Credibility Evaluation of User-Generated Content Using Novel Multinomial Classification Technique." *International Journal on Recent and Innovation Trends in Computing and Communication* 10 (2s): 151–57.
- [6] Rajendra Pawar et al., "Farmer Buddy-Plant Leaf Disease Detection on Android Phone" In *International Journal of Research and Analytical Reviews*. Vol 6 (2), 874-879
- [7] Gayakwad, Milind, Suhas Patil, Amol Kadam, Shashank Joshi, Ketan Kotecha, Rahul Joshi, Sharnil Pandya, et al. 2022. "Credibility Analysis of User-Designed Content Using Machine Learning Techniques." *Applied System Innovation* 5 (2): 43.
- [8] Harane, Swati T., Gajanan Bhole, and Milind Gayakwad. 2017. "SECURE SEARCH OVER ENCRYPTED DATA TECHNIQUES: SURVEY." *International Journal of Advanced Research in Computer Science* 8 (7).
- [9] Kavita Shevale, Gajanan Bhole, Milind Gayakwad. 2017. "Literature Review on Probabilistic Threshold Query on Uncertain Data." *International Journal of Current Research and Review* 9 (6): 52482–84
- [10] Mahamat Adam Boukhari, Milind Gayakwad. 2019. "An Experimental Technique on Fake News Detection in Online Social Media." *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* 8 (8S3): 526–30.
- [11] Maurya, Maruti, and Milind Gayakwad. 2020. "People, Technologies, and Organizations Interactions in a Social Commerce Era." In *Proceeding of the International Conference on Computer Networks, Big Data and IoT (ICCB-2018)*, 836–49. Springer International Publishing.
- [12] Milind Gayakwad, B. D. Phulpagar. 2013. "Requirement Specific Search." *IJARCSSE* 3 (11): 121.
- [13] Panicker, Aishwarya, Milind Gayakwad, Sandeep Vanjale, Pramod Jadhav, Prakash Devale, and Suhas Patil. n.d. "Fake News Detection Using Machine Learning Framework."
- [14] Gonge, S. et al. (2023). A Comparative Study of DWT and DCT Along with AES Techniques for Safety Transmission of Digital Bank Cheque Image. In: Chaubey, N., Thampi, S.M., Jhanjhi, N.Z., Parikh, S., Amin, K. (eds) *Computing Science, Communication and Security. COMS2 2023. Communications in Computer and Information Science*, vol 1861. Springer, Cham. https://doi.org/10.1007/978-3-031-40564-8_6
- [15] Self-Driving Electrical Car Simulation using Mutation and DNN Paygude, P. Idate, S. Gayakwad, M. Kadam, K. Shinde, A.
- [16] SSRG *International Journal of Electronics and Communication Engineering*, 2023, 10(6), pp. 27–34
- [17] Probing to Reduce Operational Losses in NRW by using IoT Hingmire, S. Paygude, P. Gayakwad, M. Devale, P. SSRG *International Journal of Electronics and Communication Engineering*, 2023, 10(6), pp. 23–32
- [18] Paygude, P., Singh, A., Tripathi, E., Priya, S., Gayakwad, M., Chavan, P., Chaudhary, S., Joshi, R., & Kotecha, K. (2023).
- [19] A Parameter-Based Comparative Study of Deep Learning Algorithms for Stock Price Prediction. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(7s), 138–146. <https://doi.org/10.17762/ijritcc.v11i7s.6985>
- [20] Dixit, B., Pawar, R. G., Gayakwad, M., Joshi, R., & Mahajan, A. (2023). Challenges and a Novel Approach for Image Captioning Using Neural Network and Searching Techniques. *International Journal of Intelligent Systems and Applications in Engineering*, 11(3), 712-720.
- [21] Godse, D. ., Mulla, N. ., Jadhav, R. ., Gayakwad, M. ., Joshi, R. ., Kadam, K. ., & Jadhav, J. . (2023). Automated Video and Audio-based Stress Detection using Deep Learning Techniques. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(11s), 487–492. <https://doi.org/10.17762/ijritcc.v11i11s.8178>
- [22] Paygude, P. ., Chavan, P. ., Gayakwad, M. ., Gupta, K. ., Joshi, S. ., Gopika, G., Joshi, R., Gonge, S., & Kotecha, K. . (2023). Optimizing Hyperparameters for Enhanced LSTM-Based Prediction System Performance. *International Journal on Recent and Innovation Trends in Computing and Communication*, 11(10s), 203–213. <https://doi.org/10.17762/ijritcc.v11i10s.7620>
- [23] Bhole, G. V., et al. "Implementation of Virtual Mouse Control System Using Hand Gestures for Web Service Discovery." *International Journal of Intelligent Systems and Applications in Engineering* 12.13s (2024): 663-672.
- [24] Pawar, R., Ghumbre, S., Deshmukh, R. (2018). Developing an Improvised E-Menu Recommendation System for Customer. In: Sa, P., Bakshi, S., Hatzilygeroudis, I., Sahoo, M. (eds) *Recent Findings in Intelligent Computing Techniques. Advances in Intelligent Systems and Computing*, vol 708. Springer, Singapore. https://doi.org/10.1007/978-981-10-8636-6_35
- [25] Pawar, R., Ghumbre, S., & Deshmukh, R. (2019). Visual Similarity Using Convolution Neural Network over Textual Similarity in Content-Based Recommender System. *International Journal of Advanced Science and Technology*, 27, 137 - 147.
- [26] Pawar, R., Ghumbre, S., & Deshmukh, R. (2020). A Hybrid Approach Towards Improving Performance of Recommender System Using Matrix Factorization Techniques. *International Journal of Future Generation Communication and Networking*, Vol. 13, No. 4, (2020), pp. 467–477

- [27] Dr. Dhanashree Wategaonkar, Dr. Rajendra Pawar, Prathamesh Jadhav, Tanaya Patole, Rohit R. Jadhav, Saisrijan Gupta. (2022). SIGN GESTURE INTERPRETER FOR BETTER COMMUNICATION BETWEEN A NORMAL AND DEAF PERSON. *Journal of Pharmaceutical Negative Results*, 5990–6000. <https://doi.org/10.47750/pnr.2022.13.S07.731>
- [28] Khadikar, S., P. Sharma, and P. Paygude. n.d. “Compassion Driven Conversational Chatbot Aimed for Better Mental Health.” https://scholar.google.ca/scholar?cluster=14383174570134551860&hl=en&as_sdt=0,5&scioldt=0,5.
- [29] Mishra, S., P. Paygude, and S. Chaudhary. 2018. “Use of Data Mining in Crop Yield Prediction.” 2018 2nd International. <https://ieeexplore.ieee.org/abstract/document/8398908/>.
- [30] Modak, A., S. D. Chaudhary, and P. S. Paygude. 2018. “Techniques to Secure Data on Cloud: Docker Swarm or Kubernetes?” 2018 Second. <https://ieeexplore.ieee.org/abstract/document/8473104/>.
- [31] Paygude, P., R. Garg, P. Pathak, A. Trivedi, and A. Raj. n.d. “IMAGE PROCESSING USING MACHINE LEARNING,=” *Academia.edu*. <https://www.academia.edu/download/84231706/IJSDR2009078.pdf>.
- [32] Paygude, P., D. S. D. Joshi, and D. M. Joshi. 2020. “Fault Aware Test Case Prioritization in Regression Testing Using Genetic Algorithm.” *International Journal of Emerging Trends in*. https://scholar.google.ca/scholar?cluster=11372764101768453854&hl=en&as_sdt=0,5&scioldt=0,5.
- [33] Paygude, P., and S. D. Joshi. 2020. “Use of Evolutionary Algorithm in Regression Test Case Prioritization: A Review.” *Proceeding of the International Conference on*. https://link.springer.com/chapter/10.1007/978-3-030-24643-3_6.
- [34] Paygude, Priyanka, and Shashank Joshi. 2020. “Optimization of Test Case Prioritization through Calibration of Genetic Algorithm.” *Solid State Technology* 63 (1s): 2377–86.
- [35] Paygude, Priyanka, and Aatmic Tiwari. 2021. “Comparative Analysis on Different Generations of Cryptocurrency.” *International Journal of Future Generation Communication and Networking* 14 (1): 3016–26.
- [36] Paygude, Priyanka, Aatmic Tiwari, Bhavya Goel, and Akshat Kabra. 2023. “Comparative Analysis of Stock Prices by Regression Analysis and FB Prophet Models.” In *Data Intelligence and Cognitive Informatics*, 295–307. Springer Nature Singapore.
- [37] Priyanka, P. 2019. “Comparative Analysis of Test Case Prioritization Approaches in Regression Testing.” *International Journal of Advanced Trends in Computer*. https://scholar.google.ca/scholar?cluster=11774303718873317097&hl=en&as_sdt=0,5&scioldt=0,5.
- [38] Sachdeva, S., A. Arya, and P. Paygude. 2018. “Prioritizing User Requirements for Agile Software Development.” On *Advances in ...* <https://ieeexplore.ieee.org/abstract/document/8529454/>.
- [39] Tyagi, Shubham, Rishabh Solanki, Adarsh Tiwari, Rohit Ray, and Priyanka Paygude. 2021. “Analysis of Various Machine Learning Models for Detecting Depression in Twitter Tweets.” *Turkish Online Journal of Qualitative Inquiry* 12 (7): 6616–25.
- [40] Varshney, Abhishek, Saumya Asthana, Chetan Sharma, Swapnil Patil, Ankur Kashyap, and Priyanka Paygude. 2020. “Study of Solving Knapsack Problem Using Genetic Algorithm.” *Solid State Technology*, February, 93–99.
- [41] Verma, Rauhil, Priyanka Paygude, Snehal Chaudhary, and Sonali Idate. 2018. “Real Time Traffic Control Using Big Data Analytics.” In *2018 International Conference On Advances in Communication and Computing Technology (ICACCT)*, 637–41.
- [42] Patil, T. B., Kashif Jamal, Keshav Anand, S. T. Sawant-Patil, Aditya Bhateja, and Priyanka Paygude. 2023. “Real-Time Clickstream Data Processing and Visualization Using Apache Tools.” In , 1–5.
- [43] Kadam, A., Garg, B., Gayakwad, M., Kotecha, K., & Joshi, R. (2024). Novel DSIDS-Deep Sniffer Intrusion Detection System. *International Journal of Intelligent Systems and Applications in Engineering*, 12(16s), 400-407.
- [44] Khatik, I., Kadam, S., Gayakwad, M., Joshi, R., & Kotecha, K. (2024). Automatic Diagnosis of Fracture using Deep Learning and External Validation: A Systematic Review and Meta-Analysis. *International Journal of Intelligent Systems and Applications in Engineering*, 12(16s), 41-48.