Vol. 11 No.2 Tahun 2025, pp 110-119
Jurnal Teknologi dan Manajemen Informatika
http:// http://jurnal.unmer.ac.id/index.php/jtmi
P-ISSN: 1693-6604 E-ISSN: 2580-8044

Application of YOLO11 and Long Short-Term Memory Architecture for Exercise Form Evaluation in Weightlifting

Nicholas Dylan Lienardi [1], Evan Tanuwijaya [2]

Informatika, Fakultas Teknologi Informasi, Universitas Ciputra, Surabaya, Indonesia

Article Info

Article History
Received : 28-08-2025
Revised : 23-10-2025
Accepted : 05-11-2025

Keywords
Pose analysis;
YOLO;
LSTM;
Weightlifting

Corresponding Author
Evan Tanuwijaya
Universitas Ciputra
evan.tanuwijaya@ciputra.ac.id

ABSTRACT

Exercise provides significant benefits for physical health, and weightlifting has become increasingly popular among fitness enthusiasts. However, improper lifting techniques often lead to injuries, discouraging beginners and affecting long-term training consistency. To address this issue, this study proposes a deep learning approach that automatically evaluates weightlifting form through movement classification. The proposed method integrates the YOLO11n-pose algorithm for detecting keypoints from exercise video recordings and the Long Short-Term Memory (LSTM) network for classifying movement types and determining the correctness of form execution. The model achieved a mean average precision of 88.8% using side-view recordings of single-repetition weightlifting exercises. YOLO11n-pose extracts the coordinates of body keypoints, which are converted into joint angle data and analyzed over time using LSTM to identify movement quality based on expert-validated training data. The trained model was implemented into an iOS application called KorForm, developed using FastAPI, to provide real-time feedback for users. The results demonstrate that combining YOLO11n-pose and LSTM effectively supports weightlifting form evaluation and offers a practical solution for promoting safer and more consistent exercise habits.

INTRODUCTION

Regular physical activity is essential for maintaining both physical fitness and mental well-being. Despite this, more than 80% of adolescents worldwide remain insufficiently active, leading to an estimated 20–30% higher risk of premature mortality [1]. In practice, beginning a fitness routine is often challenging, especially for beginners who struggle with both technical execution and psychological barriers. One of the main difficulties lies in performing movements with proper posture [11]. Common mistakes, such as rounding the back during a deadlift or allowing the knees to drift inward during a squat, can increase the likelihood of injury and reduce the overall effectiveness of training [2]. Exercises like strength training and high-intensity interval training (HIIT) are particularly prone to such risks, especially among those who have not yet mastered basic techniques [3].

While guidance from a professional trainer can help reduce these risks, it is not always accessible or affordable. Moreover, personal factors—such as lack of confidence or fear of being judged—often discourage consistent practice. To overcome these issues, an automated system that provides real-time feedback on movement quality could serve as an effective solution. With recent progress in Human Action Recognition (HAR), researchers have explored various methods to evaluate exercise performance automatically. For instance, a YOLO v7-Pose model enhanced with Coordinate Attention and ConvNeXt modules achieved a mean

average precision (mAP) of 95.9% for detecting joint angles in bodyweight exercises [4]. Other studies using CNN-based frameworks, such as OpenPose and MediaPipe, have successfully identified posture errors like back curvature or knee misalignment, though performance varied depending on the algorithm and environmental factors such as lighting [5].

In addition, hybrid models combining Convolutional Neural Networks (CNN) [12] and Long Short-Term Memory (LSTM) networks [13], [14] have proven effective in recognizing movement patterns from video sequences. One study examining the traditional Chinese exercise Baduanjin demonstrated that a VGG16-LSTM architecture achieved 96.43% accuracy, outperforming handcrafted feature-based methods [6]. The strength of combining CNNs for spatial feature extraction with LSTMs [7], [8] for temporal sequence learning lies in their ability to model complex and continuous human motions.

Building on these insights, this research adopts a CNN-LSTM framework [9] integrated with the YOLO architecture [10], [15] for exercise motion classification. This approach leverages the complementary strengths of spatial detection and temporal pattern analysis to deliver accurate and efficient form evaluation. In this study, the proposed method is applied to assess deadlift and squat movements among beginner users through a mobile-based application. The system is designed to assist users in identifying and correcting improper form independently, ultimately promoting safer and more consistent training habits.

METHODS

The dataset used in this study consists of video recordings labelled with the type of fitness exercise and posture classification—either correct or incorrect. This dataset was collected firsthand, comprising original recordings of barbell back squat and deadlift exercises that were validated by a certified expert. A previous study conducted in Taif, Saudi Arabia, identified the most injury-prone areas among weightlifting athletes, listed in descending order of frequency as the shoulders, knees, upper and lower back, and wrists [2]. Insights from a professional fitness practitioner, Iwan Ragil, further supported these findings. Based on his observations, clients and gym members most experienced injuries in the back and knee regions. Discussions with him led to the selection of deadlift and squat movements as the focus of this research, as these exercises are the most likely to contribute to injuries in those areas. Consequently, the classification categories for this study were defined as Correct Squat, Incorrect Squat, Correct Deadlift, and Incorrect Deadlift.

Data collection was conducted in collaboration with Iwan Ragil, a certified personal trainer at Family Clubhouse Citraland, Surabaya. With 16 years of experience in fitness and 8 years as a professional trainer, he holds certification from Ray Institute and is qualified to demonstrate both correct and incorrect exercise forms for the barbell back squat and deadlift. All recordings were captured using an iPhone 15 equipped with a 12 MP True Depth camera, set to HD resolution at 30 frames per second (FPS).

Each video captured the full body of the participant from a side view to ensure clear visualization of the movement. Recordings covered one full set of either squats or deadlifts, consisting of 10 repetitions per video, with each repetition lasting approximately three seconds to capture sufficient motion detail. During the preprocessing phase, each video was segmented into 10 individual clips, each representing a single repetition. Every clip was manually labelled as correct or incorrect based on the performer's posture and movement tempo. The labelling process was validated by the expert trainer to ensure accuracy and alignment with proper exercise standards like Figure 1 and 2.

A total of 84 videos were collected—21 recordings per class—which were further segmented into 840 short clips lasting 2–3 seconds each. This resulted in 210 clips for each of the four movement classes: Correct Squat, Incorrect Squat, Correct Deadlift, and Incorrect Deadlift. The dataset was then divided into three subsets: 70% for training, 20% for validation, and 10% for testing, to support model development and performance evaluation.

Figure 1. Example of Correct (Left) and Incorrect (Right) Squat Form
Figure 2. Example of Correct (Left) and Incorrect (Right) Deadlift Form

To develop a model capable of recognizing human actions during exercise using body keypoints and joint angle calculations, the first step involves extracting keypoints from video frames with the pre-trained YOLO11n-pose model. This model is designed to detect 17 keypoints on the human body, including the head, shoulders, elbows, wrists, hips, knees, and ankles. However, not all these keypoints are equally important for every type of exercise; therefore, it is essential to focus on the most relevant ones for the specific movements being analyzed. In this study, four keypoints were selected as the most critical for identifying movement patterns and potential injury risks. According to expert feedback, deadlift and squat movements should be analyzed from a side view to clearly observe the alignment of the knees, back, feet, and shoulders. Based on this consideration, the keypoints chosen for this research are the shoulder, hip, knee, and ankle.

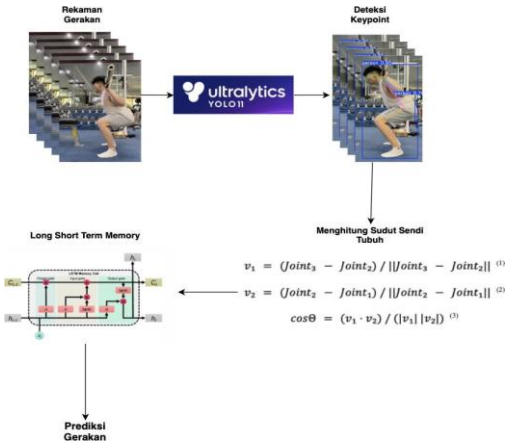

Figure 3. Machine Learning Model System Pipeline

$$V_1 = (Joint_3 - Joint_2) \,/\, ||\, Joint_3 - \, Joint_2||$$

$$V_2 = (Joint_2 - Joint_1) \,/\, ||\, Joint_2 - \, Joint_1||$$

$$\cos\theta = (V_1.\ V_2) \,/\, |(V_1 \mid \mid V_2)|$$

Figure 3 illustrates the proposed machine learning model pipeline. After extracting the keypoints using the pre-trained YOLO11n-pose model, the next step is to calculate the joint angles for each video frame. This process involves measuring the angle between adjacent keypoints that represent body joints. For instance, to calculate the knee angle, the hip, knee, and ankle keypoints are used to form vectors, and the angle between these vectors is computed. The same approach is applied to all relevant joints involved in the exercise. As a result, a sequence

of joint angles is obtained for each video frame. These angles are calculated through vector operations, where the *x* and *y* coordinates of each joint form the basis of the vectors.

The angles between body segments are calculated based on the coordinates of detected joint points. Vector V_1 represents the direction from $Joint_2$to $Joint_3$, computed as $V_1 = (Joint_3 - Joint_2)/||\ Joint_3 - Joint_2\ ||$. Vector V_2 represents the direction from $Joint_1$to $Joint_2$, defined as $V_2 = (Joint_2 - Joint_1)/||\ Joint_2 - Joint_1\ ||$. Both vectors are normalized to ensure that the calculation focuses solely on direction rather than the actual segment length. Vector V_3 represents $\cos\theta = (V_1 \cdot V_2)/(||\ V_1\ ||\ \ ||\ V_2\ ||)$, is then used to determine the angle θbetween the two segments, where $V_1 \cdot V_2$represents the dot product of the two vectors. The resulting angle θcaptures the relative movement between joints, serving as a key feature for motion analysis.

The computed joint angles are stored in a tensor format, where the dimensions correspond to the number of samples (exercise repetitions), the number of angles per frame (based on the selected joints), and the number of frames per repetition. This tensor is then fed into the LSTM model, which learns the temporal patterns of joint movements over time—distinguishing correct forms from incorrect ones based on alignment, range of motion, and rhythm. By training the LSTM on sequences of joint angles, the model can effectively differentiate between proper and improper postures for each exercise repetition.

Once model training is completed, the next step involves integrating the model into an application via FastAPI. Through this integration, the mobile application can send captured images or videos to the server for analysis by the trained model and receive classification results in real-time. This enables efficient and seamless inference through a connected backend.

The computer vision architecture is implemented within an iOS-based mobile system, utilizing YOLO11n-pose, vector angle operations, and LSTM to generate analysis results that are displayed directly to the user. The user interface is developed using SwiftUI, while Apple's PhotoKit framework is employed to manage and store user workout videos within the device's gallery. These stored videos are processed by the YOLO11n-pose model to detect and extract keypoint information, which is then used to calculate body joint angles. The resulting data are passed through the LSTM architecture to perform action classification. The final classification results are displayed on the mobile screen, allowing users to view and assess their exercise form. The entire analysis and classification process is executed through FastAPI integration, providing a real-time feedback loop between the deep learning model and the mobile application. An overview of this architecture is illustrated in the corresponding figure.

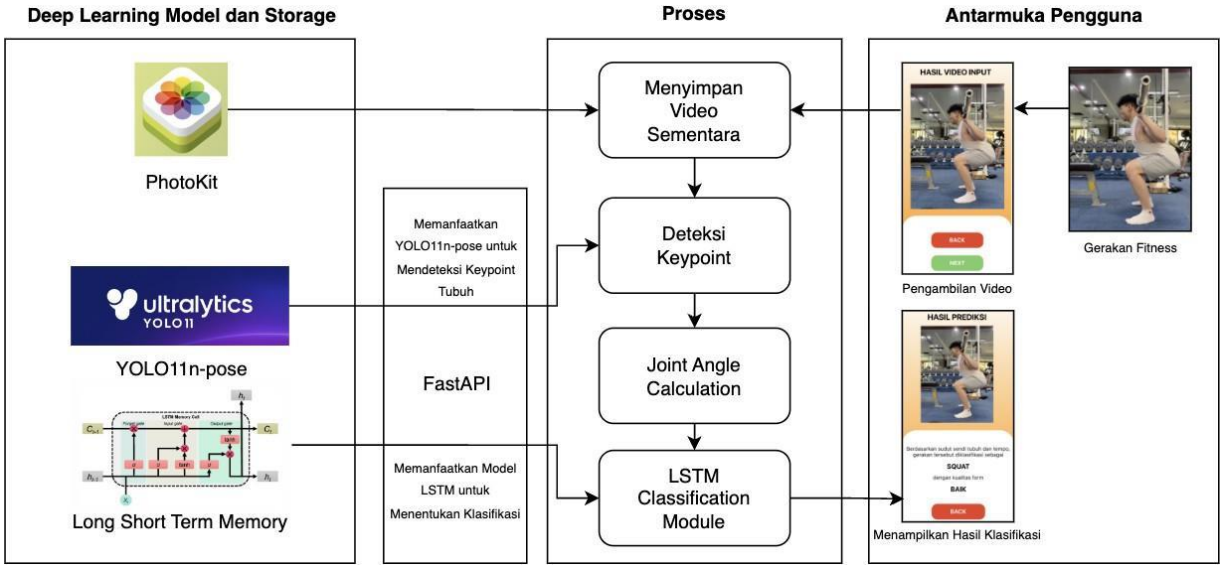

Figure 4. System Architecture Design

The following describes the testing design for the developed application. The evaluation will be conducted through an alpha testing phase using the previously constructed dataset. The primary focus of this testing is to evaluate the performance of the trained LSTM model in recognizing weightlifting movements based on features extracted from exercise videos.

To assess the model's accuracy, several evaluation metrics will be employed. The analysis begins with a confusion matrix, which provides a detailed breakdown of classification errors. This matrix includes four key components: True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN). Based on these components, the following metrics are calculated:

Precision measures the proportion of correctly predicted positive instances out of all positive predictions, indicating how reliable the positive classifications are.

$$Precision = TP \;/\; (TP + FP)$$

Recall (Sensitivity) measures how effectively the model identifies actual positive cases, showing the model's ability to capture all relevant instances.

$$Recall = TP \;/\; (TP + FN)$$

F1-Score represents the harmonic mean between Precision and Recall, providing a balanced measure of the model's performance—particularly useful when dealing with imbalanced datasets.

$$F1score = 2 * ((Precision * Recall))/(Precision + Recall)$$

Accuracy evaluates the overall performance of the model across all classes during both training and testing phases.

$$Accuracy = (TP + TN)/(TP + TN + FP + FN)$$

In addition to these, Average Precision (AP) will also be utilized. AP summarizes the Precision–Recall (PR) curve into a single value, effectively evaluating the model's performance across multiple threshold levels. Furthermore, Model Loss (derived from the specific loss function) and Accuracy (as defined above) will be continuously monitored throughout the training and testing processes to ensure that the model does not experience overfitting or underfitting.

After computing all evaluation metrics, an additional analysis will be performed to assess the model's accuracy in distinguishing between Deadlift and Squat movements. For this evaluation, the labels Correct Deadlift and Incorrect Deadlift will be grouped under a single Deadlift category, while Correct Squat and Incorrect Squat will be grouped under Squat. This approach allows the assessment of how effectively the model differentiates between exercise types, regardless of execution quality.

RESULTS AND DISCUSSION

This section presents the implementation results and performance analysis of the developed system. Figure 5 illustrates the keypoint detection results generated by the YOLO11n-pose model on human motion images. The model successfully identified and marked the main body joints—such as the head, shoulders, elbows, knees, and ankles—with high confidence scores. This stage serves as an essential foundation for the feature extraction process, as the detected keypoint coordinates are subsequently used to calculate joint angles, which are later analyzed for movement prediction using the machine learning model.

Figure 5. Keypoint Detection Visualization

To evaluate the model's performance, several metrics were employed, including the confusion matrix, precision, recall, AP, F1-score, loss, and accuracy. Beyond assessing whether movements were performed correctly or incorrectly, the evaluation also examined the model's ability to distinguish between different exercise types through label grouping (superset analysis). This approach provides a comprehensive understanding of the model's accuracy, robustness, and stability, particularly when dealing with imbalanced datasets.

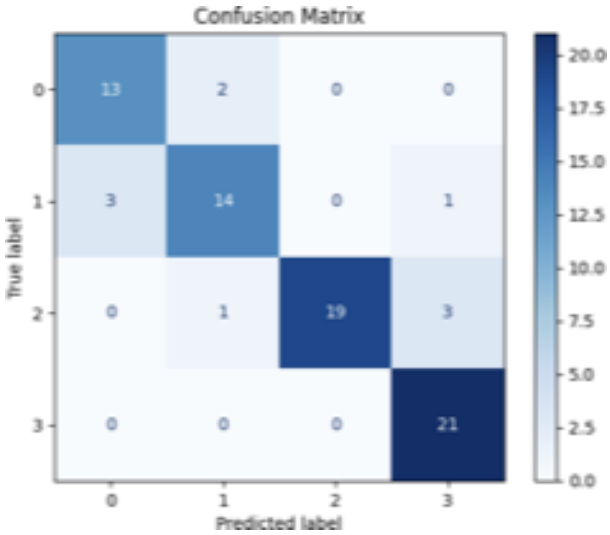

Figure 6. Confusion Matrix

As shown in Figure 6, the class labels were defined as follows: 0 for Incorrect Deadlift, 1 for Correct Deadlift, 2 for Incorrect Squat, and 3 for Correct Squat. The model demonstrated strong classification performance across all categories, correctly identifying Incorrect Deadlift in 13 out of 15 cases (86.7%), Correct Deadlift in 14 out of 18 cases (77.8%), Incorrect Squat in 19 out of 23 cases (82.6%), and Correct Squat in all 21 cases (100%). These results indicate that the model performs effectively in distinguishing between different movement types, with particularly high accuracy in recognizing correctly executed squat movements.

However, several misclassifications were also observed. Incorrect Deadlift was misclassified as Correct Deadlift twice, while Correct Deadlift was misclassified as Incorrect Deadlift three times and as Correct Squat once. This suggests that the model still faces challenges in differentiating between correct and incorrect executions of the same exercise type, especially for deadlifts. Additionally, Incorrect Squat was misclassified once as Correct

Deadlift and three times as Correct Squat, indicating that the model occasionally interprets improper squats as either correct movements or as belonging to a different exercise category.

Table 1. Model Classification Performance Report

Class	Precision	Recall	F1-Score	Support
0 – Incorrect Deadlift	0.81	0.87	0.84	15
1 – Correct Deadlift	0.82	0.78	0.80	18
2 – Incorrect Squat	1.00	0.83	0.90	23
3 – Correct Squat	0.84	1.00	0.91	21
Overall Accuracy	**–**	**–**	**0.87**	**77**
Macro Average	**0.87**	**0.87**	**0.86**	–
Weighted Average	**0.88**	**0.87**	**0.87**	–

The performance metrics summarized in Table 1, derived from the test dataset containing 77 video samples from the train–test–validation split, show that the model achieved an AP of 87%. Precision varied among classes, with Incorrect Deadlift recording the lowest precision (81%) and Correct Squat achieving the highest (100%). These findings demonstrate that the model is more effective in identifying squat-related movements, as reflected in the higher precision scores for labels 2 and 3 (Incorrect Squat and Correct Squat). Conversely, Correct Deadlift (Label 1) showed the lowest recall, suggesting that this movement remains the most challenging for the model to classify consistently.

Table 2. Summary of Overall Model Evaluation Metrics

Metric	Value
Precision (Macro)	0.8690
Recall (Macro)	0.8676
F1-Score (Macro)	0.8641
Average Precision (AP, Macro)	0.8888
Superclass Accuracy	0.9481

Overall, from table 2, the model achieved an impressive accuracy of 94.81% in classifying both Deadlift and Squat movements, whether performed correctly or incorrectly. Within the validation dataset, only two misclassifications were recorded: one Correct Deadlift identified as Correct Squat and one Correct Squat identified as Correct Deadlift.

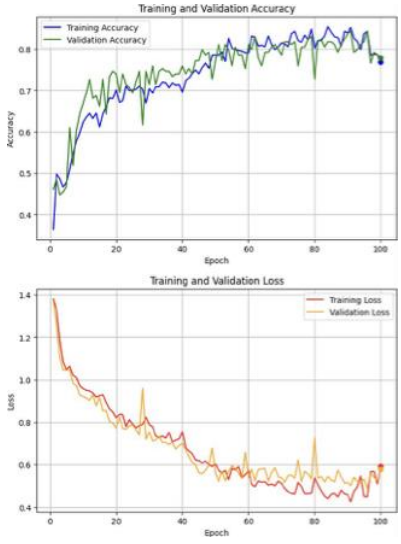

Figure 7. Loss and Accuracy Training

As shown in Figure 7, the accuracy and loss curves during training demonstrate a stable and balanced performance between the training and validation datasets. The accuracy increased steadily up to approximately the 40th epoch before stabilizing with minimal fluctuations, indicating effective learning without significant overfitting. The loss curves also show a consistent decline for both datasets, reaching stability around the 50th epoch, with training loss slightly lower than validation loss. Although small fluctuations appeared in the validation loss, the overall trend remained steady and downward.

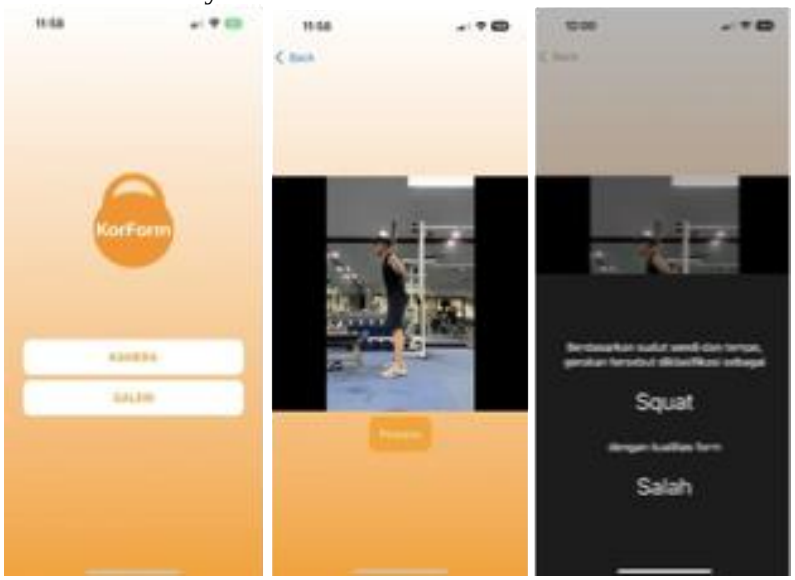

Figure 8. Application KorForm

These results confirm that the YOLO11n-pose and LSTM-based system was successfully trained and exhibits strong generalization performance on unseen data. The final trained model was integrated into the mobile application for real-time implementation, as shown in Figure 8 enabling users to receive instant feedback and classification results on their exercise performance.

CONCLUSIONS AND RECOMMENDATIONS

Based on the implementation and evaluation of the proposed YOLO11n-pose and LSTM-based system for classifying and assessing exercise form in deadlift and squat movements, several conclusions can be drawn. The classification model using the LSTM algorithm achieved a mean average precision (mAP) of 88.8% in distinguishing between correct and incorrect forms of weightlifting movements. Furthermore, the model demonstrated an overall accuracy of 94.81% on the test dataset derived from the train–validation–test split, indicating strong reliability in differentiating between deadlift and squat exercises.

The experimental results also showed that combining YOLO11n-pose for keypoint detection with LSTM for temporal sequence learning provides an effective approach for motion recognition tasks. YOLO11n-pose efficiently extracted keypoints and calculated joint angles from side-view exercise recordings, while the LSTM model successfully analyzed temporal changes in those angles to classify movement forms. The integration of both components enabled the system to evaluate exercise form automatically and accurately.

In addition, the model was successfully deployed in an iOS-based application, developed using SwiftUI and connected through FastAPI, allowing real-time processing of user-generated workout videos. This integration demonstrates the system's capability to provide immediate feedback on exercise performance directly on mobile devices, making it a practical tool for fitness training and injury prevention.

For future work, several improvements are recommended. Expanding the dataset to include more participants, varied camera angles, and additional exercise types could further enhance the model's robustness and generalization. Incorporating 3D keypoint detection or multi-view pose estimation may also improve accuracy in complex movements. Moreover, optimizing inference speed and on-device deployment through model quantization or lightweight architectures (e.g., MobileNet or TensorRT) could make the system more efficient for real-time mobile applications.

Overall, this study demonstrates that integrating YOLO11n-pose and LSTM offers a promising and practical approach to exercise form evaluation, contributing to safer, data-driven fitness practices and paving the way for intelligent training assistance systems.

REFERENCES

[1] World Health Organization (WHO). (2022). Global Status Report on Physical Activity 2022. Geneva, Switzerland: WHO Health Promotion. Retrieved from https://www.who.int/teams/health-promotion/physical-activity/global-status-report-on-physical-activity-2022

[2] Bukhary, H. A., Basha, N. A., Dobel, A. A., Alsufyani, R. M., Alotaibi, R. A., & Almadani, S. H. (2023). Prevalence and pattern of injuries across the weight-training sports. *Cureus*. https://doi.org/10.7759/cureus.49759

[3] Noteboom, L., Kemler, E., van Beijsterveldt, A. M. C., Hoozemans, M. J. M., van der Helm, F. C. T., & Verhagen, E. A. L. M. (2023). Factors associated with gym-based fitness injuries: A case-control study. *JSAMS Plus*, 2, 100032.

[4] Fu, H., Gao, J., & Liu, H. (2023). Human pose estimation and action recognition for fitness movements. *Computers & Graphics*, 116, 418–426. https://doi.org/10.1016/j.cag.2023.09.008

[5] Manivannan, S., Pradhan, Y., Muhammed, Z., Pooja, H., & Bharathi, R. (2024). Automated gym exercise form checker: Deep learning-based pose estimation. In *Smart Trends in Computing and Communications: Proceedings of SmartCom 2024, Volume 1* (pp. 71–84). Springer. https://doi.org/10.1007/978-981-97-1320-2

[6] Chen, J., Wang, J., Yuan, Q., & Yang, Z. (2023). CNN-LSTM model for recognizing video-recorded actions performed in a traditional Chinese exercise. *IEEE Journal of Translational Engineering in Health and Medicine*, 11, 351–359. https://doi.org/10.1109/JTEHM.2023.3282245

[7] Huang, C., Gochoo, M., & Tan, T. (2023). Two-stream architecture using RGB-based ConvNet and pose-based LSTM for video action recognition. In *Proceedings of the 2023 15th International Conference on Innovations in Information Technology (IIT)* (pp. 127–131). https://doi.org/10.1109/IIT59782.2023.10366415

[8] Kong, Y., *et al.* (2024). Unlocking the power of LSTM for long-term time series forecasting. *Proceedings of the AAAI Conference on Artificial Intelligence*, 39(11), 11968–11976. https://doi.org/10.1609/aaai.v39i11.33303

[9] Purnama, B., Erfianto, B., & Wirawan, I. R. (2024). Time series classification of badminton pose using LSTM with landmark tracking. *Jurnal Elektronika, Elektromedika, dan Informatika Medis*, 7(1), 27–37.

[10] Muriyah, N. M., Sim, J. H., & Yulianto, A. (2024). Evaluating YOLOv5 and YOLOv8: Advancements in human detection. *Journal of Information Systems and Informatics*, 6(4), 2999–3015. https://doi.org/10.51519/JOURNALISI.V6I4.944

[11] Effendi, Y., Kristian, Y., L. Z. P. C. S. W., & Yutanto, H. (2023). Pemanfaatan Mediapipe body pose estimation dan dynamic time warping untuk pembelajaran Tari Remo. *Jurnal Teknologi dan Manajemen Informatika*, 9(2), 183–190. https://doi.org/10.26905/JTMI.V9I2.10408

[12] Nainggolan, S. D. A., & Yuadi, I. (2025). Classification of wrist accessories: Advanced watches with logistic regression, SVM, and deep features from Inception V3 and VGG-19. *Jurnal Teknologi dan Manajemen Informatika*, 11(1), 25–34. https://doi.org/10.26905/JTMI.V11I1.15571
[13] Cinthiya, C., & Oetama, R. S. (2023). Enhancement of coronary heart disease prediction using stacked long short-term memory. *Jurnal Teknologi dan Manajemen Informatika*, 9(1), 28–36. https://doi.org/10.26905/JTMI.V9I1.9707
[14] Rijal, K. A., Vitianingsih, A. V., Kristyawan, Y., Maukar, A. L., & Wati, S. F. A. (2024). Forecasting model of Indonesia's oil & gas and non-oil & gas export value using VAR and LSTM methods. *Jurnal Teknologi dan Manajemen Informatika*, 10(1), 59–69. https://doi.org/10.26905/JTMI.V10I1.13127
[15] Hendrawan, N. D., & Kolandaisamy, R. (2023). A comparative study of YOLOv8 and YOLO-NAS performance in human detection image. *Jurnal Teknologi dan Manajemen Informatika*, 9(2), 191–201. https://doi.org/10.26905/JTMI.V9I2.12192