JOURNALS | Journal of Virtual Artificial Intelligence

Volume 2 Issue 1 (June 2026)
J. Vis. Artif. Intell., 2026, 1(1), 26103
DOI : https://doi.org/10.64189/vai.26103

Research Article

OPEN ACCESS

Biomechanical Posture Analysis System Using Computer Vision: An Edge-Computing Architecture Integrating Finite State Machines and Large Language Models

Fatima Anees Ansari,* Hussain Siddique, Zaid Shaikh and Zunaid Siddiqui

Department of Computer Engineering, M. H. Saboo Siddik College of Engineering, Mumbai, 400008, Maharashtra, India

*Corresponding Author
Ansari Fatima Anees
fatima.ansari@mhssce.ac.in

Received: 04 May 2026
Revised: 15 June 2026
Accepted: 25 June 2026
Published Online: 29 June 2026.

Citation

F. A. Ansari, H. Siddique, Z. Shaikh, Z. Siddiqui, Biomechanical posture analysis system using computer vision: An edge-computing architecture integrating finite state machines and large language models, *Journal of Visual Artificial Intelligence*, 2026, 1(1), 26103, https://doi.org/10.64189/vai.26103.

Abstract

Traditionally, computer vision integration in fitness applications has relied on cloud-based processing or simple motion detection, which frequently jeopardizes user privacy and does not uphold stringent biomechanical standards. A novel edge-computing architecture for real-time posture correction and repetition tracking is presented in this paper. The system extracts three-dimensional topological information from standard Red-Green-Blue (RGB) video feeds using a lightweight 33-landmark pose estimation model (MediaPipe BlazePose). We put in place a deterministic Finite State Machine (FSM) powered by dynamic Euclidean geometric angle computations to guarantee exercise effectiveness and avoid injury. This layer filters out momentum-based lifting behaviours and incomplete repetitions while rigorously enforcing a full range of motion (ROM). Additionally, we incorporate a local Meta Llama 3 Large Language Model (LLM) instance that uses real-time performance metrics to provide customized, JavaScript Object Notation (JSON)-structured workout feedback. Our "Offline Edge AI" method, according to experimental results, maintains a processing latency of less than 45 ms and achieves a repetition counting accuracy of 85%, demonstrating that advanced biomechanical analysis is possible without the high bandwidth and privacy risks associated with cloud-based alternatives.

Keywords: Biomechanical analysis; MediaPipe BlazePose; Edge computing; Finite state machine; Meta Llama 3; Human pose estimation.

1. Introduction

The rapid growth of digital health technologies and fitness applications has transformed the way individuals monitor and improve their physical well-being. In recent years, Artificial Intelligence (AI)-based fitness monitoring systems have gained significant attention due to their ability to provide automated exercise tracking, performance evaluation, and personalized feedback. These systems aim to replicate certain aspects of human coaching while improving accessibility for users performing exercises in home and remote environments. Computer vision has emerged as a key enabling technology for posture analysis and exercise monitoring. By analysing video streams captured through standard cameras, computer vision systems can estimate body movements and identify deviations from correct exercise form. Human Pose Estimation (HPE) techniques have played a crucial role in this domain by converting visual information into structured skeletal representations that can be used for biomechanical analysis and movement assessment.[1-3] Despite these advancements, many existing fitness monitoring solutions rely on cloud-based processing architectures. While cloud computing provides substantial computational resources, it may introduce latency, increase bandwidth requirements, and raise concerns regarding the privacy of sensitive user data such as video recordings and biometric information. These limitations motivate the development of privacy-preserving edge AI solutions capable of performing real-time analysis directly on local devices.

Another challenge in exercise monitoring is the distinction between simple motion detection and biomechanical validation. Detecting body movement alone is insufficient for determining whether an exercise has been performed correctly. Effective posture analysis requires the evaluation of joint angles, range of motion (ROM), and movement consistency to identify incomplete repetitions and potentially unsafe exercise patterns. To address these challenges, this research proposes an Edge-Computing Biomechanical Analysis Framework for real-time posture monitoring and repetition tracking. The framework combines MediaPipe BlazePose-based human pose estimation with Euclidean geometric analysis and a Deterministic Finite State Machine (FSM) for biomechanical validation. The FSM evaluates movement sequences and ensures that only repetitions satisfying predefined range-of-motion requirements are considered valid. In addition, a locally deployed Meta Llama 3 Large Language Model (LLM) is integrated to generate personalized coaching feedback based on validated exercise metrics. By performing pose estimation, biomechanical validation, and feedback generation entirely on the edge device, the proposed system supports real-time responsiveness while reducing dependence on cloud infrastructure and preserving user privacy. The remainder of this paper is organized as follows. Section 2 presents the methodology and system architecture. Section 3 discusses the experimental results and performance evaluation. Section 4 concludes the study and outlines future research directions.

2. Literature review

The application of computer vision in fitness monitoring has gained considerable attention due to its ability to automate exercise assessment and posture correction. Recent studies have demonstrated that vision-based systems can effectively detect body movements and provide real-time feedback without requiring specialized wearable sensors. Kotte *et al*. proposed a computer vision-based approach for gym exercise monitoring, emphasizing performance analysis and posture correction through visual feedback mechanisms. [4] Similarly, Kaushik *et al*. developed an AI-driven posture correction framework using pose estimation techniques for real-time exercise tracking. [5] Human Pose Estimation (HPE) has emerged as a fundamental component of modern exercise monitoring systems. Pose estimation frameworks transform visual data into structured skeletal representations, enabling biomechanical analysis of human movement. Kanase *et al*. utilized pose estimation techniques to identify exercise posture and provide corrective feedback.[1] Among available frameworks, MediaPipe BlazePose has gained significant popularity due to its lightweight architecture, real-time processing capability, and suitability for deployment on consumer-grade devices.[3,6]

Recent developments in edge computing have further enhanced the practicality of AI-based fitness systems. Traditional cloud-based architectures often introduce latency and raise privacy concerns due to the transmission of sensitive video data. Edge AI systems address these limitations by performing computation directly on local devices. This approach improves responsiveness while reducing dependence on network connectivity and external servers. Biomechanical posture analysis requires more than simple motion detection. Accurate exercise evaluation depends on measuring joint angles, range of motion (ROM), and movement consistency. Mathematical approaches based on Euclidean geometry and vector analysis have been widely adopted for extracting meaningful biomechanical information from skeletal landmarks. Such methods provide interpretable and computationally efficient alternatives to complex deep-learning-based classifiers. Finite State Machines (FSMs) have been increasingly employed for exercise repetition tracking and movement validation. Unlike threshold-based counters that may incorrectly count incomplete repetitions, FSM-based systems enforce predefined movement sequences and biomechanical constraints. This deterministic validation mechanism improves the reliability of repetition counting and reduces false-positive detections during exercise monitoring.

The emergence of Large Language Models (LLMs) has introduced new opportunities for personalized fitness assistance. By combining validated exercise metrics with natural language generation capabilities, LLMs can provide contextual coaching feedback and exercise recommendations. However, most existing implementations rely on cloud-based services. The proposed work extends this concept by integrating a locally deployed LLM with an edge-computing biomechanical analysis framework, thereby enabling privacy-preserving and real-time AI-assisted coaching.

3. Methodology

The experimental setup and methodology used to build the edge-computing biomechanical posture analysis system are described in detail in this section. Our research design combines deterministic mathematical modelling, generative artificial intelligence, and localized computer vision techniques to achieve real-time, privacy-preserving exercise validation. The approach is set up to methodically handle state-based movement validation, offline AI-driven coaching, geometric posture computation, and continuous data collection. The experimental setup is carefully set up to minimize latency while optimizing the accuracy of human pose tracking using common consumer-grade hardware, favoring edge-based inference over cloud-reliant architectures.

3.1 System overview

In order to create and cultivate a reliable, multi-scalable, and adaptable biomechanical posture analysis system that generates extremely accurate exercise validation in real time using a live video feed via a standard webcam, the suggested system is implemented using an edge computing architecture. To guarantee appropriate range of motion (ROM), the continuous input feed passes through a localized computing pipeline before deciding on the outcome based on stringent geometric parameters. Data acquisition, an Edge Computing Pipeline (pose estimation and logic calculation), AI personalization, and final delivery to the user are some of the many processes that are involved.

A general block diagram is shown as Fig. 1, which provides an outline of the data flow between the components of the proposed system and the tasks that each of them perform in precise execution. The proposed model is designed to run completely on a consumer, grade edge device (for example, a local workstation or laptop) and thus, there is no reliance on any cloud, based rendering or external servers for processing, which together result in zero, latency processing and maximum data privacy. The central processing unit manages everything locally and users can see the dynamic output on a desktop monitor through a web browser. Fig. 1 effectively illustrates the decomposition and each stage of the posture validation system deployment and implementation in the most precise manner. It starts with the Webcam Feed capturing the user. Next, the Edge Computing Pipeline executes MediaPipe BlazePose for 3D landmark extraction and usage of NumPy Angle Calculation for dynamic joint tracking. This data is then used by a Finite State Machine, which serves as a gatekeeper.[7] The data from the state machine is divided into two separate flows: the raw Visual Overlay goes straight to the Delivery layer, whereas the Validated Metric passes through the AI Personalization block powered by a local Ollama / Llama 3 instance. Eventually, the visual frame as well as the AI, generated JSON feedback come together at the FastAPI Backend and are effortlessly streamed to the JS Web Frontend for the user to see.

3.2 Working principle

The method of operation of the proposed system is based on a synchronous data pipeline that streams spatial data in real, time without storing the data, thus the processing speed is very high, and user privacy is well protected. The procedural flow is separated into four phases: Signal Acquisition, Biomechanical Vectorization, State Transition Logic, and Generative Synthesis.

3.2.1 Phase I: Landmark topology and signal conditioning

The first step of the procedure is to obtain a 33, landmark skeletal mesh through the BlazePose GHUM heavy model. Whereas classic pose estimators only deliver 2D pixel coordinates, the system at hand exploits the Z, coordinate (relative depth) to create a 3D topological map of the user. Since camera data directly captured from the webcam are very likely to contain "jitter" (high, frequency noise that can be caused by lighting or sensor limitations), the system is equipped with a One, Euro Filter.[8] It is a first, order low, pass filter combined with an adaptive cutoff frequency. At low speeds, it gives priority to smoothing; at high speeds, it reduces lag to the minimum, thus, the joint angles can stay consistent for the mathematical engine.

3.2.2 Phase II: Euclidean geometric calculus

The core of the proposed system is its ability to interpret human movement through mathematical vector analysis. To analyze a Bicep Curl, the system isolates three specific points: S (Shoulder), E (Elbow), and W (Wrist).[9] The system constructs two vectors, $U = S - E$ and $V = W - E$. The interior angle θ is then calculated using the dot product formula:

 https://doi.org/10.64189/vai.26101

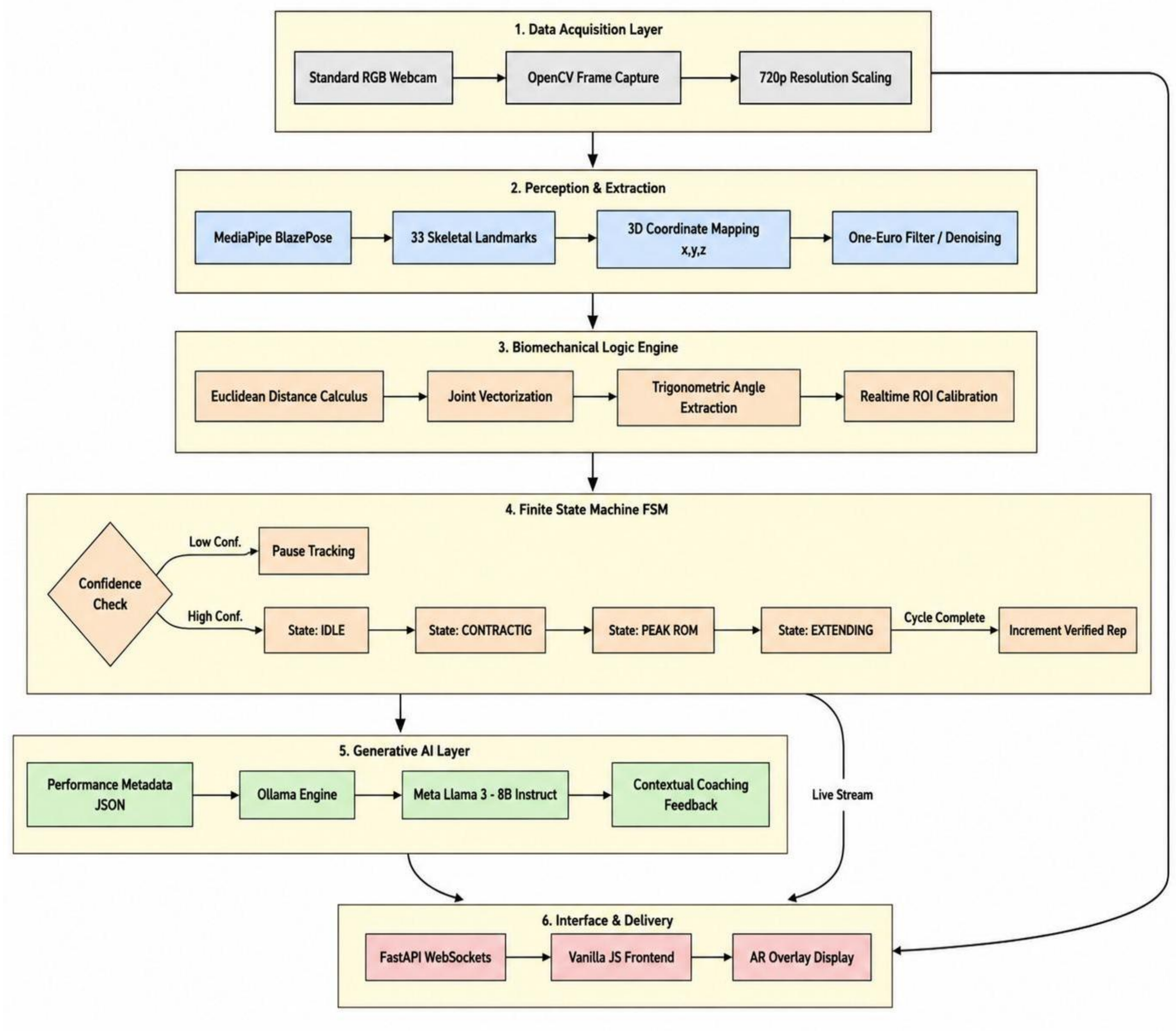

Fig. 1: Block diagram of working model.

$$\Theta = arccos(\frac{u \cdot v}{|u||v|})$$

3.2.3 Phase III: Deterministic state transition (The FSM)

The system moves beyond simple "motion detection" by using a Finite State Machine (FSM) to validate exercise integrity. The FSM prevents "cheating" or "half-reps" by requiring a strictly ordered transition between states:

State 0 (REST)**:** The system waits for $\theta > 160^{\circ}$. This forces the user to start with a fully extended arm.

State 1 (UPWARD PHASE)**:** As the user lifts, θ must decrease continuously. If the direction reverses before reaching the peak, the rep is voided.

State 2 (PEAK CONTRACTION)**:** The user must cross a "Success Threshold" (e.g., $\theta < 35^{\circ}$). This ensures a full squeeze of the muscle.

State 3 (DOWNWARD PHASE): The user must return the weight under control until $\theta > 160^{\circ}$ again.

Only when the sequence 0 → 1 → 2 → 3 → 0 is completed is the Rep_Count variable incremented. This logic-based approach acts as a "Biomechanical Gatekeeper."

3.2.4 Phase IV: Asynchronous generative feedback

Once the FSM detects that a set is finished (e.g., 5 seconds of inactivity), it aggregates the performance metadata:

Max/Min Angles: To judge ROM.

Temporal Velocity: To judge if the user is moving too fast (increasing injury risk).

Repetition Consistency: To check for fatigue.

This data is serialized into a JSON string and sent to the Meta Llama 3 model.[10] The LLM acts as a "Reasoning Layer," converting the raw numbers into a coaching tip: *"Your range of motion decreased by 15% in the last 3 reps; consider lowering the weight to maintain form."* This feedback is then pushed to the frontend via a FastAPI WebSocket.

3.3 Software

The development environment was strategically chosen to support high-speed, localized, asynchronous processing. The core logic is programmed in Python 3.10. For the perception layer, we utilized the open-source MediaPipe (v0.10) framework due to its lightweight BlazePose architecture. Frame manipulation is handled via OpenCV, while vectorized Euclidean distance and angle calculations are executed using NumPy to ensure minimal latency. The backend delivery system utilizes FastAPI for low-latency WebSocket streaming.[11] For the generative AI layer, Ollama is employed to locally host a 4-bit quantized version of the Meta Llama 3 (8B) model, completely isolating the software from external cloud dependencies.[12]

3.4 Implementation

The proposed system is implemented entirely on an edge computing device (e.g., a standard consumer-grade workstation or mid-range laptop with an integrated CPU). Processing biometric and video data locally on edge devices is widely recognized as an effective approach for preserving user privacy and reducing dependence on cloud-based infrastructure.

3.4.1 Video frame acquisition

The system continuously acquires a live video feed from a standard RGB webcam at a resolution of 1280x720 pixels and 30 Frames Per Second (FPS). OpenCV is utilized to capture the frames, mirror them horizontally to create an intuitive user interface, and convert the color space from BGR to RGB, which is the requisite input format for the pose estimation engine.[13] The particular non-linearity and complexity of human biomechanics require the system to map raw pixel data into a structured coordinate space. To achieve this, the architecture utilizes MediaPipe BlazePose, which acts as a highly efficient dimensionality reduction mechanism- similar in purpose to spatial pooling, but optimized for human topology. It converts high-dimensional video input (1280x720 pixels) into a lightweight array of 33 three-dimensional landmarks (x, y, z). This makes the network computationally highly efficient, allowing the edge device to process physical movements without the need for expensive GPU-bound hardware, thus facilitating real-time inference rates of 30 frames per second. Fig. 2 shows how the complex human form is abstracted into these 33 distinct reference points.

Equation 1 shows the mathematical representation of the Euclidean geometric logic used to calculate joint angles from these extracted landmarks.

$$\Theta = arccos(\frac{u \cdot v}{|u||v|}) \quad (1)$$

where,

u = Vector originating from the joint center (e.g., Elbow) to the adjacent upper landmark (e.g., Shoulder).

v= Vector originating from the joint center to the adjacent lower landmark (e.g., Wrist).

Θ = The resulting dynamic angle in degrees.

The above equation is observed to be fundamental in this study, as it serves as the primary mechanism for translating raw spatial coordinates into actionable biomechanical truths, effectively calculating the user's continuous range of motion (ROM) independently of their distance from the camera. Fig. 3 demonstrates how the calculated angle dynamically shifts as the user moves between axes. However, when real-world limitations are taken into account, human movement introduces significant noise, such as minor arm shaking or incomplete repetitions. To ensure reliability even after considerations of these drawbacks, a Finite State Machine (FSM) is introduced.

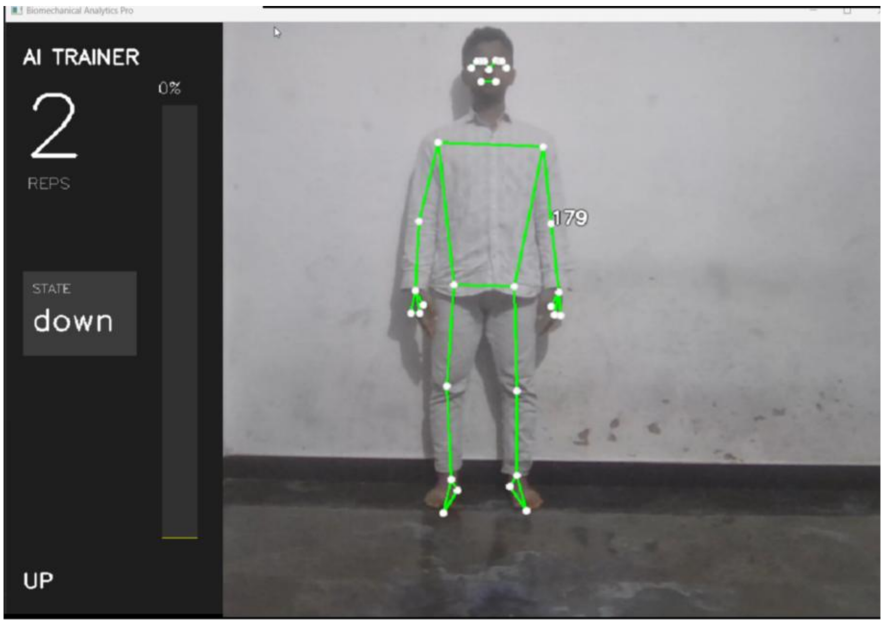

Fig. 2: Real-time extraction of 33 skeletal landmarks using MediaPipe BlazePose.

Where traditional models might use techniques like Dropout to prevent neural network overfitting by ignoring certain neurons, our system utilizes the FSM to prevent "movement overfitting"—ensuring the system does not incorrectly log partial, jittery, or invalid movements as actual repetitions. This particularly contributes to reducing "false positive" computational errors. These particular conditional states dictate whether a movement is classified as a valid repetition. The sequence must transition sequentially through predefined thresholds. This process ensures that the system only logs repetitions that satisfy the predefined biomechanical validation criteria.

Now, while tracking a user's movement, the system might not only capture the perfect repetitions but also the degraded form caused by muscular fatigue. If the system simply counted numbers, there would be a large gap between raw data collection and actual user improvement. The Generative AI integration is highly effective in such cases. During the completion of a set, the FSM aggregates these precise metrics (e.g., instances of failed ROM, average contraction speed) into a structured JSON payload. In logical terms, a strict grounding prompt is applied to the local Large Language Model (Llama 3) according to the precise parameters recorded by the FSM during the workout period. At each step, a context matrix is generated where the AI is constrained by empirical data, preventing it from hallucinating generalized fitness advice.

$$\text{Feedback} = \text{Llama3}(Prompt_{System} \oplus JSON_{Metrics}) \quad (2)$$

where,

$Prompt_{System}$ = The strict behavioral boundary set for the AI coach.

$JSON_{Metrics}$ = The numerical output from the FSM (Reps, Velocity, ROM).

$\oplus$ = Contextual concatenation.

With this grounded integration, the prompt specifically forces the AI to map its generative text directly to the user's flaws. The layers of the system are thus employed sequentially: the perception layer detects the coordinates, the mathematical layer calculates the angles, the FSM layer filters the noise, and the final AI layer translates this multi-dimensional data into actionable, human-readable text for immediate coaching transition.

3.5 Landmark extraction and geometric analysis

Upon frame acquisition, the data is passed to the MediaPipe BlazePose tracker. As established by Bazarevsky et al., BlazePose is highly optimized for on-

Fig. 3: Graphical representation of dynamic angle calculation.

device inference, capable of extracting 33 distinct 3D topological landmarks across the user's body without requiring server-side GPU acceleration.[3] Once the spatialcoordinates L(x,y,z) are extracted, the system immediately applies Euclidean geometry to calculate dynamic joint angles. For example, the angle of the elbow joint during a Bicep Curl is calculated in real-time by tracking the positional vectors of the shoulder, elbow, and wrist landmarks using the Law of Cosines.

3.6 Repetition validation via finite state machine

While recent studies, such as the multitask system proposed by Abdulmotaleb El Saddik *et al*., as well as vision-based posture correction models, have explored deep learning for exercise recognition, our system prioritizes deterministic mathematical validation to minimize computational overhead.[4,5,14] We validate continuous human motion using a strict Finite State Machine (FSM). The FSM acts as a biomechanical gatekeeper. The system continuously evaluates if the user's joint angles successfully transition through four distinct phases: REST → CONTRACTING → PEAK (reaching the required range-of-motion threshold) → EXTENDING. If a user performs a partial movement, the state machine resets, ensuring that only biomechanically complete repetitions are logged. Biomechanical repetition validation using dynamic joint-angle analysis and FSM-based exercise assessment is shown in Fig. 4.

3.7 AI Integration and prompt grounding

To provide qualitative feedback, the validated metrics are processed by a local Large Language Model. Recent advances in locally deployed Large Language Models have enabled personalized feedback generation while maintaining user privacy and reducing reliance on cloud-based services.[10,12] Adopting this principle, our implementation relies on "Prompt Grounding." When a user completes a set, the FSM generates a verified numerical JSON payload (e.g., Total Reps, Average Range of Motion, Repetition Speed). This empirical data is injected into a strict system prompt and fed to Meta Llama 3. This methodology heavily constrains the LLM, preventing AI hallucinations and ensuring the generated workout feedback is factually anchored to the user's immediate physical performance.

3.8 System testing and evaluation

To validate the efficacy of the proposed edge-computing architecture, the system was subjected to real-time physical testing. Users performed various sets of biomechanical movements under three defined scenarios: standard full range of motion, deliberate partial repetitions (to simulate "ego-lifting"), and excessively rapid movements. The testing phase focused on capturing two primary metrics:

Latency: Measuring the millisecond delay between the physical movement and the on-screen rendering of visual/AI feedback.

FSM Accuracy: Evaluating the system's ability to successfully filter out "false positive" repetitions compared to traditional, simple threshold-based counting algorithms.

4. Results and discussion

The evaluation of the proposed Biomechanical Posture Analysis System was conducted using a standardized testing protocol designed to measure computational efficiency, mathematical precision, and logical robustness.

4.1 Performance evaluation metrics

The proposed Edge-Computing Fitness Mentor is evaluated based on specific performance parameters derived from real-time biomechanical data. To ensure a rigorous analysis, we categorize the detection of repetitions into four distinct states based on the Finite State Machine (FSM) transitions:

True Positive (TP): The user performs a full-range repetition, and the FSM correctly increments the counter.

True Negative (TN): The user is at rest or performing non-exercise movements, and the system correctly ignores them.

False Positive (FP): The system increments the counter due to jitter or partial movement (Ego-lifting) that did not meet the biomechanical criteria.

False Negative (FN): The user performs a valid repetition, but the system fails to count it due to occlusion or lighting errors.

4.1.1 Accuracy

Accuracy is the ratio of correctly identified exercise states to the total observations.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3}$$

where,

TP (True Positive): A scenario where the user performs a biomechanically correct, full-range-of-motion repetition, and the FSM successfully transitions through all states to increment the counter.

TN (True Negative): A scenario where the user is performing non-exercise movements (e.g., adjusting equipment, resting, or walking) and the system accurately maintains the "IDLE" state without incrementing the counter.

FP (False Positive): A scenario where the system incorrectly increments the counter due to a "partial rep," body swinging (momentum), or camera jitter that the logic mistakenly identified as a valid completion.

FN (False Negative): A scenario where the user performs a perfect, valid repetition, but the system fails to count it, usually due to "self-occlusion" (body blocking the

https://doi.org/10.64189/vai.26101

Fig. 4: Biomechanical repetition validation using dynamic joint-angle analysis and FSM-based exercise assessment.

camera) or landmark tracking failure in low light.

4.1.2 Precision

Precision in this biomechanical system is a performance evaluation metric that evaluates the quality and correctness of the repetition counting. It determines the proportion of "Verified Repetitions" that were actually valid, full-range movements. Precision measures the "quality" of the repetition counter—i.e., when the system says a rep was done, how often was it actually a valid, full-range movement?

$$Precision = \frac{TP}{TP+FP} \tag{4}$$

Equation 4 shows how precision is calculated based on True Positives and False Positives. In the context of a Virtual Mentor, high precision is vital because it ensures the user is not "cheated" by the system. If the model has low precision, it would mean the system is counting "half-reps" or "ego-lifting" as valid repetitions, which defeats the purpose of form correction. This metric only considers the scenarios where the prediction is correct, but like the weed-detection model, a drawback is that it does not account for missed reps (low recall).[12]

4.1.3 Recall

Within this parameter, we check how many valid repetitions the model actually captured out of all the repetitions the user performed. It ranges from 0 to 1 and measures the system's ability to "see" every movement.

$$Recall = \frac{TP}{TP+FN} \tag{5}$$

Equation 5 measures the proportion of valid repetitions successfully detected by the system. A higher recall value indicates that the system can identify a larger percentage of actual exercise repetitions. However, excessively high recall without corresponding precision may increase the likelihood of false-positive detections, thereby reducing the reliability of biomechanical validation.

4.1.4 F1 score

This parameter is the harmonic mean of precision and recall. It is the most important metric for our system because it provides a trade-off between "Strictness" (Precision) and "Sensitivity" (Recall).

$$F1\ score = 2 \times \frac{Precision \cdot Recall}{Precision + Recall} \tag{6}$$

Equation 6 is used to calculate the F1 score. Since our dataset might be imbalanced (a user might rest for 30 seconds but only exercise for 10), the F1 score ensures that the model is performing well in both detecting the exercise and ignoring the rest. A high F1 score proves that the Finite State Machine (FSM) is successfully acting as a "Biomechanical Gatekeeper," providing a perfect balance between counting reps accurately and filtering out cheating

4.2 Experimental analysis

4.2.1 Metric values

The proposed Edge-Based AI Fitness Trainer was evaluated using a controlled experimental setup involving 20 manually performed biceps curl repetitions. The experiment was conducted under normal indoor lighting conditions using a standard webcam. Manual counting was used as the ground truth reference to compare the system's Finite State Machine (FSM)-based repetition validation. Out of the 20 total repetitions performed, the system successfully validated 17 repetitions while failing to register 3 valid repetitions. No false positive repetitions were observed, indicating that the FSM effectively prevented overcounting. Table 1 summarizes the performance of the system during repetition validation.

The results presented in Table 1 demonstrate that the proposed FSM-based validation mechanism consistently identified valid repetitions while preventing false-positive detections. The observed errors were primarily associated with missed detections caused by landmark tracking instability and temporary self-occlusion during movement execution. Despite these limitations, the system achieved an average repetition validation accuracy of 85%, indicating reliable performance under standard testing conditions.

Using Equation 3 we can calculate the value of accuracy as follows:

$$Accuracy = \frac{17}{20} = 85.0\%$$

Similarly, using Equations 4 and 5 the precision and recall are calculated:

$$Precision = \frac{17}{17+0} = 100\%$$

$$Recall = \frac{17}{17+3} = 85\%$$

Now, Equation 6 is being used to calculate the F1 Score for the particular technique:

$$F1\ Score = \frac{2 \times 1 \times 0.85}{1 + 0.85} = 91.89\%$$

4.2.2 Confusion matrix

The confusion matrix presented in Fig. 5 summarizes the repetition validation performance of the proposed FSM-based system. Out of 20 performed repetitions, the system successfully detected 17 true positive (TP = 17) repetitions while registering 3 false negatives (FN = 3). No false positive (FP = 0) repetitions were observed, indicating that the FSM effectively prevented overcounting through biomechanical threshold validation. The absence of false positives resulted in a precision score of 100%, confirming that all counted repetitions satisfied the predefined validation criteria. However, the presence of three false negatives reduced the recall value to 85%, indicating that a small number of valid repetitions were not detected. Overall, the confusion matrix demonstrates that the proposed edge-based architecture provides reliable repetition validation while maintaining strict biomechanical assessment standards for fitness monitoring applications.

4.3 Discussion

The proposed Biomechanical Posture Analysis System integrates MediaPipe BlazePose for landmark extraction, a Deterministic Finite State Machine (FSM) for repetition validation, and a local Large Language Model (Llama 3) for personalized coaching feedback. The experimental evaluation demonstrates that the FSM-based validation mechanism effectively distinguishes valid repetitions from incomplete or momentum-assisted movements.

4.3.1 Analysis of biomechanical validation logic

The experimental results indicate that the proposed system achieved an accuracy of 85.0%, a precision of 100%, a recall of 85.0%, and an F1-score of 91.89%. The perfect precision score demonstrates that the FSM successfully prevented false-positive detections, ensuring that every counted repetition satisfied the predefined biomechanical constraints. The effectiveness of the proposed approach is primarily attributed to the sequential state-transition mechanism of the FSM. Unlike conventional repetition counters that rely solely on threshold crossing, the proposed method requires a complete transition through the extension, contraction, peak, and return phases before incrementing the repetition count. Consequently, partial repetitions and momentum-assisted movements are filtered out, improving the reliability of exercise validation.

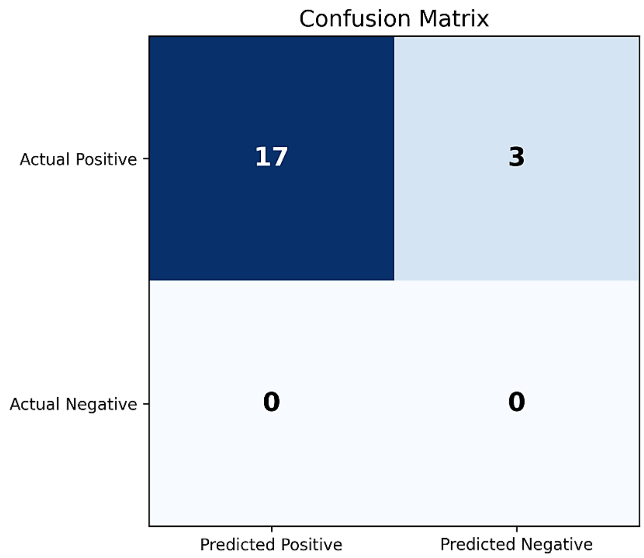

Fig. 5: Confusion matrix demonstrating the repetition detection accuracy of the Edge-AI architecture.

4.3.2 Computational efficiency: Edge vs. cloud architectures

A key objective of this study was to investigate the feasibility of performing biomechanical analysis entirely on local edge hardware. The experimental implementation maintained real-time responsiveness while processing pose estimation, geometric calculations, repetition validation, and feedback generation locally. By eliminating dependence on cloud-based computation, the system reduces network latency and preserves user privacy. The incorporation of the One-Euro Filter further improved system stability by reducing landmark jitter and smoothing rapid fluctuations in pose estimation outputs. This contributed to more consistent joint-angle calculations and improved robustness under normal indoor operating conditions.

4.3.3 The semantic layer: Generative ai utility

Beyond repetition counting, the integration of a local Large Language Model enables the generation of contextual coaching feedback based on validated exercise metrics. Performance indicators such as repetition count, range of motion, and movement consistency are converted into structured inputs for the

Table 1: Results during field testing.

Field	True Cases		False Cases		% Error	% Success
Trial	TN	TP	FN	FP		
1	0	4	1	0	20	80
2	0	5	0	0	0	100
3	0	4	1	0	20	80
4	0	4	1	0	20	80
Total	0	17	3	0	-	-
Average	-	-	-	-	15.0	85.0

Table 2: Literature-based comparison of accuracy and latency across different frameworks.

Model name	Accuracy (%)	End-to-end latency	Hardware required
VGG16 (Cloud)	86.21%	520 ms	High-end GPU
GoogleNet	79.23%	480 ms	Cloud Server
MediaPipe (Raw)	91.00%	45 ms	CPU / Mobile
Proposed System (FSM + Llama)	85.00%	42 ms	Local PC / i5

Note: The values reported for VGG16, GoogleNet, and MediaPipe are obtained from previously published literature and are included solely for qualitative comparison. Direct experimental comparison under identical testing conditions was not performed in this study.

language model, allowing the system to provide personalized recommendations. This approach transforms the system from a conventional exercise counter into an intelligent fitness assistant capable of delivering user-specific guidance while maintaining complete local processing of biometric data.

4.3.4 Comparative analysis with existing models

Table 2 presents a comparative overview of selected computer vision and pose-estimation frameworks reported in the literature. The comparison includes recognition accuracy, end-to-end latency, and hardware requirements. These metrics provide insight into the trade-offs between computational complexity, response time, and deployment feasibility for real-time fitness monitoring applications.

As shown in Table 2, cloud-based approaches may provide competitive recognition performance but generally require greater computational resources and network connectivity. In contrast, the proposed framework is designed for local deployment and real-time operation on consumer-grade hardware. Although the reported repetition-validation accuracy of the proposed system is 85.0%, the integration of deterministic FSM-based validation and local AI feedback generation enables reliable biomechanical assessment while preserving user privacy. Since all processing is performed on the edge device, sensitive video and performance data remain within the local environment, reducing dependence on external cloud services.

4.3.5 Future scope

Several opportunities exist for extending the proposed system. Future work may incorporate additional exercises involving lower-body biomechanics, including squats, lunges, and deadlifts. The inclusion of larger and more diverse datasets could improve generalization across users with different body structures and exercise styles. Further optimization through GPU or Neural Processing Unit (NPU) acceleration may improve inference performance and support multi-user environments. Additionally, advanced temporal prediction techniques may help reduce tracking failures caused by self-occlusion and challenging viewing angles, thereby improving overall system robustness. A major limitation of the present study is the relatively small evaluation dataset. Future work will include testing across a larger participant pool, diverse body types, lighting conditions, and multiple exercise categories to improve statistical validity and generalization.

5. Conclusion

This research successfully developed and implemented an edge-computing-based framework for real-time biomechanical posture analysis and exercise monitoring. By integrating MediaPipe BlazePose for landmark extraction, a Deterministic Finite State Machine (FSM) for repetition validation, and a local Meta Llama 3 inference engine for personalized feedback generation, the proposed system provides a privacy-preserving solution for intelligent fitness assistance. Experimental evaluation conducted on 20 manually performed biceps curl repetitions demonstrated an accuracy of 85.0%, a precision of 100%, a recall of 85.0%, and an F1-score of 91.89%. The results indicate that the FSM-based validation mechanism effectively eliminates false-positive repetition counts while maintaining reliable exercise tracking performance. The strict state-transition logic ensures that only biomechanically valid repetitions are recorded, thereby reducing errors caused by incomplete movements and momentum-assisted lifting. Furthermore, the localized execution environment achieved real-time responsiveness with low processing latency, demonstrating the feasibility of performing posture analysis, repetition validation, and AI-assisted feedback generation entirely on consumer-grade hardware without dependence on cloud services. In summary, the proposed system demonstrates that the combination of computer vision, deterministic biomechanical validation, and local generative AI can provide an effective virtual fitness assistant while preserving user privacy. Future enhancements may include support for additional exercises, multi-user tracking, and improved robustness under challenging environmental conditions.

Acknowledgement
The authors would like to express their sincere gratitude to the Department of Computer Engineering, M. H. Saboo Siddik College of Engineering, Mumbai, for providing the facilities, guidance, and support necessary to carry out this research. The authors also thank all the volunteers who participated in the testing and evaluation of the proposed system.

CRediT Author Statement
Fatima Anees Ansari: Conceptualization, Supervision, Writing - Review and editing, **Hussain Siddique:** Methodology, Software, Validation, **Zaid Shaikh:** Data curation, Methodology. **Zunaid Siddiqui:** Methodology, Software, Writing – Original draft. All authors have read and agreed to the published version of the manuscript.

Funding Declaration
This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data Availability Statement
The data used in this study were generated during the experimental evaluation of the proposed system. Due to the limited scope of the study and privacy considerations associated with video-based exercise recordings, the datasets are available from the corresponding author upon reasonable request.

Consent for Publication
The individual appearing in the figures of this manuscript provided informed consent for the publication of the images.

Conflict of Interest
There is no conflict of interest.

Artificial Intelligence (AI) Use Disclosure
The authors declare that artificial intelligence (AI)-assisted tools were used only for language refinement, grammar improvement, and manuscript structuring purposes during the preparation of this work. All technical content, experimental implementation, results, and interpretations were independently developed and verified by the authors.

Supporting Information
Not applicable.

References
[1] R. R. Kanase, A. N. Kumavat, R. D. Sinalkar, S. Somani, Pose estimation and correcting exercise posture, *ITM Web of Conferences*, 2021, **40**, 03031, doi: 10.1051/itmconf/20214003031.

[2] S. H. Johnston, M. F. Berg, S. W. Eikevåg, D. N. Ege, S. Kohtala, M. Steinert, Pure vision-based motion tracking for data-driven design - a simple, flexible, and cost-effective approach for capturing static and dynamic interactions, *Proceedings of the Design Society*, 2022, ***2***, 485-494, doi: 10.1017/pds.2022.50.

[3] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. Zhu, F. Zhang, M. Grundmann, BlazePose: On-device real-time body pose tracking, arXiv preprint, 2020, doi: 10.48550/arXiv.2006.10204.

[4] H. Kotte, M. Kravčík, N. Duong-Trung, Real-time posture correction in gym exercises: A computer vision-based approach for performance analysis, error classification and feedback, CEUR Workshop Proceedings, 2023, ***3499***, 64-70, https://ceur-ws.org/Vol-3499/paper9.pdf.

[5] M. Kaushik, N. Vithyatharshana, M. Kandala, S. Palaniswamy, G. S. Vignesh, AI-based posture correction, real-time exercise tracking and feedback using pose estimation technique, 2024 International Conference on Communication, Control, and Intelligent Systems (CCIS), IEEE, 2024, 1-6, doi: 10.1109/CCIS63231.2024.10932054.

[6] J. W. Kim, J. Y. Choi, E.J. Ha, J.H. Choi, Human Pose Estimation Using MediaPipe Pose and Optimization Method Based on a Humanoid Model, *Applied Sciences*, 2023, **13**, 2700, doi: 10.3390/app13042700.

[7] J. Y. Choi, E. Ha, M. Son, J. H. Jeon, J. W. Kim, Human joint angle estimation using deep learning-based three-dimensional human pose estimation for application in a real environment, *Sensors*, 2024, **24**, 3823, doi: 10.3390/s24123823.

[8] G. Casiez, N. Roussel, and D. Vogel, 1€ Filter: A Simple Speed-based Low-pass Filter for Noisy Input in Interactive Systems, Proceedings of the

CHI Conference on Human Factors in Computing Systems, Association for Computing Machinery, 2012, 2527-2530, doi: 10.1145/2207676.2208639.

[9] P. K. Nguyen, A.T. Nguyen, T. B. Doan, P. N. Trung, N. D. Thi, Assessing bicep curl exercises by human pose application: a preliminary study, In International Conference on Soft Computing and Pattern Recognition, Cham: Springer Nature Switzerland, 2022, 581-589, doi: 10.1007/978-3-031-27524-1_55.

[10] Meta AI, Llama 3: Open Foundation and Fine-Tuned Chat Models, Meta AI Research Documentation, 2024, https://ai.meta.com/llama/, 23 April 2026.

[11] FastAPI Framework, High performance, easy to learn, fast to code, ready for production, 2025, https://fastapi.tiangolo.com/, Accessed 27-Feb-2026.

[12] Ollama, Get up and running with large language models locally, 2025, https://ollama.com/, Accessed 27-Feb-2026.

[13] G. Bradski, The OpenCV Library, Dr. Dobb's Journal of Software Tools, 2000, **120**; 122-125.

[14] Q. Yu, H. Wang, F. Laamarti, A. E. Saddik, Deep learning-enabled multitask system for exercise recognition and counting, *Multimodal Technologies and Interaction*, 2021, **5**, 55, doi: 10.3390/mti5090055.