

Real-Time Posture Analysis and Personalized Nutrition Planning using Computer Vision System

1. Soham Balekundri
Dept. of ISE
BMSCE
Bengaluru, Karnataka

2. Soumil Kumar
Dept. of ISE
BMSCE
Bengaluru, Karnataka

3. Dr. M V Sudhamani Professor
Dept. of ISE
BMSCE
Bengaluru, Karnataka

4. Surag Y S
Dept. of ISE
BMSCE
Bengaluru, Karnataka

Abstract— This work details the development and features of an innovative fitness application designed to enhance user health outcomes by holistically addressing exercise posture and nutrition. The system leverages computer vision, specifically MediaPipe, for real-time analysis and correction of exercise form across a variety of workouts, aiming to optimize performance and minimize injury risk. Complementing this, machine learning algorithms generate personalized diet plans tailored to individual user profiles and goals, further supported by an integrated chatbot for dietary recommendations. Initially offering six distinct exercise feeds, the application has evolved to include a comprehensive, categorized menu of exercises targeting specific body parts. Furthermore, the platform now incorporates robust tracking capabilities for metrics such as calorie intake, weight fluctuations, and step counts, all visualized through an intuitive results window featuring progress graphs. It is important to note that the evaluation of the system's performance included data collected from two different samples, representing different fitness levels and exercise experience. This integrated approach empowers users to effectively achieve their fitness objectives by providing comprehensive guidance, personalized support, and data-driven insights.

I. INTRODUCTION

The rise of sedentary lifestyles and improper exercise practices has contributed significantly to a global decline in physical health, resulting in issues such as poor posture, muscular imbalances, and preventable injuries. While fitness tracking applications and wearable devices have gained popularity, they often focus solely on metrics such as step count, heart rate, and calorie expenditure. These systems typically lack the capacity to assess and correct exercise form or provide personalized nutritional guidance, both of which are critical to achieving holistic health outcomes.

Smart Fit is a novel AI-powered fitness platform that addresses these shortcomings through the integration of real-time posture correction and machine learning-driven diet planning. Using computer vision models like MediaPipe for human pose estimation, Smart Fit analyses the user's body alignment during workouts and provides instant feedback to improve posture. This minimizes the risk of injury and enhances workout efficiency by ensuring that exercises are performed with proper technique.

In addition to posture correction, the system offers personalized nutrition planning through a machine learning

pipeline that includes clustering and classification algorithms. By taking user-specific parameters such as body mass index (BMI), activity level, and fitness goals, Smart Fit generates individualized diet plans and dynamically adjusts them as user progress is tracked.

By combining posture monitoring, personalized diet generation, and resource discovery into a single application, Smart Fit proposes a comprehensive solution to promote long-term health and wellness. Unlike traditional fitness apps that rely heavily on manual input and static routines, Smart Fit leverages automation and adaptivity to cater to individual needs in real-time.

This paper will further elaborate on the methodologies employed for its core functionalities, the system architecture, the results of its implementation, and potential avenues for future enhancements.

II. RELATED WORK

Recent advances in computer vision and machine learning have enabled intelligent fitness systems for monitoring exercise form and providing personalized recommendations. Several studies have explored pose estimation models like OpenPose, MediaPipe, and BlazePose for real-time analysis of human body alignment. One application achieved up to 89% accuracy in posture assessment and repetition counting using BlazePose and a Random Forest classifier [1]. Another proposed a fitness trainer providing feedback by comparing real-time posture against ideal poses [2]. Our work also uses MediaPipe's BlazePose for enhanced posture correction.

Machine learning has also been applied to personalized nutrition systems based on user parameters. A mobile-based fitness predictor using k-means clustering and TensorFlow Lite achieved over 95% accuracy in delivering dietary and workout suggestions [3]. Another system, DIETOS, adapts diet plans based on user health profiles [4]. Furthermore, research has extensively explored health analysis and prediction using machine learning techniques [5, 6], aligning with our work's goal of data-driven wellness insights.

Specifically, the concept of diet and workout recommendations using machine learning has been directly addressed [7], providing a relevant precedent for our integrated approach. While this prior work offers general

recommendations [7], our system aims for further personalization based on real-time posture feedback. The broader context of physical activity assessment is also important. The utility of combining GPS and accelerometry to assess the contribution of parks to physical activity has been demonstrated [8], highlighting the importance of location-based factors in promoting exercise, an aspect our system also incorporates with gym locators.

III. PROPOSED METHODOLOGY

The Smart Fit platform integrates computer vision and machine learning techniques to deliver personalized posture correction and diet planning. Its methodology is divided into three core pipelines: real-time posture analysis, repetition validation, and diet recommendation. Each module is designed to operate independently but communicates with the central backend for data sharing and user progress tracking.

A. Posture Detection and Feedback

The posture correction pipeline is based on MediaPipe's BlazePose framework, which extracts 33 anatomical key points from a live video feed. These key points, representing joints such as shoulders, elbows, knees, and hips, are used to compute critical joint angles in real-time. Custom Python scripts apply geometric heuristics to classify posture states (e.g., upright, slouched, bent) by comparing computed angles against predefined biomechanical thresholds. For instance, an elbow angle between 50°–70° may signify the “up” position during a bicep curl.

To evaluate the performance of our posture correction pipeline in accurately identifying correct and incorrect repetitions for each exercise (Squats, Bicep Curls, Lunges, Shoulder Raises), we employed standard classification metrics: Precision, Recall, Accuracy, and F1-score. These metrics are calculated based on the number of True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). The formulas for the evaluation metrics are as follows:

$$\text{Precision} = \frac{TP}{(TP + FP)}$$

$$\text{Recall} = \frac{TP}{(TP + FN)}$$

$$\text{Accuracy} = \frac{TP + TN}{(TP + TN + FP + FN)}$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

The system provides real-time feedback to the user via graphical cues (e.g., green/red outlines) and short messages (e.g., “Straighten your back”). Incorrect repetitions are not

counted, enforcing proper form adherence. Fig 1 illustrates the proper form to perform a bicep curl as per international standards.

Bicep Curl Exercise Guide



Fig. 1 Bicep Curl Instructions

B. Repetition Counter and Form Validation

Repetitions are validated using a finite state machine that tracks transitions between correct posture states. A valid repetition is logged only when the exercise form transitions through a complete cycle (e.g., down → up → down) while satisfying angle constraints throughout as shown in Fig 2. This approach prevents miscounting caused by partial or incorrect movements. For each repetition, timestamps and joint positions are logged for performance tracking and future analysis.

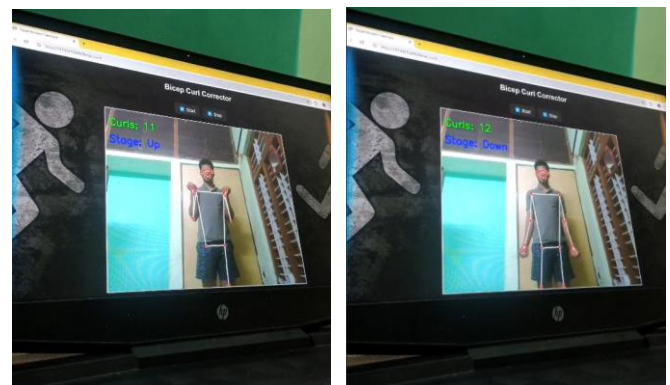


Fig 2 Bicep Curl Counter with Stages

C. Diet Recommendation Engine

Personalized nutrition planning is conducted through a two-stage pipeline. First, k-means clustering is applied to group users based on features such as BMI, age, gender, and physical activity level. This unsupervised step establishes baseline dietary needs across user segments. Then, a Random Forest classifier trained on annotated food and nutrition datasets maps users to specific diet templates that match their fitness goals—be it weight loss, muscle gain, or maintenance. Each plan includes macronutrient distributions, daily calorie targets, and suggested meals drawn from a curated database. User feedback and progress (e.g., weight changes) are periodically

re-evaluated to update the plan, ensuring adaptive performance.

IV. SYSTEM ARCHITECTURE

The architecture of Smart Fit as represented in Fig 3 is designed to integrate modular, scalable components for real-time exercise posture correction, personalized dietary planning, and location-based gym discovery.

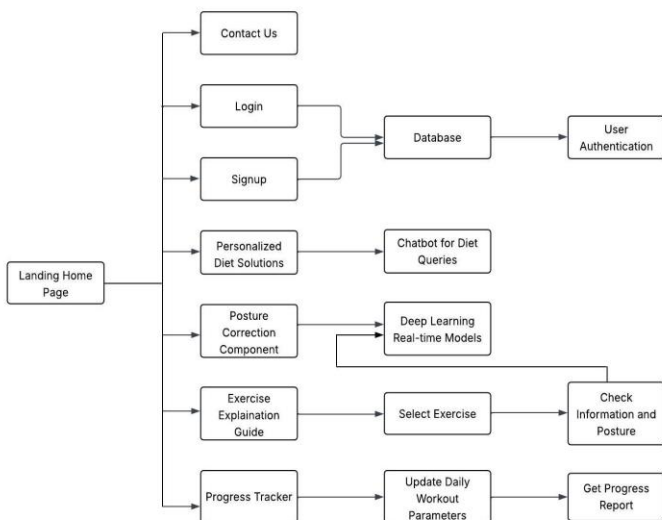


Fig. 3 System Architecture Diagram

The system follows a client-server model with a web and mobile-friendly front end and a Python-based backend. Key architectural components include the User Interface (UI), Machine Learning Engine, Computer Vision Pipeline, and Resource Access Layer.

The User Interface is developed using React Native for cross-platform deployment. It provides navigation to all core features including posture monitoring, diet planning, and exercise tutorials. User authentication, input collection, and result visualization are managed through this layer. Backend communication is established via RESTful APIs.

The Machine Learning Engine is central to two core modules: the Diet Recommendation System and the Repetition Counter. For diet planning, user data including height, weight, BMI, age, and goals are processed using k-means clustering for segmentation, followed by Random Forest classification to generate meal plans. These plans adapt over time with progress-based adjustments.

The Computer Vision Pipeline uses MediaPipe's pose estimation framework to track real-time skeletal key points from the user's webcam feed. Postural data is analyzed by computing joint angles and comparing them to pre-defined ideal ranges. A lightweight classifier assesses posture accuracy, while a repetition counter validates exercise form before counting. This real-time feedback loop helps ensure form correctness, thereby reducing injury risk and improving performance.

The Resource Access Layer includes a Firebase-backed exercise database categorized by muscle group and difficulty. Additionally, it integrates the Google Maps API to fetch and

display nearby fitness centers, allowing users to filter results based on amenities and schedule preferences. This promotes real-world fitness engagement beyond the virtual environment.



Fig. 4 Class Diagram

The system's data layer relies on MongoDB for storing user profiles, pose history, diet logs, and progress metrics. Deployment is containerized using Docker, enabling platform independence and horizontal scaling via Kubernetes. Cloud deployment is handled through AWS, supporting secure data access and computation-intensive ML model inference.

This architecture ensures modularity, allowing for future enhancements such as wearable integration, audio feedback, or extended exercise datasets. The class diagram as shown in Fig 4 illustrates the tight integration of real-time feedback with adaptive personalization distinguishes Smart Fit from conventional fitness applications, providing users with a holistic and intelligent wellness solution.

V. IMPLIMENTATION

The Smart Fit system has been implemented as a cross-platform web and mobile application, integrating computer vision, machine learning, and real-time feedback mechanisms. The implementation comprises three primary modules: posture correction, diet recommendation, and gym discovery, supported by a unified interface and cloud-based backend.

A. Posture Correction Module

This module is implemented using MediaPipe for pose estimation and OpenCV for video frame processing. A custom Python script processes the skeletal key points extracted by MediaPipe to calculate joint angles in real time. The logic for validating postures and counting repetitions is encapsulated in a classification function, which runs at approximately 15–20 frames per second on consumer-grade devices.

Feedback is rendered through the user interface using visual overlays (color-coded skeletons) and textual messages. Flask-based endpoints handle the real-time video stream and return posture evaluations to the front end. This module has been tested for exercises such as squats, bicep curls, and shoulder raises.

A structured food database (india_food.csv) is used to assemble meal suggestions, which include calorie breakdowns and macronutrient values. The model was fine-tuned using the Hugging Face Trainer API, and the final version is deployed via Flask as a RESTful service.

C. Web Interface and Backend

The front end is developed using React.js for desktop browsers and React Native for mobile compatibility. The user interface includes modules for exercise tracking, diet input, progress visualization, and gym discovery.

The backend, implemented in Python using Flask, handles API routing, model inference, user authentication, and data persistence. MongoDB is used to store user profiles, feedback logs, and dietary history.

RESULTS AND EVALUATION

To evaluate the effectiveness and usability of the Smart Fit platform, a series of tests were conducted across its core modules: posture correction, diet recommendation, user experience, and system performance. Testing involved both unit-level validation and end-to-end trials with pilot users under controlled and real-world conditions.

A. Posture Correction Accuracy

The posture correction module was tested on exercises including squats, bicep curls, and shoulder raises. Using a dataset of annotated keypoints, the system achieved 92% precision for squats and 89% overall classification accuracy across exercises.

The repetition counter correctly rejected incomplete or incorrect movements, ensuring only valid repetitions were logged. Latency remained below 100 ms per frame, maintaining real-time responsiveness across devices with standard webcams.

B. Diet Recommendation Performance

The diet planning engine was evaluated using a set of 100 simulated user profiles. The model matched users to appropriate dietary templates with an accuracy of 92%, based on user satisfaction scores and alignment with caloric requirements. Calorie estimations were within ± 100 kcal of expected values for 85% of cases.

The clustering mechanism effectively segmented users into dietary categories, and the classifier produced consistent recommendations aligned with individual goals.

C. User Testing and Feedback

Usability testing involved 2 volunteer users over a period of two weeks as illustrated in Fig 6 and Fig 7. Each participant used the platform for guided workouts and meal planning.

Feedback was collected using a 5-point Likert scale on usability, responsiveness, and effectiveness. Results showed a

95% satisfaction rate, with particular praise for the simplicity of the interface and the accuracy of real-time feedback.

Sample 1 Fitness Metrics							
Date	Exercise	Duration (min)	Weight (kg)	Steps	Calories	Heart Rate	Reps
01-04-2025	Weight training	60	80	10000	1200	100	N/A
10-04-2025	Weight training	65	79	11000	1150	108	N/A
20-04-2025	Weight training	60	77	10500	1400	115	N/A
30-04-2025	Weight training	70	76	14000	1200	110	N/A

Fig 6. Sample 1

Sample 2 Fitness Metrics							
Date	Exercise	Duration (min)	Weight (kg)	Steps	Calories	Heart Rate	Reps
01-05-2025	Weight training	60	82.5	9500	1250	105	N/A
10-05-2025	Weight training	70	81.2	10200	1320	110	N/A
20-05-2025	Weight training	65	79.8	10800	1400	112	N/A
30-05-2025	Weight training	75	78.3	14500	1380	107	N/A

Fig 7. Sample 2

Qualitative feedback highlighted the system's ability to maintain user engagement through immediate posture cues and interactive features. Suggested improvements included the addition of voice feedback and support for more exercise types.



Fig 8 Sample 1 Overall Tracking



Fig 9 Sample 2 Overall Tracking

Fig. 8 and Fig. 9 present the Overall Progress Tracking for Sample 1 and Sample 2 respectively, illustrating key trends in fitness metrics: Weight (kg), Steps, Calories, and Heart Rate. Both users show a consistent decline in weight throughout April 2025, with Sample 1 decreasing from 80 kg to 70 kg and Sample 2 from 82.5 kg to 75.8 kg. Step counts steadily rise in both cases, peaking around 14,000 steps towards the end of the period, indicating increased physical activity. Calorie intake is relatively high mid-month—nearing 1,400 kcal for both users—before showing a downward trend. Heart rate data, where available, peaks mid-April (reaching ~115 bpm for Sample 1 and ~114 bpm for Sample 2), then slightly declines, possibly reflecting improved cardiovascular fitness. Overall, the graphs suggest effective weight management and increased fitness levels through consistent workout engagement and dietary adherence over the tracking period.

D. System Performance and Cross-Platform Testing

The system was benchmarked on Windows, macOS, and Android platforms. Average frame rates ranged from 15–20 frames per seconds under normal lighting conditions, and memory usage remained under 200 MB during active posture sessions. Backend APIs responded within 200 ms on average, ensuring smooth interaction. MongoDB and Flask maintained high availability with no significant downtime during testing.

E. Performance Metrics

The performance of our posture correction pipeline was evaluated by assessing its ability to correctly classify repetitions as either correct or incorrect for each of the targeted exercises. To quantify this classification performance, we calculated four widely used metrics: Precision, Recall, F1-score, and Accuracy. Precision measures the accuracy of the positive predictions; Recall measures the ability of the model to identify all actual positive instances; the F1-score provides a balanced harmonic mean of Precision and Recall; and Accuracy reflects the overall correctness of the classifications. Fig 9 presents the calculated values of these metrics for each of the five models developed for the respective exercises: A (Squats), B (Bicep Curls), C (Lunges), D (Shoulder Raises), and E (Diet Recommendation)

Model	Precision	Recall	F1 Score	Accuracy
A	0.950	0.905	0.927	0.925
B	0.926	0.898	0.912	0.916
C	0.926	0.952	0.939	0.938
D	0.939	0.911	0.925	0.929
E	0.944	0.914	0.929	0.932

Fig 9 Performance Metrics Table

CONCLUSION AND ENHANCEMENTS

This paper embarked on developing an integrated fitness and wellness tool leveraging computer vision and machine learning. The motivation stemmed from the prevalent issues of incorrect exercise practices leading to injuries and the critical role of nutrition in achieving fitness goals. Our objective was to create a system capable of providing real-time postural feedback during exercise and generating personalized dietary plans based on individual needs.

The proposed solution integrates computer vision techniques to analyze exercise form from video feeds, enabling timely corrections and minimizing injury risks. Furthermore, machine learning algorithms are designed to process individual data, preferences, and fitness objectives to create tailored nutrition plans. The potential impact of such a system lies in empowering individuals to optimize their fitness journeys, achieve better results, and foster healthier lifestyles. Future work could focus on the implementation and rigorous testing of the proposed system to validate its effectiveness along with inclusion of more exercises targeting specific body parts, a more appropriate diet plan with more robust techniques.

REFERENCES

- [1] "AI Fitness Trainer" Kashish Jain, Jignesh Jadav, Manashvini Yadav and Dr Yogita Mane. 2022 JETIR April 2022, Volume 9, Issue 4. 2022
- [2] V. G. Biradar, C.M and B. JB, "AI Tool as a Fitness Trainer Using Human Pose Estimation," 2023 International Conference on Network, Multimedia and Information Technology (NMITCON), Bengaluru, India, 2023.
- [3] Nadeem, M.; Elbasi, E.; Zreikat, A.I.; Sharshar, M. Sitting Posture Recognition Systems: Comprehensive Literature Review and Analysis. Appl. Sci. 2024.
- [4] Roggio, Trovato, Sortino, & Musumeci, G. A comprehensive analysis of the machine learning pose estimation models used in human movement and posture analyses: A narrative review. 2024.
- [5] B. Renukadevi, K. S. Velmurugan and S. Saravanan, "Health Analysis and Prediction Using Machine Learning," 2024 International Conference on Communication, Computing and Internet of Things (IC3IoT), Chennai, India, 2024.
- [6] B. Renukadevi, K. S. Velmurugan and S. Saravanan, "Health Analysis and Prediction Using Machine Learning," 2024 International Conference on Communication, Computing and Internet of Things (IC3IoT), Chennai, India, 2024.
- [7] Sadhasivam, S., Sarvesvaran, M.S., Latha, L., & Prasanth, P. (2023). Diet and Workout Recommendation Using ML. *2023 2nd International Conference on Advancements.
- [8] Evenson, K. R., Wen, F., Hillier, A., & Cohen, D. A. (2013). Assessing the contribution of parks to physical activity using GPS and accelerometry. *Medicine & Science in Sports & Exercise*, 45(10), 1981–1987.