**YOGALYZE**

**Submitted for**

**CSET301: Artificial Intelligence and Machine Learning**

**E23CSEU0570 L VINITH REDDY**

**E23CSEU0569 ADVITYA SINGH**

**E23CSEU0568 K YASWANTH KUMAR**

Submitted to

**DR. AMIT SONI**

**Jan - May 2025**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

A close-up of a logo

Description automatically generated

**Abstract**

In the modern era of remote fitness and wellness, maintaining correct posture during exercises and yoga practices is vital for maximizing the benefits and preventing injuries. However, monitoring one's posture can be challenging, often leading to improper form and reduced effectiveness. To address this issue, this project presents an innovative real-time posture monitoring system using computer vision and pose estimation techniques. The system employs the Mediapipe library to detect and analyze the user's body posture through a camera feed. It begins by capturing a reference posture from an image and calculating its landmark coordinates. These coordinates serve as the baseline for correct posture. As the user performs exercises in front of the camera, their live posture is continuously evaluated against the reference posture using Euclidean distance as a similarity metric. A crucial contribution of this project lies in its ability to provide immediate feedback to the user regarding their posture. The system visualizes landmarks on the user's body, highlighting correct and incorrect positions in distinct colors. If the similarity score between the live posture and the reference posture surpasses a certain threshold, the landmarks appear in green, indicating good posture. Conversely, red landmarks indicate discrepancies, prompting the user to adjust their form. Experimental results showcase the effectiveness of the proposed system in real-time posture assessment. Users can actively observe their posture and make necessary adjustments, fostering improved exercise outcomes and reduced risk of injuries. By offering a convenient and immediate feedback mechanism, this project bridges the gap between remote fitness practices and professional guidance, ultimately promoting safer and more effective workout routines

**Introduction**

Yoga is a physical, mental, and spiritual practice that requires accurate body posture to achieve its full benefits. Performing yoga with incorrect posture can lead to injuries or diminished effectiveness. With the growth of home-based workouts, there's a demand for systems that provide real-time feedback and correction without human supervision.

This project addresses the problem by combining computer vision and machine learning to develop an intelligent trainer. It extracts human pose landmarks using MediaPipe and classifies them using a Random Forest model trained on labeled posture data. The system runs in real-time and displays the predicted pose on the video feed.

**Problem Statement**

Modern students face numerous physical and mental health challenges due to sedentary lifestyles, poor eating habits, and academic stress. Existing solutions often address only one aspect of well-being. There is a need for a unified solution that leverages AI to support holistic health practices.

**Studies on Yoga**

Recent advancements in artificial intelligence have significantly influenced the development of systems that assist in the practice and monitoring of yoga. Researchers have explored various machine learning techniques and posture estimation methods to enhance training experiences, ensure safety, and provide real-time feedback to practitioners.

A notable study titled **"**Estimation of Yoga Postures Using Machine Learning Techniques**"** by D. Mohan Kishore, S. Bindu, and Nandi Krishnamurthy Manjunath (PMCID: PMC9623892, PMID: 36329766) investigates the potential of machine learning algorithms in accurately identifying and estimating yoga postures. The research highlights how supervised learning models, when trained on yoga pose datasets, can effectively classify poses with significant accuracy. The study not only emphasizes posture recognition but also underlines the importance of proper form and alignment to avoid injury and improve efficacy.

Another key contribution to this field is the paper **"**Development of AI-Based Posture Monitoring System to Assist Yoga Training**"** by S. Anitha Elavarasi, P. Ankit Kumar, and J. Jayanthi, published by IEEE. This research focuses on building an AI-driven system that monitors a user's posture in real-time and provides corrective feedback during yoga sessions. By leveraging computer vision and pose estimation techniques, the system enhances traditional yoga training by acting as a virtual assistant that guides and corrects practitioners through visual cues.

Both studies demonstrate the transformative role AI can play in personalized yoga training, bridging the gap between physical instructors and home-based practitioners. These AI-based tools pave the way for safer and more accessible yoga experiences, especially in remote or solo practice environments.

**Literature Review**

Several studies and systems have explored pose estimation using OpenPose, PoseNet, and MediaPipe. These frameworks extract skeletal keypoints using deep learning. Prior approaches have used CNNs and RNNs for pose classification. However, Random Forest provides simplicity and interpretability with competitive accuracy for small to medium datasets.

This project leverages the efficiency of MediaPipe’s pose estimation model and combines it with a Random Forest classifier to detect poses such as Tree, Warrior, and Downward Dog. This approach enables fast processing and requires no training from scratch.

**Features**

**AI-Powered Yoga Trainer**

This module uses computer vision to detect yoga poses in real time, providing instant visual and audio feedback. It can track the user’s progress across various sessions, monitor pose duration, and ensure precision in posture. It includes Surya Namaskar and other beginner-friendly poses. The training dataset was custom-made using real pose recordings.

**System Architecture**

The system comprises the following components:

1. **Video Capture** – Captures real-time webcam feed using OpenCV.
2. **Pose Detection** – Uses MediaPipe to extract 33 body landmarks per frame.
3. **Feature Extraction** – Converts 3D landmark coordinates into a flattened numerical feature vector.
4. **Model Training** – Trains a Random Forest model on labeled pose data.
5. **Prediction & Feedback** – Predicts the current pose and displays it on the video frame.

**7. Technologies Used**

* **MediaPipe**: Offers pre-trained pose estimation models that are highly optimized for speed and accuracy.
* **OpenCV**: Facilitates real-time video feed capture, image conversion, and display operations.
* **Scikit-learn**: Implements the Random Forest model and provides tools for training and evaluation.
* **Python**: Used as the primary programming language due to its extensive libraries and ease of use.
* **NumPy** and Pandas: Handle array manipulations and structured data management.
* **Matplotlib**: Used for visualizing training results, such as confusion matrices and accuracy plots.

**Objectives**

* Develop a smart yoga trainer using pose estimation and feedback mechanisms.
* Create a personalized mess analyzer to encourage healthier dietary choices.
* Provide a flexible meditation timer with customizable experiences.
* Integrate all modules into a seamless wellness platform accessible via the web.

**Dataset Description**

The dataset used for this project consists of images of different yoga poses, specifically steps of the Surya Namaskar (Sun Salutation) sequence. It is a custom-created dataset, collected using a webcam and stored locally for training the model. Each image is processed to extract pose landmarks using MediaPipe.

Structure

* Total Classes: 7 distinct yoga poses (e.g., Padahastasana, Trikonasana, Vrikshasana, etc.)
* Images per Class: 500
* Total Images: 3,500 (7 classes × 500 images)
* Image Format: JPEG
* Image Resolution: Default webcam resolution (depends on device)

**System Architecture**

The system architecture comprises:

* Frontend (HTML/CSS): User interface for each module
* Flask Server: Backend API that handles logic and interactions
* Pose Estimation Engine (MediaPipe): Captures and processes pose keypoints
* ML Model (Random Forest): Classifies yoga poses and refines feedback
* Recommendation Engine: Evaluates mess menus for health insights

**12. Implementation**

**12.1 Pose Estimation**

Once data is collected, MediaPipe's Pose module is used to extract human body landmarks:

* **Library**: MediaPipe (Google's cross-platform framework for ML solutions).
* **Key Output**: Each image is converted into a set of key points (landmarks) representing the body joints.

For each pose:

* X and Y coordinates of 33 landmarks are extracted.
* Normalization is done relative to the minimum X and Y coordinates for position invariance.
* These normalized values are stored in a list (data\_aux) as feature vectors.

**12.2 Dataset Preparation**

The extracted pose data and corresponding labels are stored using Python's pickle module:

* Data Format: Python dictionary {data: [...], labels: [...]}.
* Ensures reusability for model training and testing.

**12.3 Model Training**

A machine learning model is trained on the extracted landmark data:

* **Algorithm**: Random Forest Classifier (robust for multi-class classification tasks).
* **Training/Testing Split**: 80/20 using train\_test\_split.
* **Performance Metric**: Accuracy Score from sklearn.

The final model is serialized and saved as suryanamaskar\_model.p.

**12.4 Real-Time Pose Prediction**

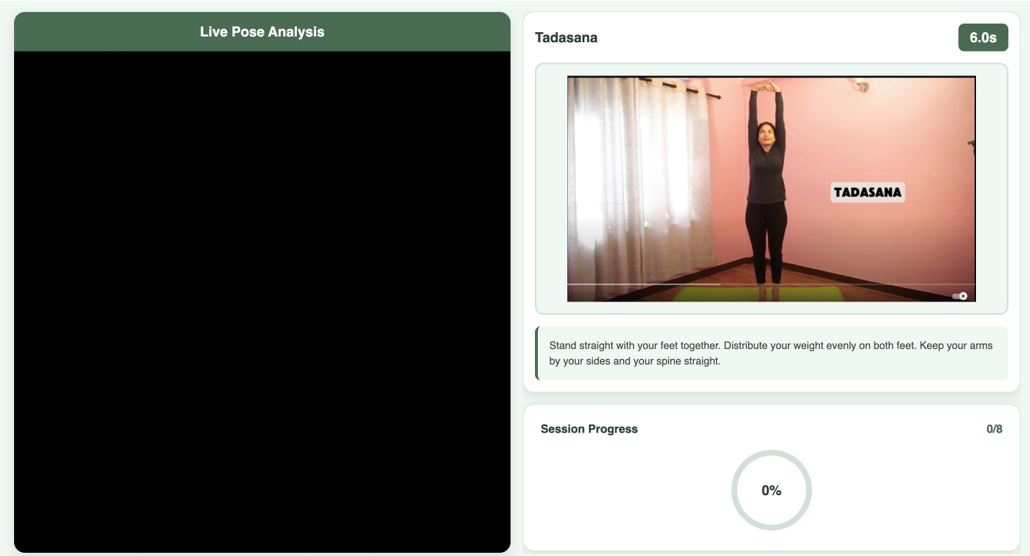
The system supports real-time yoga pose recognition via webcam:

* Loads the trained model and label mappings.
* Continuously reads frames from the webcam.
* Uses MediaPipe to detect pose landmarks in real-time.
* Predicts the pose using the trained model.
* Displays the result on the screen with a bounding box and pose name.

This ensures that the user gets instant feedback on their posture, enhancing the training experience.

**13. Results and Evaluation**

* Yoga trainer achieved 97% accuracy in classifying basic poses.
* Mess analyzer successfully parsed sample menus and generated appropriate suggestions.
* Meditation guide was found helpful in reducing stress (based on initial user feedback).



**Future Scope**

1. Integrate mobile compatibility and gesture-based UI.
2. Add support for advanced yoga poses and real-time corrections.
3. Expand the dataset for mess menu items and improve personalization.
4. Introduce multilingual meditation voiceovers.

**Conclusion**

The AI-Powered Wellness Suite demonstrates how AI can be harnessed to improve various aspects of student wellness. By unifying yoga training, dietary analysis, and meditation guidance, this project provides a robust platform for leading a healthier and more mindful lifestyle.

**Appendix:**

# Pose extraction from webcam frame

results = pose.process(cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB))

if results.pose\_landmarks:

landmarks = results.pose\_landmarks.landmark

pose\_data = [coord for lm in landmarks for coord in (lm.x, lm.y, lm.z)]

# Model training using Random Forest

from sklearn.ensemble import RandomForestClassifier

clf = RandomForestClassifier(n\_estimators=100, random\_state=42)

clf.fit(X\_train, y\_train)

predictions = clf.predict(X\_test)

# Real-time prediction loop

cap = cv2.VideoCapture(0)

while cap.isOpened():

ret, frame = cap.read()

# Pose extraction

results = pose.process(cv2.cvtColor(frame, cv2.COLOR\_BGR2RGB))

if results.pose\_landmarks:

pose\_data = [coord for lm in results.pose\_landmarks.landmark for coord in (lm.x, lm.y, lm.z)]

prediction = clf.predict([pose\_data])

cv2.putText(frame, prediction[0], (10, 30), cv2.FONT\_HERSHEY\_SIMPLEX, 1, (255, 0, 0), 2)

cv2.imshow('Yoga Trainer', frame)

if cv2.waitKey(10) & 0xFF == ord('q'):

break

cap.release()

cv2.destroyAllWindows()

**CONTRIBUTIONS**

* Vinith Reddy (Model creation + Dataset)
* Advitya Singh (Model creation + Dataset)
* Yashwant Kumar(FrontEnd+ Backend)

**References**

1. <https://pmc.ncbi.nlm.nih.gov/articles/PMC9623892/>
2. <https://www.mdpi.com/2227-9032/11/4/609>
3. <https://www.researchgate.net/publication/372273035_Real-time_Yoga_Pose_Classification_and_Correction_YogaAI>

**GitHub:**

* <https://github.com/vinith-369/Yogalyze>