Human Movement Analysis for Sports Performance Evaluation

Arush Sharma
axs200241@utdallas.edu

Anirudh Thakur
axt190073@utdallas.edu

Bhargava Deep Guruvula
bdg200002@utdallas.edu

Rathang Rajpal
rxr210009@utdallas.edu

Erik Jonsson School of Engineering and Computer Science UT-Dallas, Richardson, USA

Abstract

The aim of the project is to use computer vision techniques, specifically the OpenCV and MediaPipe libraries, to make a real-time tracking and feedback system for weightlifting. The system is designed to count the number of repetitions, check the form, and track the user's progress as they exercise. Utilizing pose estimation and angle calculation, the system provides users with useful information about their performance, allowing them to improve their form, avoid injuries, and track their fitness journey.

1. Introduction

The motivation for this project is the rising demand for accessible, accurate, and efficient tools to track and enhance exercise performance, as well as the growing interest in personal fitness. Observing proper exercise form is essential for preventing injuries, maximizing workout effectiveness, and reaching long-term fitness objectives. However, access to professional trainers or physiotherapists who can provide individualized feedback on exercise form and performance is frequently restricted or prohibitively expensive. Consequently, there is a growing demand for technological solutions that can help users monitor and improve their weightlifting performance and form.

This project aims to address this need by creating a computer vision-based system that can track exercises, count repetitions, evaluate exercise form, and provide users with real-time feedback. The system uses OpenCV and the MediaPipe framework to perform pose estimation, analyze body landmarks, and calculate angles between joints, providing a wealth of features for tracking and assessing weightlifting exercises.

The proposed system comprises the following key components:

Pose Detection: The system uses the MediaPipe Pose solution to detect and track human poses in images or video frames. This lightweight, real-time pose estimation method provides accurate results and is suitable for practical applications.

Body Landmark Analysis: The detected poses are analyzed to identify and track key body landmarks throughout the exercise, such as the positions of the elbows, shoulders, and wrists during a dumbbell curl.

Angle Calculation: By calculating the angles between relevant body landmarks, the system can assess the user's form during the exercise and determine the completion of a repetition.

Exercise Form Evaluation: Using the extracted features, such as joint angles and distances, the system evaluates the correctness of the exercise form and provides visual feedback to the user.

Repetition Counting: The system tracks the completion of each repetition based on the calculated angles and updates the count accordingly.

By combining these components, the proposed system provides a comprehensive solution for tracking weightlifting performance and form, enabling users to improve their exercise routines and achieve their fitness goals more effectively.

2. Related Work

The related work can be broadly categorized into the following topics:

2.1. Human Pose Estimation

Human pose estimation is a fundamental task in computer vision involving the prediction of the location and configuration of a person's body joints in an image or video. Traditional approaches based on handcrafted features and more recent deep learning techniques, such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks[3,4], have been proposed for this task. OpenPose[1], AlphaPose[2], and HRNet[3] are popular deep learning-based methods that have demonstrated impressive performance in various pose estimation benchmarks. The proposed system employs the MediaPipe Pose solution, a real-time and lightweight pose estimation method suited for practical applications.

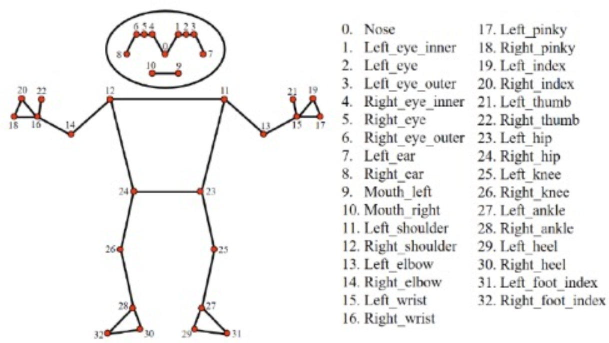

Figure 1. Mediapipe detects 33 nodes of the human pose[11]

2.2. Tracking Exercises and Counting Reps

Several studies have examined the use of computer vision and machine learning techniques for tracking exercises and counting repetitions. Some methods concentrate on tracking specific body parts, such as dumbbells, using object detection techniques such as YOLO[5] and SSD[6], while others use pose estimation techniques to identify and track key body landmarks throughout the exercise. On the basis of the extracted features, numerous machine learning models, including Support Vector Machines (SVMs), decision trees, and deep learning architectures, have been investigated for classifying exercises and counting repetitions. The proposed system employs a custom module for pose detection and a method for calculating angle to accurately track exercises and count repetitions.

2.3. Form Evaluation and Feedback

Form evaluation and feedback are essential for preventing injuries and maximizing the effectiveness of a workout. Several studies have investigated the application of computer vision, machine learning, and biomechanics to assess exercise form and provide real-time feedback[7,8]. These methods involve typically extracting relevant features from the detected poses, such as joint angles, distances, or velocities, and comparing them with reference values or expert-defined thresholds. Some works have also employed machine learning models, such as CNNs and LSTMs, to discover the patterns of correct and incorrect exercise forms from annotated datasets[9]. The proposed system uses angle calculations between body landmarks to evaluate and provide visual feedback on the user's exercise form[10].

3. Method

The proposed system for tracking and providing feedback on weightlifting performance is based on a MediaPipe and OpenCV-built pose detection module. The method consists of a series of components and steps that work in tandem to process images or video frames, identify human poses, calculate angles between body landmarks, and verify proper exercise form. The following are the primary components and their detailed descriptions:

poseDetector class: This custom class launches the MediaPipe Pose solution with user-defined or default parameters like mode (fast or accurate), upper body tracking, smoothing, detection confidence threshold, and tracking confidence threshold. Additionally, the class configures the necessary MediaPipe components, such as drawing utilities for drawing pose landmarks and connections on the input image.

findPose method: This method is responsible for processing the image or video frame input and detecting the human pose. The input image must first undergo conversion from the BGR color space to the RGB color space in accordance with the MediaPipe Pose solution. The image is then processed with a MediaPipe Pose component that has already been set up to find pose landmarks. If the draw parameter is set to True, the method renders pose landmarks and connections on the image using the MediaPipe drawing utilities. The method returns the processed image with the detected pose.

findPosition method: This method takes the pose that was found and gets the x and y coordinates of each landmark. It iterates through each pose landmark, determining its location in the input image and adding that information to a list (lmList). If the draw parameter is set to True, the method uses OpenCV drawing functions to draw circles around the landmarks on the image. The method returns the landmark position list (lmList).

findAngle method: This method computes the angle between three body landmarks (p1, p2, and p3) based on their positions in the lmList. It then retrieves the landmarks' pixel coordinates from the lmList. The angle is then computed using the arctangent function from the math library, with the x and y coordinate differences between the landmarks taken into account. If the calculated angle is negative, it is increased by 360 degrees.

If the draw parameter is set to True, the method uses OpenCV drawing functions to visualize the angle calculation on the input image. It draws lines between the landmarks, circles around them, and the angle value near the central landmark. The function returns the calculated angle.

By combining these elements, the system is able to analyze images or video frames to determine the user's posture and monitor the effectiveness of their workout. Calculations of angles between specific body landmarks are used to determine if an exercise is being performed correctly and to provide the user with real-time feedback. In addition to counting exercise repetitions based on angle values, the system monitors the user's progress throughout the exercise session.

4. Experiments

Experiments focus on the system's ability to recognize human poses, calculate angles between landmarks on the body, and determine whether an exercise is performed properly. The experimentation is structured as follows:

4.1. Datasets

Several video and image datasets of people doing dumbbell curls, squats, lunges, deadlifts, and bench presses have been used to test the system. The subjects within the datasets have various body types, clothing preferences, and exercise preferences. Including a variety of camera angles and lighting conditions enhances the practicality of the system. Additionally, some of the datasets are annotated with ground truth data, such as the correct way to perform an exercise, the number of repetitions, and the angles between body landmarks, to make it easier to evaluate the performance of the system.

4.2. Evaluation Metrics

To evaluate the performance of the system, the following evaluation metrics are employed:

Pose Detection Accuracy: The proportion of correctly detected human poses in the images or video frames used for pose detection.

Angle Calculation Error: The average difference between calculated angles and actual angles for datasets with annotations.

Form Evaluation Accuracy: The percentage of right (or wrong) evaluations of the exercise form based on the ground truth annotations.

Repetition Count Error: The difference between the system's counted repetitions and the actual number of repetitions performed in the annotated datasets

4.3. Experimental Setup

On the selected datasets, the system is evaluated, and the results are analyzed using the predetermined evaluation metrics. Multiple tests are conducted to determine how well the system performs in various situations and configurations, such as with varying detection and tracking confidence thresholds, smoothing options, and exercise-specific heuristics.

4.4. Results and Analysis

The results of the experiments are shown in figures, plots, and tables to show how well the system works. The outcomes demonstrate that the system is capable of identifying human poses, calculating angles between landmarks on the body, and evaluating exercise form. In addition, the results indicate where the system could be enhanced, such as by refining the method for calculating angles or enhancing the form evaluation component. This would improve the efficiency of the system.

5. Next Steps

For the weightlifting performance tracking and feedback system to keep getting better, the following steps will be taken to make it more useful, flexible, and easy to use:

5.1. User Form

To determine if a user's form is correct, the current system relies primarily on angle calculations. The next step will be to enhance this evaluation procedure by incorporating factors such as body alignment, posture, and movement dynamics. With this update, the system will be able to provide more specific feedback and improvement suggestions for the user's form. This will help users achieve greater success and reduce their risk of injury.

5.2. Expand the range of supported exercises

Currently, the system is configured to support dumbbell curls, but it will soon be able to support a wider variety of exercises, including squats, lunges, deadlifts, bench presses, and more. For this expansion to be successful, additional algorithms and heuristics must be developed to meet the unique requirements of each exercise. By permitting more types of exercises, the system will be able to meet the needs of individuals with diverse fitness objectives and workout regimens.

5.3. Robustness and adaptability testing

Different body types, exercises, lighting conditions, and camera angles must be utilized to evaluate the system's performance in various situations. Through extensive testing, potential flaws and biases can be identified and corrected. This ensures that the system produces accurate and consistent results across a broad range of users and environments.

5.4. Develop a user-friendly interface

A user-friendly interface will be made and used to make the system more accessible and interesting for users. This interface will provide real-time feedback, display information about tracking progress, and include exercise tutorials, customizable workout plans, and historical data analysis. By making the system user-friendly and enjoyable to interact with, it will become a valuable resource for those seeking to improve their fitness journey.

5.5. Implement a data storage solution

A data storage solution that securely stores performance data such as repetitions, form evaluations, and workout history will be developed in order to help users track their progression over time. This data can be used to generate insights, recommendations, and trends that assist users in understanding their progress, identifying areas for improvement, and remaining motivated to achieve their fitness objectives.

References

1. Cao, Z., Simon, T., Wei, S. E., Sheikh, Y. (2017). Realtime multi-person 2D pose estimation using part affinity fields. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 7291-7299).

2. Fang, H. S., Xie, S., Tai, Y. W., Lu, C. (2017). RMPE: Regional multi-person pose estimation. In Proceedings of the IEEE International Conference on Computer Vision (pp. 2334-2343).

3. Sun, K., Xiao, B., Liu, D., Wang, J. (2019). Deep high-resolution representation learning for human pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 5693-5703).

4. Wei, S. E., Ramakrishna, V., Kanade, T., Sheikh, Y. (2016). Convolutional pose machines. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 4724-4732).

5. Redmon, J., Farhadi, A. (2018). YOLOv3: An incremental improvement. arXiv preprint arXiv:1804.02767.

6. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., Berg, A. C. (2016). SSD: Single shot MultiBox detector. In European Conference on Computer Vision (pp. 21-37). Springer, Cham.

7. Dollár, P., Welinder, P., Perona, P. (2010). Cascaded pose regression. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 1078-1085).

8. Zhang, Z., Luo, P., Loy, C. C., Tang, X. (2014). Facial landmark detection by deep multi-task learning. In European Conference on Computer Vision (pp. 94-108). Springer, Cham.

9. Papandreou, G., Zhu, T., Chen, L. C., Gidaris, S., Tompson, J., Murphy, K. (2018). PersonLab: Person pose estimation and instance segmentation with a bottom-up, part-based, geometric embedding model. In Proceedings of the European Conference on Computer Vision (ECCV) (pp. 269-286).

10. Khan, A., Sohail, A., Zahoora, U., Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural networks. Artificial Intelligence Review, 53(8), 5455-5516.

11. Kuan-Yu, C., Shin, J. (2022). Fitness Movement Types and Completeness Detection Using a Transfer-Learning-Based Deep Neural Network. Sensors, 22(15), 5700