YOGA POSE CLASSIFICATION AND CORRECTION

PROJECT REPORT

Submitted by

Manjusha Kolli - CH.EN.U4AIE22024

Shankar Mahadev - CH.EN.U4AIE22049

Mounika V - CH.EN.U4AIE22063

KSS Sahithi Athreya - CH.EN.U4AIE22075

BTECH IN COMPUTER SCIENCE ENGINEERING

Under the guidance of
Dr. K Deepak

Submitted to

AMRITA
VISHWA VIDYAPEETHAM
DEEMED TO BE UNIVERSITY UNDER SECTION 3 OF UGC ACT, 1956

AMRITA VISHWA VIDYAPEETHAM
AMRITA SCHOOL OF COMPUTING
CHENNAI – 601103

ABSTARCT:

Yoga has a multitude of physical and mental health benefits, but poor posture during practice can result in injury or diminished effectiveness. Manual oversight is too frequently not available or within reach for the majority of users. In this paper, we introduce a real-time, machine learning-powered Yoga Pose Detection and Correction system to help practitioners maintain accurate posture. The system uses the Mediapipe framework to extract 33 important body landmarks from input video frames or images. The landmarks are taken as input features of a Random Forest classifier, which classifies the performed yoga pose from a pre-defined set of common poses. Model robustness is enhanced by preprocessing operations like rescaling, normalization, and augmentation. The system also identifies deviations from perfect poses and gives corrective feedback, maintaining safety and correct alignment while practicing. This low-cost, lightweight solution fills the gap between self-practice and guided training, making yoga more accessible and precise for users at all skill levels.

Keywords:

Yoga Pose Classification, Pose Correction, Mediapipe, Random Forest Classifier, Human Pose Estimation, Machine Learning, Real-time Feedback, Posture Analysis, Landmark Detection, Image Processing

INTRODUCTION:

Yoga is a globally practiced discipline known for its numerous health benefits, but incorrect postures can lead to inefficiency and potential injuries. Many beginners struggle with maintaining the right form, especially without expert supervision. Traditional methods, such as hiring personal trainers or attending live yoga sessions, can be costly and inaccessible. This project aims to bridge this gap by developing an AI-driven yoga pose correction system that provides real-time feedback, ensuring safe and effective practice.

Incorrect yoga postures can lead to injury risks, reduced exercise effectiveness, and limited accessibility to expert feedback, particularly for individuals practicing at home. Ensuring correct posture is crucial for preventing injuries, maximizing benefits, and enhancing the overall yoga experience. An AI-based correction system provides automated guidance without requiring a human instructor, delivers real-time feedback for instant corrections, and makes yoga coaching cost-effective and widely accessible.

To achieve this, the project leverages computer vision and deep learning to analyze and correct yoga postures. Using Mediapipe Pose Estimation, we extract 33 key body landmarks from user images or videos. A deep-learning model then assesses posture deviations by comparing these landmark positions with ideal yoga postures. Based on the analysis, the system provides instant recommendations, visually highlighting errors and guiding users on necessary adjustments. The feedback is delivered through an intuitive interface, making corrections clear and actionable. By integrating pose estimation, deep learning, and real-time feedback, this project offers a scalable, affordable, and intelligent solution for improving yoga posture, helping users practice yoga safely and effectively.

RELATED WORK:

Bansal and Sharma [1] introduced DeepYoga, a deep learning-based system that relies on CNNs trained on body landmarks to precisely classify five yoga poses. The model gives real-time feedback for posture correction and achieved 99.02% accuracy, providing an easy solution for safe, expert-free yoga practice.Shah and Gupta [2] suggested a real-time yoga pose detection and correction system based on Explainable Artificial Intelligence (XAI) to provide interpretable feedback. Their system detects significant body landmarks and offers real-time correction guidance for misalignments, improving both safety and effectiveness in virtual yoga classes.

Gurav et al. [3] applied OpenPose to yoga pose detection and correction using part affinity fields and a multi-CNN architecture to localize anatomical keypoints. Their method efficiently detects body parts and their relations, making accurate human pose estimation possible for yoga purposes.Akash et al. [4] proposed a yoga pose classification method based on transfer learning on the Yoga-82 dataset, which consists of 82 difficult pose classes. Through fine-tuning DenseNet-121, VGG-16, and ResNet variants.

Kou and Li [5] presented an improved yoga pose recognition technique employing ensemble learning along with a multi-head attention method to improve the accuracy and resistance. With data augmentation by using Mixup, features of ResNet101, and VGGNet19 as input, they obtained a high testing accuracy of 93.79% as well as 297 FPS-fast detection speed.Hande et al. [6] put forward a correction system for yoga postures via OpenCV and VGG-19 based on GPU-based transfer learning to achieve higher accuracy for home-based workouts.

Yang [7] introduced a collaborative network that integrates pose classification and scoring through residual block-based Siamese CNN. Utilizing skeleton images from OpenPose, the model efficiently offers pose type recognition. Jadhav et al. [8] created a CNN-based yoga pose correction system with OpenCV integration for real-time feedback and feature extraction. The system is safety- and accessibility-focused, offering corrective alignment cues appropriate for beginners and experienced practitioners alike.

DATASET:

This data set consists of a nicely balanced collection of pictures for six widely recognized yoga postures. The data has been pre-processed by removing PNG files (which caused PyTorch to display warnings) and low-resolution or wrongly classified photos. This dataset is segregated into separate train and test data, making it very suitable for the use in classification tasks through deep learning.

Dataset Features

- Yoga Poses Included:

  - Downdog
  - Plank
  - Side Plank
  - Warrior
  - Tree
  - Goddess

- File Format: All images are in JPG format
- Data Split: Split into training and test sets for supervised learning
- Applications: Applicable to image classification, pose recognition, and AI yoga research

This dataset is a good starting point for our yoga pose detection and correction project as it provides well-labeled images of important yoga poses. We can utilize it to:

1. Train a Classification Model – Correctly classify various yoga poses with CNN-based architecture.
2. Improve Pose Estimation – Refine models such as OpenPose or MediaPipe using this dataset for training and validation.
3. Help in Pose Correction – By combining classification with pose keypoint analysis, we can assist in real-time correction of misplaced postures.

Using this cleaned and formatted dataset, we can improve the accuracy of our pose detection model while providing dependable results in yoga applications.

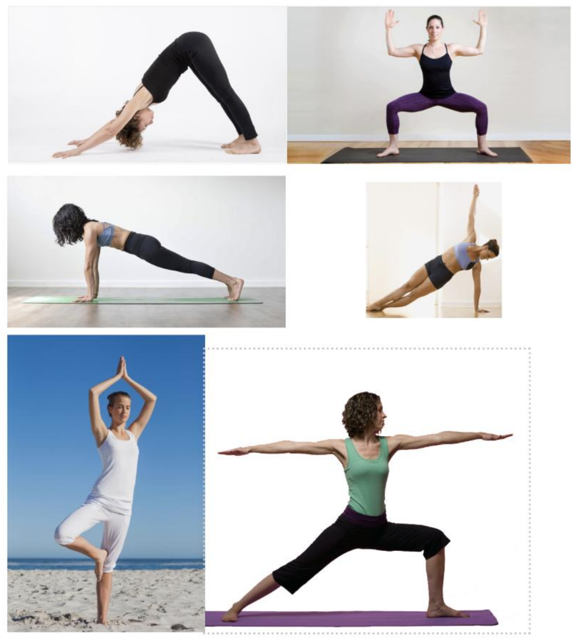


FIG 01: DATASET POSES

METHODOLOGY:

The system should be able to automatically detect the presence of each yoga pose within an image. This is possible by extracting media keypoints from the image through the use of MediaPipe, numerically encoding the key joint coordinates, and fitting a Random Forest classifier on that data.

1. MediaPipe-based feature extraction:

To categorize yoga poses, we initially had to get meaningful body posture information from images. We employed MediaPipe Pose, which identifies 33 significant human body joints (such as shoulders, elbows, hips, knees, etc.).

Every joint is represented by 3 values:

- x: Horizontal location (normalized)
- y: Vertical location (normalized)
- z: Depth value (relative)

This results in 33 × 3 = 99 values per image, forming a fixed-length feature vector representing the pose.

2. Labeling the Data:

Each image is categorized into one of the following yoga poses such as: Tree ,Downdog , Goddess , Plank , Side Plank ,Warrior

The folder name of the image is used as the label (target class) during model training.

3. Dataset Preparation:

1. All the image vectors and their labels are combined into a dataset:
2. The 99 pose features constitute the X (input) matrix
3. The pose labels ("plank", "tree") constitute the y (target) vector
4. The dataset is subsequently:

- Encoded (labels in numeric form)
- Divided into training and test sets (most often 80% training, 20% testing)
- Shuffled and stratified to maintain class balance

4. Classification Using Random Forest:

To train the classifier, we selected a Random Forest model, which is an effective and interpretable ensemble learning method.

- Works well with high-dimensional data (such as 99 features)
- Robust to overfitting
- Gives class probabilities (confidence)
- Handles nonlinear relationships between joints and classes

5. Hyperparameter Optimization:

1. Grid Search Cross-Validation was used to discover the optimal parameters for the Random Forest:

2. Parameters tuned included:
- Number of trees (n_estimators)
- Tree depth (max_depth)
- Minimum samples to split/leaf

Cross-validation helps ensure that the model generalizes well by testing various combinations of parameters on many splits of the training set.

6. Pose Prediction on New Images:

Once trained, the model can predict the pose class for any new image:

- The image is processed by MediaPipe to extract joint landmarks
- The 99-dimension feature vector is constructed
- The trained Random Forest makes the prediction of the pose class

Output consists of:

- Predicted pose (e.g., "Tree")
- Confidence percentage
- Overlayed pose skeleton for visualization

Pose Correction:

The goal of the pose correction system is to compare a user's yoga pose with a reference pose and return:

- A percentage score showing how close it is
- Human-readable, detailed feedback (e.g., "Lift your arm higher")
- Visual side-by-side comparison for easier understanding

This component of the system is driven by MediaPipe landmark detection, angle-based geometric analysis, and rule-based feedback generation.

1. Pose Landmark Detection (MediaPipe)

We first detected 33 anatomical landmarks on the body via the MediaPipe Pose Landmarker in order to analyze posture.

It Returns:

- 3D coordinates (x, y, z) for every joint
- Normalized positions (between 0 and 1) in image space
- Functions on both reference images and user images

Landmarks used particularly for correction:

- Left/Right Elbow
- Left/Right Shoulder
- Left/Right Hip
- Left/Right Knee

These joints play a vital role in the majority of yoga poses and have a great impact on posture.

2. Calculation of Joint Angle

After we have the joint coordinates, we compute the angle between three joints by using trigonometric vector mathematics.

Example, To calculate the left elbow angle, we use:

- A = left shoulder
- B = left elbow (vertex)
- C = left wrist

We apply the dot product formula:

This leaves us with the angular posture of every body joint — a value between 0° and 180°.

3. Pose Comparison:

1. We now have angles from: A correct/reference pose, user pose
2. We subtract them joint-by-joint to obtain the angular deviation:

- Small deviation → good alignment

- Large deviation → incorrect posture

4. Accuracy Scoring:

To make it easier to interpret, we convert angle deviations into accuracy scores using threshold logic:

- This scoring is calculated per joint
- Overall pose score is the mean of all joint scores

5. Feedback Generation

Based on the direction and magnitude of error, we provide actionable feedback for every body part.

Example Rules:

- If the angle of the elbow is too small → "Increase angle → Raise your arm higher"
- If the angle of the hip is too large → "Decrease angle → Lower your hip"
- If in ±5°, then we say: "☑ Good alignment!"

The reasoning is rule-based (if-else), but translates the geometric outcome into natural language.

6. Visual Feedback

To help users visually understand corrections

We display side-by-side images:

- Left: Reference pose
- Right: User pose

Each pose is overlaid with:

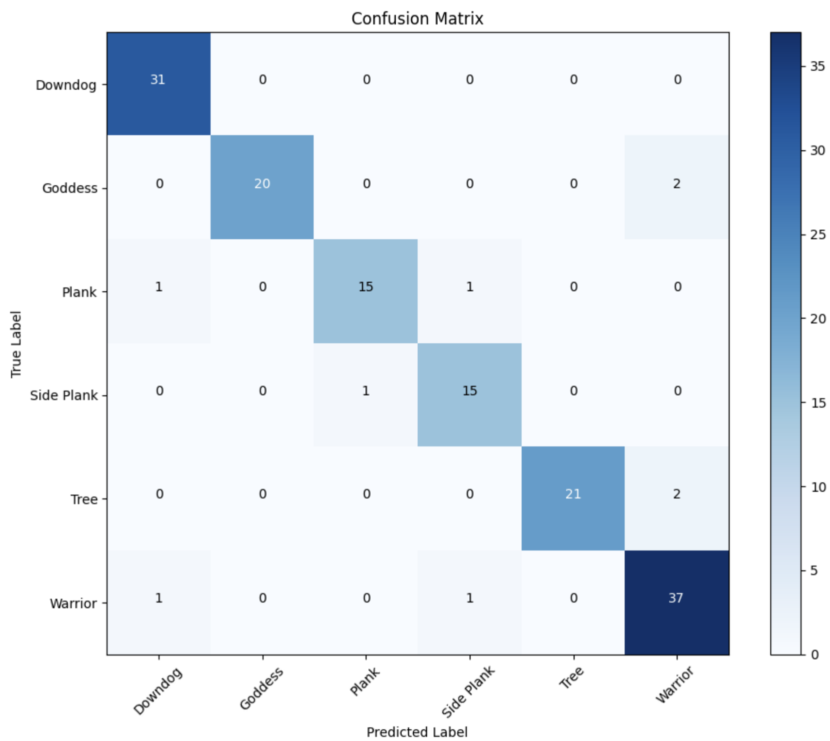
- Stickman skeleton
- Joint numbers
- Angle values

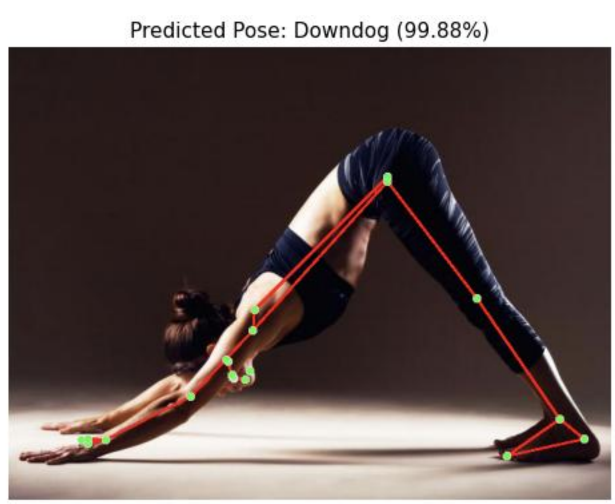
Colors and labels show:

- Correct joints
- Misaligned joints

RESULTS:

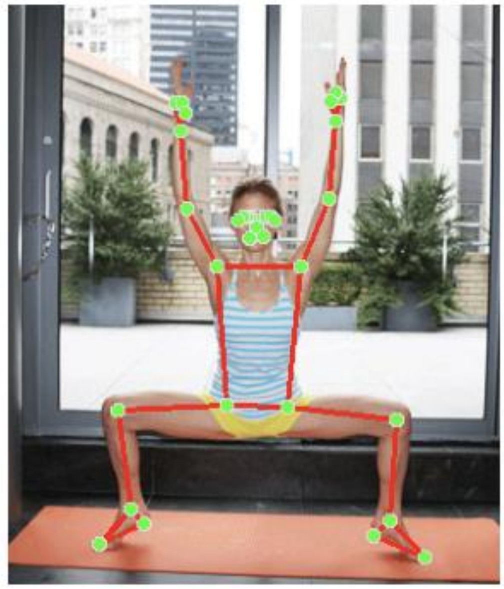
Yoga pose detection:





Predicted Pose: Tree (94.33%)

Predicted Pose: Goddess (90.33%)



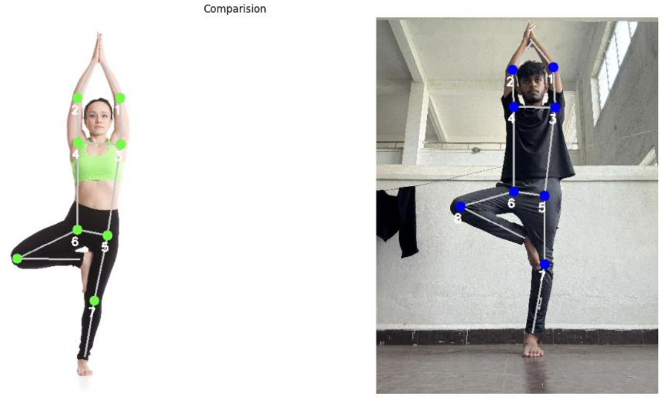
Predicted Pose: Plank (86.87%)

```
Classification Accuracy: 93.92%

Classification Report:
              precision    recall  f1-score   support

           0       0.94      1.00      0.97        31
           1       1.00      0.91      0.95        22
           2       0.94      0.88      0.91        17
           3       0.88      0.94      0.91        16
           4       1.00      0.91      0.95        23
           5       0.90      0.95      0.92        39

    accuracy                           0.94       148
   macro avg       0.94      0.93      0.94       148
weighted avg       0.94      0.94      0.94       148
```

Yoga pose correction:

```
🔄 Processing Pose Pair 1...
[[-42.18438888 -23.55330507 -40.36082187 -48.4340831  -13.02224886
   17.75201507   4.45472171   2.57843777]]
[[-42.18438888 -23.55330507 -40.36082187 -48.4340831  -13.02224886
   17.75201507   4.45472171   2.57843777]]
🟢 Pose Accuracy: 60.84%
```

Comparision

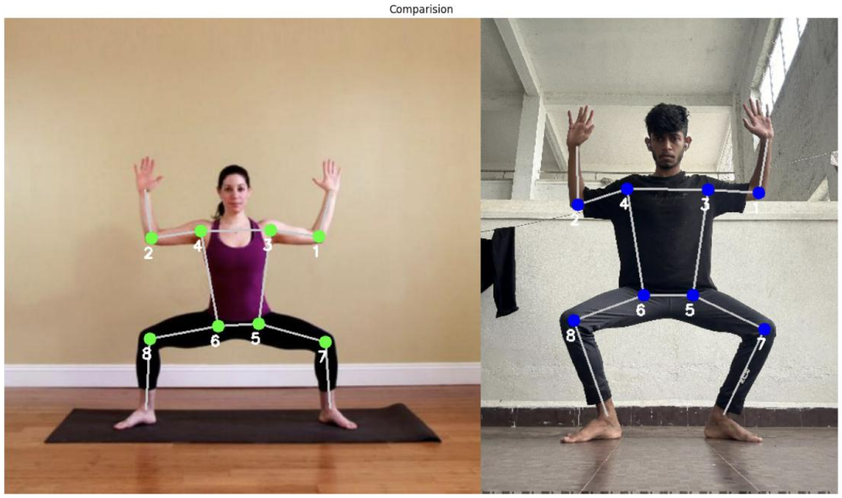


	Label	Body Part	Correct Angle	User Angle	Correction Needed
0	1	Left Elbow	154.72	143.82	10.90° → ↑ Increase
1	2	Right Elbow	153.52	144.85	8.67° → ↑ Increase
2	3	Left Shoulder	171.22	170.58	0.64° → ↑ Increase
3	4	Right Shoulder	178.80	178.84	0.04° → ↓ Decrease
4	5	Left Hip	176.73	172.95	3.78° → ↑ Increase
5	6	Right Hip	116.33	106.16	10.17° → ↑ Increase
6	7	Left Knee	178.52	169.84	8.68° → ↑ Increase
7	8	Right Knee	25.97	42.15	16.17° → ↓ Decrease

```
🔄 Processing Pose Pair 2
```

```
🔄 Processing Pose Pair 6...
[[ 29.53515536  11.60672568  23.61695959  22.39467813 -21.09597397
  -26.79670382   7.53136884   6.15715071]]
[[ 29.53515536  11.60672568  23.61695959  22.39467813 -21.09597397
  -26.79670382   7.53136884   6.15715071]]
🟢 Pose Accuracy: 76.55%
```

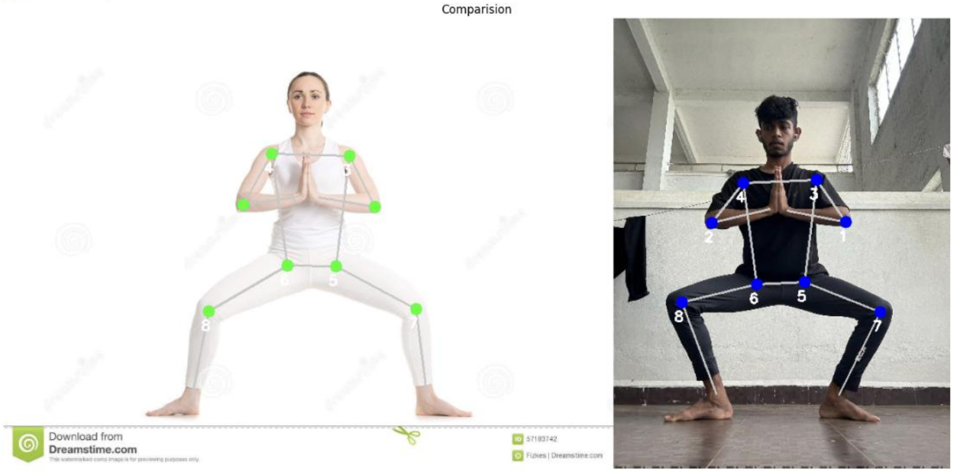
Comparision



	Label	Body Part	Correct Angle	User Angle	Correction Needed
0	1	Left Elbow	97.07	95.64	1.43° → ↑ Increase
1	2	Right Elbow	89.50	72.17	17.33° → ↑ Increase
2	3	Left Shoulder	90.06	94.87	4.82° → ↓ Decrease
3	4	Right Shoulder	92.26	80.60	11.66° → ↑ Increase
4	5	Left Hip	97.64	107.39	9.75° → ↓ Decrease
5	6	Right Hip	90.31	101.25	10.94° → ↓ Decrease
6	7	Left Knee	109.24	90.20	19.04° → ↑ Increase
7	8	Right Knee	103.52	91.15	12.37° → ↑ Increase

```
🔄 Processing Pose Pair 7...
[[ -5.74008018  -3.35766892  20.4832906   15.53966207 -14.59141938
  -14.28060954 -27.06361488 -22.77774858]]
[[ -5.74008018  -3.35766892  20.4832906   15.53966207 -14.59141938
  -14.28060954 -27.06361488 -22.77774858]]
🟢 Pose Accuracy: 82.89%
```

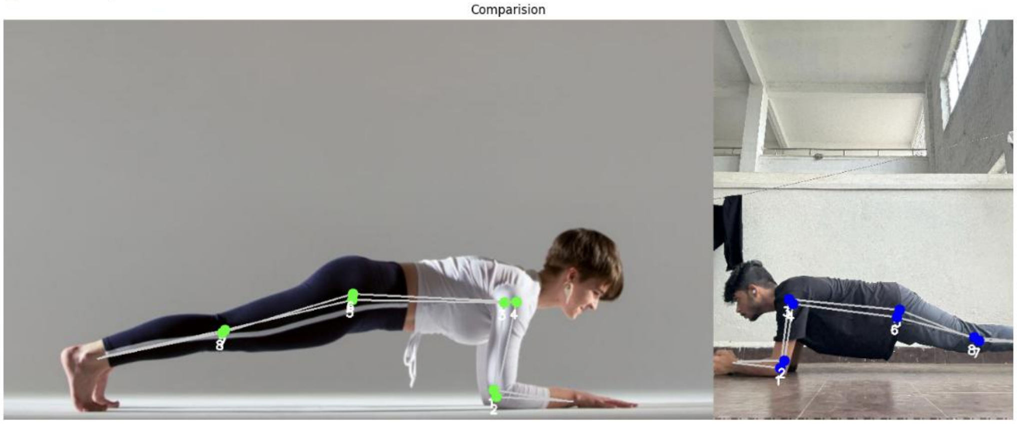
Comparision



	Label	Body Part	Correct Angle	User Angle	Correction Needed
0	1	Left Elbow	54.01	43.79	10.22° → ↑ Increase
1	2	Right Elbow	52.21	38.45	13.76° → ↑ Increase
2	3	Left Shoulder	33.39	42.06	8.67° → ↓ Decrease
3	4	Right Shoulder	37.33	45.95	8.62° → ↓ Decrease
4	5	Left Hip	110.84	104.86	5.98° → ↑ Increase
5	6	Right Hip	112.07	95.86	16.21° → ↑ Increase
6	7	Left Knee	125.03	84.73	40.30° → ↑ Increase
7	8	Right Knee	130.71	82.83	47.88° → ↑ Increase

```
🔄 Processing Pose Pair 9...
[[-14.14497906 -10.61472601  13.17229403   4.75566006  26.87607881
   -0.0969358   -1.86771869 -13.69695597]]
[[-14.14497906 -10.61472601  13.17229403   4.75566006  26.87607881
   -0.0969358   -1.86771869 -13.69695597]]
🟢 Pose Accuracy: 91.09%
```

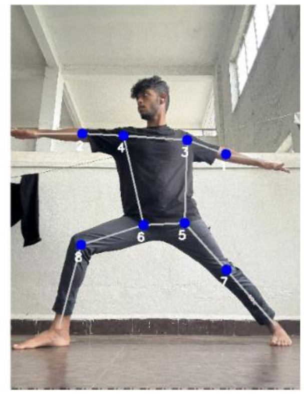
Comparision



	Label	Body Part	Correct Angle	User Angle	Correction Needed
0	1	Left Elbow	91.69	91.36	0.33° → ↑ Increase
1	2	Right Elbow	81.77	99.12	17.34° → ↓ Decrease
2	3	Left Shoulder	83.75	91.96	8.21° → ↓ Decrease
3	4	Right Shoulder	81.00	90.89	9.89° → ↓ Decrease
4	5	Left Hip	165.69	164.11	1.58° → ↑ Increase
5	6	Right Hip	160.84	172.86	12.02° → ↓ Decrease
6	7	Left Knee	179.04	160.49	18.55° → ↑ Increase
7	8	Right Knee	175.37	170.91	4.46° → ↑ Increase

```
🔄 Processing Pose Pair 10...
[[ -4.79679737  14.15630204  -4.6342404   25.24945152 -30.58841424
  -31.63392402 -15.2159133   -6.88876962]]
[[ -4.79679737  14.15630204  -4.6342404   25.24945152 -30.58841424
  -31.63392402 -15.2159133   -6.88876962]]
🟢 Pose Accuracy: 79.11%
```

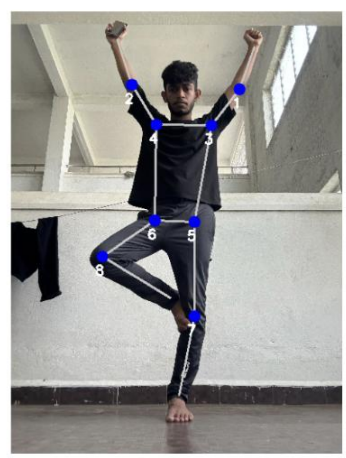
Comparision



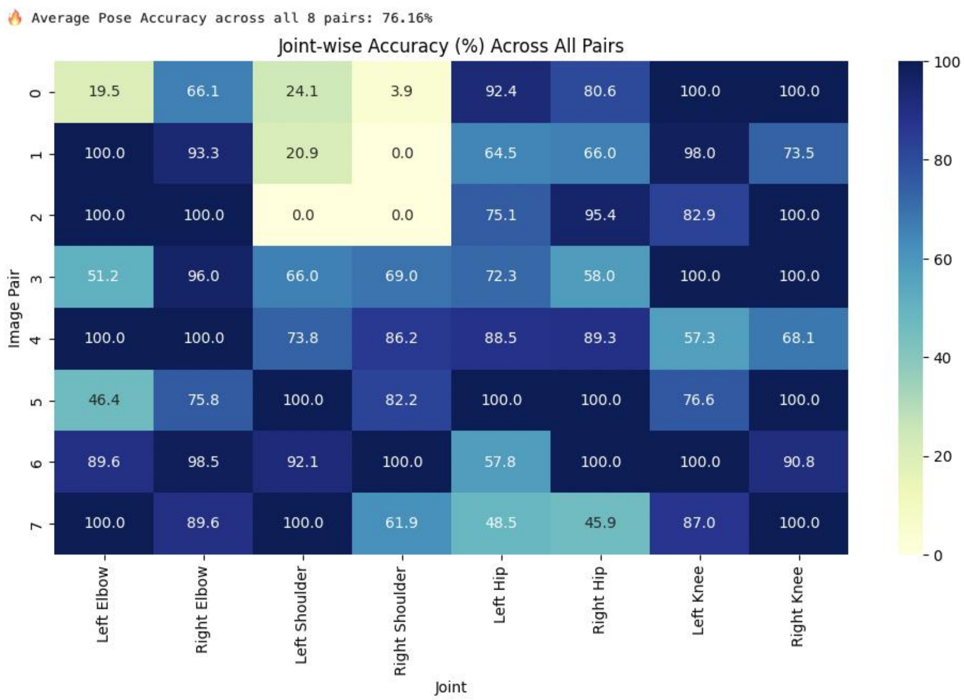
	Label	Body Part	Correct Angle	User Angle	Correction Needed
0	1	Left Elbow	166.09	169.62	3.54° → ↓ Decrease
1	2	Right Elbow	173.24	178.39	5.15° → ↓ Decrease
2	3	Left Shoulder	99.26	71.81	27.44° → ↑ Increase
3	4	Right Shoulder	95.18	104.87	9.69° → ↓ Decrease
4	5	Left Hip	118.52	136.86	18.34° → ↓ Decrease
5	6	Right Hip	96.36	95.34	1.02° → ↑ Increase
6	7	Left Knee	175.17	179.83	4.65° → ↓ Decrease
7	8	Right Knee	112.53	122.66	10.13° → ↓ Decrease

```
🔄 Processing Pose Pair 5...
[[  0.51053655   6.44949708 -57.09513226 -63.70051634 -19.95801672
   11.84420247 -16.85239943   2.51334354]]
[[  0.51053655   6.44949708 -57.09513226 -63.70051634 -19.95801672
   11.84420247 -16.85239943   2.51334354]]
🟢 Pose Accuracy: 69.17%
```

Comparision



	Label	Body Part	Correct Angle	User Angle	Correction Needed
0	1	Left Elbow	154.72	160.31	5.58° → ↓ Decrease
1	2	Right Elbow	153.52	166.86	13.33° → ↓ Decrease
2	3	Left Shoulder	171.22	151.73	19.49° → ↑ Increase
3	4	Right Shoulder	178.80	147.51	31.29° → ↑ Increase
4	5	Left Hip	176.73	169.64	7.09° → ↑ Increase
5	6	Right Hip	116.33	125.29	8.95° → ↓ Decrease
6	7	Left Knee	178.52	168.70	9.82° → ↑ Increase
7	8	Right Knee	25.97	65.77	39.80° → ↓ Decrease



CONCLUSION:

This project introduces a real-world and feasible solution for computer vision-based yoga pose classification and correction. With the use of Mediapipe for fast and real-time pose landmark detection, and Random Forest for pose classification, the system effectively detects important yoga postures and offers informative corrective feedback. The combination of preprocessing and data augmentation methods increases the model's robustness and generalizability to different lighting and camera conditions. This method avoids the requirement for costly equipment or ongoing instructor supervision, allowing users—particularly novices—to practice yoga effectively and safely. Overall, the system helps to encourage more healthy lifestyles using technology-assisted posture guidance and provides a basis for further enhancements such as support for more advanced poses, customized feedback, and real-time deployment via mobile or web apps.

REFERENCE:

[1] Bansal, Roli, et al. "DeepYoga: Enhancing Practice with a Real-Time Yoga Pose Recognition System." Engineering, Technology & Applied Science Research 14.6 (2024): 17704-17710.

[2] Alhumam, Abdulaziz. "Explainable Software Fault Localization Model: From Blackbox to Whitebox." Computers, Materials & Continua 73.1 (2022).

[3] O. Gurav, U. Kulthe, S. Bhalerao, M. Lawande, and N. Mali, "Yoga Pose Detection and Correction Using OpenPose," International Journal for Research in Applied Science and Engineering Technology

[4] Akash, M. M., et al. "Yoga Pose Classification Using Transfer Learning." arXiv preprint arXiv:2411.00833 (2024).

[5] Kou, Y., Li, H. Image-Based Fitness Yoga Pose Recognition: Using Ensemble Learning and Multi-head Attention. Int J Comput Intell Syst 17, 257 (2024).

[6] Hande, T., et al. "Yoga Postures Correction and Estimation using Open CV and VGG 19 Architecture." International Journal of Innovative Science and Research Technology 8.4 (2023).

[7] Yang, Wenxi. "A Joint Network Based CNN for Yoga Pose Classification and Scoring."

[8] N. Jadhav, M. Kulkarni, S. Kalyani, N. Hiwale, and B. Bobade, “Yoga Pose Correction System Using CNN,” *IRJMETS*, doi: 10.56726/IRJMETS66782.