Jurnal Mandiri IT
Vol. 14 No. 1, July (2025), pp. 104-114

ISSN 2301-8984 (Print), 2809-1884 (Online)
Published by Institute of Computer Science

Machine learning-based approach for evaluating physical fitness through motion detection

Chadafa Zulti Noorta[1], M. Fazil Rais[2], M Ilham Alfatrah[3],Uvi Desi Fatmawati[4], H.A Danang Rimbawa[5]

[1,2,3,4]Electrical Engineering Study,The Republic of Indonesia Defense University, Bogor, Indonesia
[5]Cyber Defense Engineering Study Program,The Republic of Indonesia Defense University, Bogor, Indonesia

ARTICLE INFO

Article history:

Received May 15, 2025
Revised Jun 22, 2025
Accepted Jun 26, 2025

Keywords:

Exercise classification;
Machine learning;
Motion detection;
Physical fitness;
Pose estimation.
.

ABSTRACT

Physical fitness assessment is crucial for evaluating an individual's physical performance and endurance. However, traditional methods often rely on manual observation, which can lead to subjectivity and inconsistent results. This study proposes a machine learning-based approach for physical fitness evaluation through motion detection using pose estimation and exercise classification models. A quantitative method was employed to train and evaluate models for four exercise types: push-ups, sit-ups, pull-ups, and chinning. Each model was trained separately and assessed using accuracy, precision, recall, and F1-score metrics, achieving accuracies of 97.50% for push-ups, 97.67% for sit-ups, 97.00% for pull-ups, and 98.50% for chinning. The maximum error margin compared to manual counting was 2.48%. System-generated outputs were validated against manual observations using standard evaluation matrices. These findings indicate that machine learning can offer a reliable, consistent, and automated solution for physical fitness assessment, with the potential to enhance training programs, support remote fitness monitoring, and reduce human error in performance evaluation.

Corresponding Author:

Chadafa Zulti Noorta,
Electrical Engineering Study Program,
The Republic of Indonesia Defense University,
IPSC Sentul area, Sukahati, Citeureup sub-district, Bogor, West Java, 16810, Indonesia.
Email: chadafa7@gmail.com

1. INTRODUCTION

Physical fitness is a critical indicator of an individual's overall health and functionality, particularly in settings that require high levels of physical performance, such as the military, healthcare, and education. Common physical fitness tests—such as push-ups, sit-ups, pull-ups, and chinning—are widely used to assess muscular endurance, strength, and flexibility. However, these tests are still predominantly evaluated manually, which makes them prone to human error, subjective interpretation, and inconsistency in scoring. These challenges underline the need for an objective, automated, and scalable assessment method.

Recent advancements in artificial intelligence have made it possible to analyze human motion more accurately through computer vision and pose estimation. MediaPipe, an open-source framework developed by Google, is capable of detecting 33 key points of the human body in real time and has been proven effective in analyzing exercise movements with high precision (Kim et al., 2023; Zhang, 2022). While MediaPipe excels at capturing body posture, it requires integration with classification algorithms to identify specific exercise types in a meaningful context.

Convolutional Neural Networks (CNN), on the other hand, have shown outstanding performance in image and motion classification tasks due to their ability to learn spatial features effectively (Hadi et al., 2024; Tian, 2020). Several studies have explored combining pose estimation with CNNs for activity recognition. However, many of these systems are limited in terms of real-time capability, generalizability across diverse exercise types, or reliance on complex hardware. Furthermore, most prior research lacks a direct comparison between automated results and manual measurements, which is crucial for validating system reliability.

This study addresses these gaps by developing a machine learning-based physical fitness assessment system that combines MediaPipe for pose estimation with CNN for classification. The system is designed to detect and evaluate four types of exercises: push-up, sit-up, pull-up, and chinning. Each exercise is classified in real time, and the system's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. The results are then compared with manual observations to ensure reliability.

The scientific contributions of this research are threefold it presents an efficient, low-cost system that requires no specialized hardware; it demonstrates high classification accuracy across multiple types of physical exercises; and it offers practical applications in diverse environments such as education (for student fitness evaluations), military (for recruitment or training assessment), and healthcare (for patient rehabilitation monitoring). By addressing the limitations of traditional methods and existing automated systems, this research provides a scalable and objective approach to physical fitness evaluation based on motion analysis.

2. RESEARCH METHOD

This study utilized a quantitative experimental approach to develop a machine learning-based system for automatic evaluation of physical fitness tests, specifically focused on components: push-up, sit-up, pull-up, and chinning. The methodology involved several stages including data acquisition, model training, and performance evaluation using pose estimation and classification techniques.

a) Dataset Collection and Preprocessing

Video data were collected from individuals performing the selected physical movements. These videos were then segmented into image frames, which served as the primary dataset. Each frame was manually labeled according to the type of movement performed. MediaPipe, an open-source framework developed by Google, was utilized for pose estimation. This framework detects 33 key body landmarks in real-time and has been proven effective for identifying and analyzing human movement patterns (Kim et al., 2023; Zhang, 2022).

To illustrate the data acquisition process, Fig. 1 presents the primary dataset collection procedure, which involves capturing images of individuals performing the selected physical exercises.

Figure 1. the primary data acquisition

The dataset was collected from a total of 20 participants, consisting of both male and female individuals within the age range of 19 to 22 years. These participants were selected to represent a sample of physically active young adults. Each participant was instructed to perform four types of physical exercises: push-ups, sit-ups, pull-ups, and chinning, with all actions recorded for dataset construction.

Video recordings were captured using a Logitech HD Webcam with a resolution of 720p, ensuring clear visibility of body movements and joint positions. The system was operated on a Lenovo ThinkPad E14 Gen 2 laptop, featuring an Intel Core i5-1135G7 processor, 8 GB RAM, and integrated Intel Iris Xe graphics, providing sufficient computational power for real-time pose estimation and dataset labeling.

The data collection was conducted in an outdoor setting under controlled environmental conditions. Recordings were scheduled during daylight hours with adequate natural lighting to ensure body landmarks were clearly visible without significant shadows or overexposure. The recording environment was carefully selected to minimize background distractions and ensure a flat, even surface for consistent movement execution. The camera was placed at an optimal angle and height to capture the participants' full body posture during each exercise.

Each exercise in the dataset was categorized into two movement phases: up and down positions. This classification was essential for training the system to recognize the full range of motion and accurately count repetitions. For instance, in the push-up exercise, the *down* position corresponds to the body lowered toward the ground, while the *up* position indicates full arm extension. Similar labeling was applied to sit-up, pull-up, and chinning movements. This binary labeling approach helped the model learn the temporal transition between positions, which is critical for precise motion detection and repetition counting.

To enhance model performance and increase dataset variability, data augmentation techniques were applied to the collected image samples. These techniques included rotation, scaling, flipping, and brightness adjustments to simulate different conditions and perspectives during movement execution. Augmentation helps prevent overfitting by exposing the model to a wider range of input variations (Zhang, 2022; Kim et al., 2023). Fig. 2 shows examples of the augmented dataset used in this study, representing different visual appearances derived from the original images.

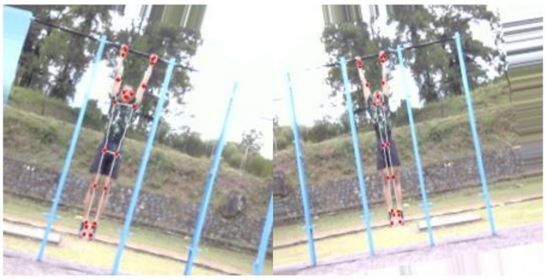

Figure 2. examples dataset

b) Dataset Preparation

The dataset used in this study consisted of a total of 8,000 labeled images, which were collected and categorized based on four types of physical exercises: push-up, sit-up, pull-up, and chinning. Each exercise type was assigned its own model and comprised 2,000 images, divided into 1,000 images of the *up* position and 1,000 images of the *down* position. This binary labeling approach was crucial for enabling the model to distinguish between the two key motion phases, thus allowing for accurate detection and repetition counting.

To ensure effective model training and evaluation, the dataset was split into three subsets using an 80:10:10 ratio: 80% (6,400 images) were allocated for training to allow the CNN to learn and generalize movement patterns. 10% (800 images) were used for validation to tune hyperparameters and monitor for overfitting during training. 10% (800 images) were reserved for testing to objectively assess the model's performance on unseen data (McAllister et al., 2018; Durden et al., 2021).

c) Model Development

Four separate Convolutional Neural Network (CNN) models were developed—each specifically trained on one type of exercise: push-up, sit-up, pull-up, and chinning.

```python
def create_pushup_model():
    # Input untuk gambar
    image_input = Input(shape=(224, 224, 3), name='image_input')
    x = Conv2D(32, (3, 3), activation='relu')(image_input)
    x = MaxPooling2D((2, 2))(x)
    x = Conv2D(64, (3, 3), activation='relu')(x)
    x = MaxPooling2D((2, 2))(x)
    x = Conv2D(128, (3, 3), activation='relu')(x)
    x = MaxPooling2D((2, 2))(x)
    x = Flatten()(x)

    # Input untuk landmark
    landmark_input = Input(shape=(landmarks_train.shape[1],), name='landmark_input')
    y = Dense(128, activation='relu', kernel_regularizer=l2(0.01))(landmark_input)

    # Gabungkan kedua input
    merged = Concatenate()([x, y])

    # Fully connected layer
    z = Dense(128, activation='relu', kernel_regularizer=l2(0.01))(merged)
    z = Dropout(0.5)(z)  # Dropout untuk mengurangi overfitting
    output = Dense(2, activation='softmax')(z)  # 2 kelas: down dan up
```

Figure 3. CNN architecture

Each model was trained independently to specialize in recognizing the motion characteristics of its respective class.

The CNN architecture for each model consisted of the following components:

1. Convolutional Layers: These layers extracted low-level spatial features such as edges, shapes, and contours from input images (Tian, 2020).
2. Activation Layers (ReLU): The Rectified Linear Unit (ReLU) function introduced non-linearity, which is essential for learning complex data distributions while accelerating convergence (Vakalopoulou et al., 2023).
3. Pooling Layers: Max pooling was applied to reduce the dimensionality of feature maps, helping the model focus on the most salient features while improving computational efficiency (Zhang, 2022).
4. Fully Connected Layers: These layers flattened the high-dimensional feature maps and performed final classification using a softmax function to determine whether the input image represented an *up* or *down* movement (Huang & Wang, 2022).

To analyze the impact of training duration on model performance, each model was trained using three different epoch settings: 20, 35, and 50 epochs. The optimal number of epochs was later determined based on validation performance.

d) Evaluation Matrix

To assess model performance in classifying physical movement phases, four standard classification metrics were used (Xu et al., 2020):

1. Accuracy: Indicates the overall percentage of correctly predicted images (both up and down phases).
2. Precision: Assesses the proportion of predicted positive labels (e.g., up) that were accurate.
3. Recall: Measures the proportion of actual positive instances correctly identified by the model.
4. F1-score: Computes the harmonic mean of precision and recall, providing a balanced evaluation, particularly in situations with imbalanced classes.

3. RESULTS AND DISCUSSIONS

To evaluate how well each CNN model learned from the dataset, training was conducted with 20, 35, and 50 epochs for four types of physical movements: push-up, sit-up, pull-up, and chinning. The focus was placed on observing the training accuracy achieved at each epoch to determine the model's learning effectiveness and convergence behavior.

a) Push-up Model

The push-up model demonstrated strong learning performance throughout the training process. At epoch 35, it achieved the highest training accuracy of 99.98%, indicating that the model had almost perfectly learned the distinguishing features between *up* and *down* positions. While the training accuracy slightly decreased to 99.14% at epoch 50, the model remained highly reliable in recognizing the movement.

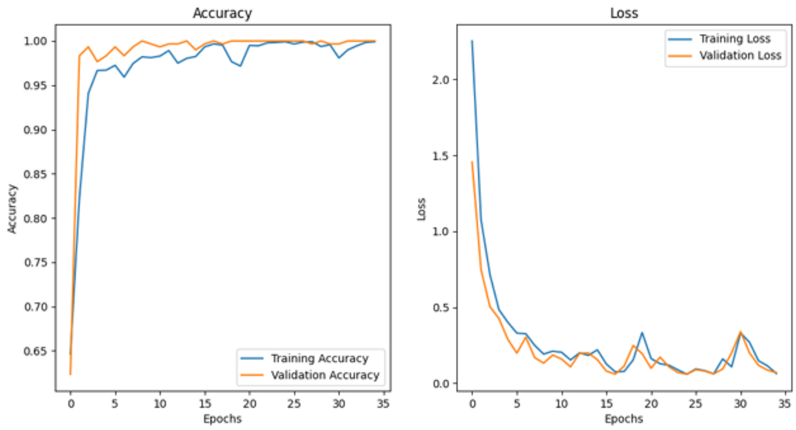

Figure. 4. Push-up Model Training Accuracy Curve

b) Sit-up Model

The sit-up model showed steady improvement over time, reaching its best performance at epoch 50, where it achieved a training accuracy of 99.75%. This result reflects a highly successful learning process, with the model being able to consistently distinguish between movement phases across the training data.

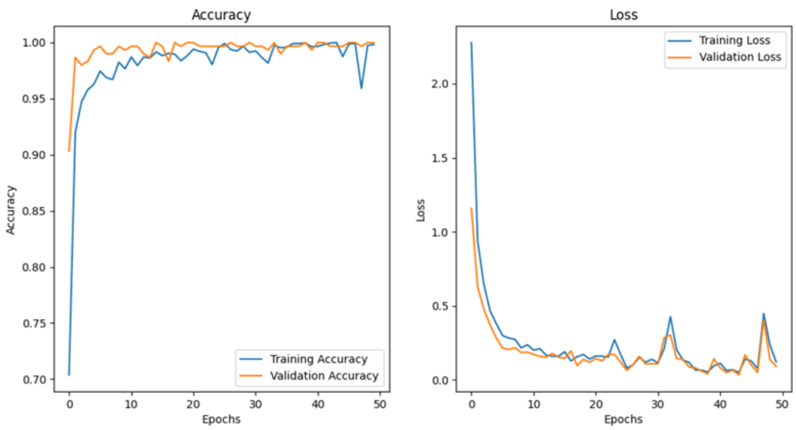

Figure. 5. Sit-up Model Training Accuracy Curve

c) Pull-up Model

Among the four models, the pull-up model achieved slightly lower accuracy but still maintained strong performance. Its highest training accuracy was 98.50% at epoch 50, suggesting that although the movement was slightly more complex to learn, the model was still able to generalize well from the training set.

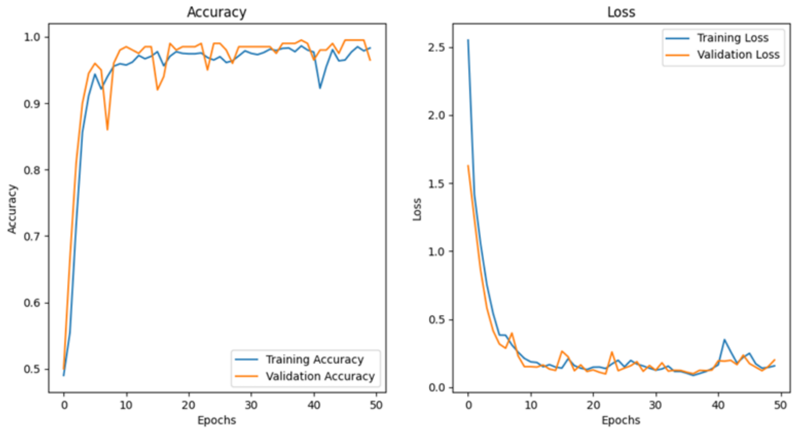

Figure. 6. Pull-up Model Training Accuracy Curve

d) Chinning Model

The chinning model demonstrated steady and reliable performance throughout all training periods. By epoch 50, it achieved a training accuracy of 99.01%, highlighting the model's effectiveness in extracting important features and accurately learning the motion phases. Additionally, the model exhibited smooth convergence, reflecting a well-balanced training process.

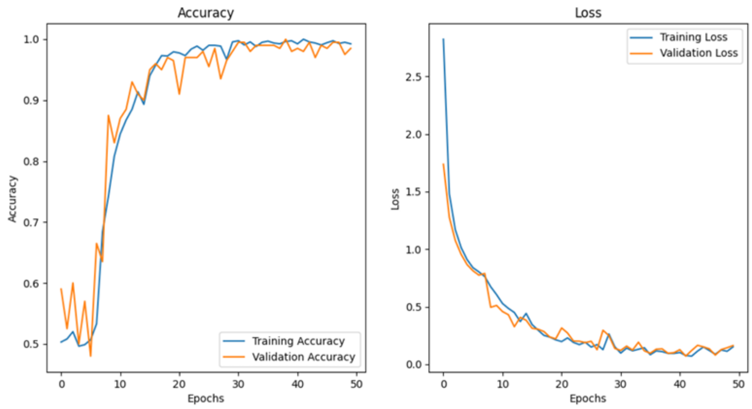

Figure. 7. Chinning Model Training Accuracy Curve

After the CNN models were trained, a series of manual testing sessions were conducted to evaluate the system's reliability and to compare it with human observation. Each model was tested using manually labeled video data, and classification accuracy was computed using performance Metrics derived from the confusion matrix include true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).

Ten test trials were performed for each exercise type, consisting of 60 repetitions per trial (30 for *up* and 30 for *down* positions) for push-up,sit-up and chinning, than 20 repetitions per trial (10 for *up* and 10 for *down* positions) for pull-up.

a) Push-up

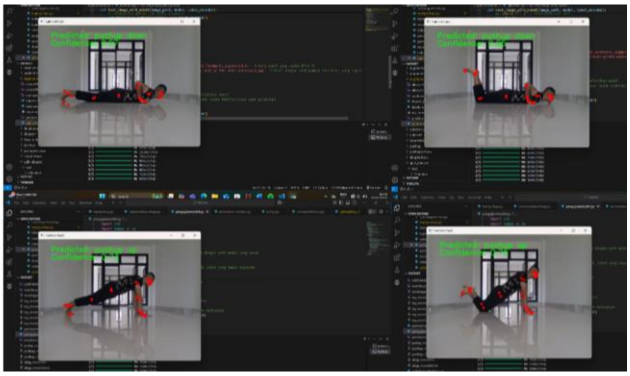

Figure 8. Push-up Test

Table 1. Manual Matrix Evaluation for Push-up

No	TP	FP	TN	FN	*Precision*	*Recall*	*F-1 score*	accuracy
1.	29	1	30	0	96,67%	100%	98,31%	98,33%
2.	30	0	29	1	100%	96,77%	98,37%	98,33%
3.	28	2	30	0	93,33%	100%	96.67%	96,67%
4.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
5.	30	0	30	0	100%	100%	100%	100%
6.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
7.	28	2	29	1	93,33%	96,55%	94,94%	95,00%
8.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
9.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
10.	30	0	30	0	100%	100%	100%	100%
		Average			97%	98%	97,49%	97,50%

b) Sit-up

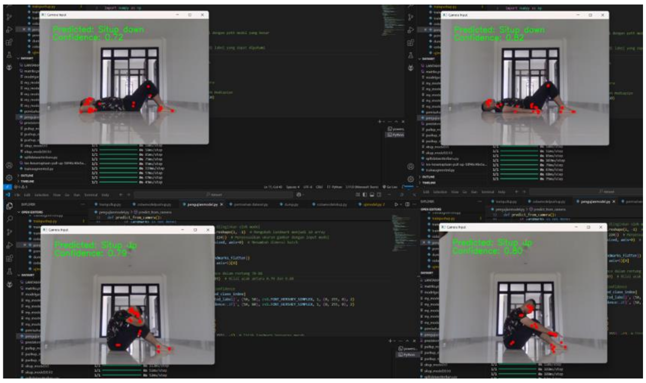

Figure 9. Sit-up Test

Table 2. Manual Matrix Evaluation for Sit-up

No	**TP**	**FP**	**TN**	**FN**	***Precision***	***Recall***	***F-1 score***	**accuracy**
1.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
2.	30	0	29	1	100%	96,67%	98,37%	98,33%
3.	28	2	30	0	93,33%	100%	96,67%	96,67%
4.	30	0	30	0	100%	100%	100%	100%
5.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
6.	28	2	30	0	93,33%	100%	96,67%	96,67%
7.	29	1	28	2	96,67%	93,55%	95,04%	95,00%
8.	30	0	30	0	100%	100%	100%	100%
9.	27	3	30	0	90,00%	100%	94,74%	95,00%
10.	30	0	30	0	100%	100%	100%	100%
		Average			97,41%	97,70%	97,57%	97,67%

c) Pull-up

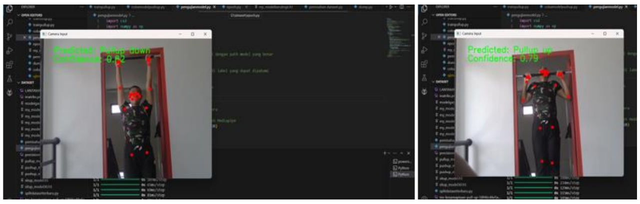

Figure 10. Pull-up Test

Table 3. Manual Matrix Evaluation for Pull-up

No	TP	FP	TN	FN	*Precision*	*Recall*	*F-1 score*	accuracy
1.	10	0	10	0	100%	100%	100%	100%
2.	9	1	9	1	90,00%	90,00%	90,00%	90,00%
3.	10	0	10	0	100%	100%	100%	100%
4.	9	1	9	1	90,00%	90,00%	90,00%	90,00%
5.	10	0	10	0	100%	100%	100%	100%
6.	10	0	10	0	100%	100%	100%	100%
7.	9	1	9	1	90,00%	90,00%	90,00%	90,00%
8.	10	0	10	0	100%	100%	100%	100%
9.	10	0	10	0	100%	100%	100%	100%
10.	10	0	10	0	100%	100%	100%	100%
		Average			97,00%	97,00%	97,00%	97,00%

d) Chinning

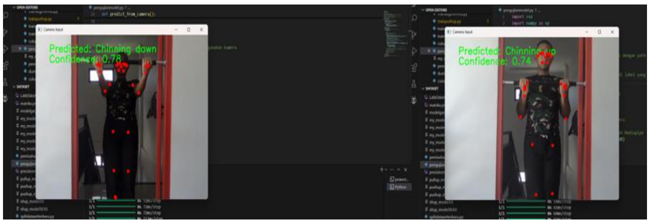

Figure 11. Chinning Test

Table 4. Manual Matrix Evaluation for Chinning

No	TP	FP	TN	FN	*Precision*	*Recall*	*F-1 score*	accuracy
1.	29	1	30	0	96,67%	100%	98,33%	98,33%
2.	30	0	30	0	100%	100%	100%	100%
3.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
4.	30	0	30	0	100%	100%	100%	100%
5.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
6.	28	2	30	0	93,33%	100%	96,67%	96,67%
7.	29	1	30	0	96,67%	100%	98,33%	98,33%
8.	30	0	29	1	100%	96,77%	98,36%	98,33%
9.	29	1	29	1	96,67%	96,67%	96,67%	96,67%
10.	30	0	30	0	100%	100%	100%	100%
		Average			98,00%	99,68%	98,50%	98,50%

These results demonstrate that all models consistently achieved high accuracy when tested manually. The push-up model showed a slight deviation of 2.48% from its training accuracy, the largest among the four. On the other hand, the chinning model showed the smallest deviation of just 0.51%, confirming its robustness under real-world conditions.

The minor differences between computational and manual accuracy could be attributed to several factors:

1. Environmental variations, such as lighting conditions and camera angle inconsistencies during testing,
2. Pose misalignment, where a participant's body position deviates slightly from the trained pose structure,

3. Limited diversity in the training dataset, which may affect the model's generalizability when applied to users with different body types, movement patterns, or speed variations.

Despite these challenges, the deviation margins remained within an acceptable range (below 3%), supporting the system's validity for real-time physical movement assessment. These findings highlight the system's potential as a dependable and objective alternative to traditional manual counting methods.

From a practical standpoint, this system holds significant applicability in various fields. In professional sports, it can be used to track athletes' performance metrics during training sessions, reducing reliance on subjective human observation. In physical education, especially in schools or universities, it enables automated monitoring and feedback without the need for instructors to manually count repetitions. In the medical rehabilitation sector, this technology can support physical therapists by monitoring patients' movement consistency and recovery progress in a non-invasive and real-time manner.

In comparison with previous studies, the model's overall accuracy is consistent with findings from other research integrating pose estimation and machine learning for activity classification. However, this study distinguishes itself by applying the model across multiple exercise types and validating it directly with manual accuracy comparison, a step that is often lacking in similar works. For instance, while other studies using CNNs have achieved accuracy rates above 95%, many rely on controlled laboratory datasets and do not account for variations in outdoor conditions or human execution errors.

The chinning model's superior performance may be explained by the distinct vertical movement pattern and limited range of body postures, which makes it easier for the algorithm to differentiate between phases. In contrast, exercises such as push-ups or sit-ups involve more nuanced body alignment and are more susceptible to subtle variations in posture, contributing to slightly higher deviation margins. From a theoretical perspective, models perform better when the movement patterns are highly consistent and exhibit clear pose transitions—criteria more easily met by exercises like chinning and pull-ups.

Overall, the findings reinforce the feasibility of combining pose estimation with deep learning for physical fitness evaluation, offering a practical, scalable, and intelligent alternative that can be tailored to diverse real-world applications.

4. CONCLUSION

This study developed a machine learning-based system for physical fitness evaluation using CNN and MediaPipe pose estimation, focusing on detecting up and down phases in four exercises: push-up, sit-up, pull-up, and chinning. Using 2,000 labeled images per exercise, the system achieved peak training accuracies of 99.98% (push-ups) and 99.01% (chinning). Manual testing showed consistent accuracy between 97.00%–98.50%, with error margins under 3%, confirming robustness in real-world conditions. Combining image data and pose landmarks improved classification by capturing movement variations across exercises. the study had limitations, including a small, homogenous sample of 20 participants aged 19–22, which may limit generalizability. Outdoor data collection also introduced environmental variability, such as lighting and background distractions, which could affect performance in uncontrolled settings. Scientifically, this research demonstrates the effectiveness of integrating pose estimation and CNN for accurate and interpretable fitness movement classification. Practically, it offers an automated alternative to manual methods in contexts like athletic training, physical education, and app-based fitness programs.Future work will focus on integrating real-time tracking (e.g., LSTM), expanding exercise types, and testing on broader populations to improve the system's scalability, adaptability, and real-world performance.

ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to all parties who have provided valuable support during the research and development of this system. Special thanks are extended to the Republic Of Indonesia Defense University for their participation in the data collection process and for sharing essential insights related to gesture interpretation and clarity.

The authors also thank their colleagues, peers, and academic supervisors whose encouragement, critical feedback, and collaboration contributed meaningfully to the successful completion of this study.

REFERENCES

Abdul Muthalib, M., Irfan, I., Kartika, K., & Selamat Meliala, S. M. (2023). Pengiraan pose model manusia pada repetisi kebugaran AI pemograman Python berbasis komputerisasi. *Infotech Journal, 9*(1), 11–19. https://doi.org/10.31949/infotech.v9i1.4233

Arlin, R., & Munir, R. (n.d.). *The development of push up counter Android application with computer vision*.

Asrese, A. S., Eravuchira, S. J., Bajpai, V., Sarolahti, P., & Ott, J. (2019). Measuring web latency and rendering performance: Method, tools, and longitudinal dataset. *IEEE Transactions on Network and Service Management, 16*(2), 535–549. https://doi.org/10.1109/TNSM.2019.2896710

Chen, L., Chen, P., & Lin, Z. (2020). Artificial intelligence in education: A review. *IEEE Access, 8*, 75264–75278. https://doi.org/10.1109/ACCESS.2020.2988510

Durden, J. M., Hosking, B., Bett, B. J., Cline, D., & Ruhl, H. A. (2021). Automated classification of fauna in seabed photographs: The impact of training and validation dataset size, with considerations for the class imbalance. *Progress in Oceanography, 196*, 102612. https://doi.org/10.1016/j.pocean.2021.102612

Ganaie, M. A., Hu, M., Malik, A. K., Tanveer, M., & Suganthan, P. N. (2022). Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence, 115*, 105151. https://doi.org/10.1016/j.engappai.2022.105151

Guo, M. H., Xu, T. X., Liu, J. J., Liu, Z. N., Jiang, P. T., Mu, T. J., Zhang, S. H., Martin, R. R., Cheng, M. M., & Hu, S. M. (2022). Attention mechanisms in computer vision: A survey. *Computational Visual Media, 8*(3), 331–368. https://doi.org/10.1007/s41095-022-0271-y

Huang, Y., & Wang, W. (2022). Cultural creativity, industrial scale, management methods, and their roles in rural revitalization from the perspective of big data. *Mathematical Problems in Engineering, 2022*, 1–9. https://doi.org/10.1155/2022/6792716

Kim, J. W., Choi, J. Y., Ha, E. J., & Choi, J. H. (2023). Human pose estimation using MediaPipe pose and optimization method based on a humanoid model. *Applied Sciences, 13*(4), 2700. https://doi.org/10.3390/app13042700

Kumar, R., Singh, S. K., Bajpai, A., & Sinha, A. (n.d.). *MediaPipe and CNNs for real-time ASL gesture recognition*.

Kusno Hadi, I., Manajemen Pertahanan Akademi Militer, P., Basuki, A., Raindra, M., & Ferdiyansyah, M. I. (2024). Pengelolaan manajemen pembinaan Garjas Pasipamops dalam meningkatkan nilai tes kesegaran jasmani sersan taruna di Batalyon Taruna Dewasa. *Jurnal Pertahanan, 11*(2), 2355–5262.

Marpaung, N. L., & Al Amzah, R. (2022). Rancang bangun program aplikasi tes kesegaran jasmani Indonesia berbasis Android. *Jurnal Teknik Informatika dan Sistem Informasi, 9*(2). http://jurnal.mdp.ac.id

McAllister, P., Zheng, H., Bond, R., & Moorhead, A. (2018). Combining deep residual neural network features with supervised machine learning algorithms to classify diverse food image datasets. *Computers in Biology and Medicine, 95*, 217–233. https://doi.org/10.1016/j.compbiomed.2018.02.008

Miftahul, A. F., Fatkhu R, I., & Maliki, O. (2020). Upaya peningkatan hasil belajar psikomotorik lompat jauh gaya jongkok melalui drill dan MediaPipe jump hanging ball pada siswa kelas VII SMPN 1 Kandeman. *JSSF, 6*(1). http://journal.unnes.ac.id/sju/index.php/jssf

Mugiat, D. (n.d.). *Pengaruh metode circuit training berbantuan alat peraga, tanpa berbantuan alat peraga, dan motivasi terhadap hasil tes kesegaran jasmani siswa TNI AL*.

Nurudin, M., Jayanti, W., Saputro, R. D., Saputra, M. P., & Yulianti, D. (2019). Pengujian black box pada aplikasi penjualan berbasis web menggunakan teknik boundary value analysis. *Jurnal Informatika, 4*(4), 2622–4615. http://openjournal.unpam.ac.id/index.php/informatika

Sánchez-Vicinaiz, T. J., Camacho-Pérez, E., Castillo-Atoche, A. A., Cruz-Fernandez, M., García-Martínez, J. R., & Rodríguez-Reséndiz, J. (2024). MediaPipe frame and convolutional neural networks-based fingerspelling detection in Mexican sign language. *Technologies, 12*(8). https://doi.org/10.3390/technologies12080124

Satrya Perbawa, D., & Nurohim, G. S. (2020). Pengujian aplikasi berbasis website dengan black box testing metode boundary value analysis dan responsive testing. *Journal SPEED, 12*(4).

Shin, J., Matsuoka, A., Hasan, M. A. M., & Srizon, A. Y. (2021). American Sign Language alphabet recognition by extracting features from hand pose estimation. *Sensors, 21*(17), 5810. https://doi.org/10.3390/s21175810

Tian, Y. (2020). Artificial intelligence image recognition method based on convolutional neural network algorithm. *IEEE Access, 8*, 125731–125744. https://doi.org/10.1109/ACCESS.2020.3006097

Verma, A. R., Singh, G., Meghwal, K., Ramji, B., & Dadheech, P. K. (2024). Enhancing sign language detection through MediaPipe and convolutional neural networks (CNN). *arXiv preprint*. http://arxiv.org/abs/2406.03729

Wu, J. (2017). *Introduction to convolutional neural networks*. https://cs.nju.edu.cn/wujx/paper/CNN.pdf

Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C. L., & Grundmann, M. (2020). MediaPipe hands: On-device real-time hand tracking. *arXiv preprint* arXiv:2006.10214. https://arxiv.org/abs/2006.10214

Zhang, Y. (2022). *Applications of Google MediaPipe pose estimation using a single camera*.

Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems, 30*(11), 3212–3232. https://doi.org/10.1109/TNNLS.2018.2876865