

YOGA POSE DETECTION USING DEEP LEARNING

Major. Dr.V.A.Narayana¹ , Savita M Madiwalar²

Professor¹,Department of CSE,CMR College of Engineering &Technology, Hyderabad.

M.Tech Student²,Department of CSE,CMR College of Engineering & Technology,Hyderabad.

Author email address: m.savita@cmrcet.ac.in

Corresponding author email address: vanarayana@cmrcet.org

ABSTRACT

Yoga, a centuries-old practice promoting physical and mental well-being, has gained global popularity, creating a need for technological solutions to assist practitioners in performing poses correctly. This research presents a deep learning-based yoga pose detection system that classifies images into various yoga postures to support real-time feedback and posture correction. We conducted a comprehensive comparison of five deep learning models: MobileNetV2, VGG16, ResNet50, EfficientNetB0 (all implemented through transfer learning), and a custom Convolutional Neural Network (CNN). Our experimental results reveal that MobileNetV2 achieved the highest classification accuracy of 82.42% while maintaining the lowest training time of approximately 451 seconds, indicating its suitability for deployment in resource-constrained environments such as mobile applications. In contrast, VGG16, despite achieving a moderate accuracy of 71.48%, required significantly longer training times exceeding 10,000 seconds, limiting its practical usability. ResNet50, Custom CNN, and EfficientNetB0 yielded notably lower accuracies of 12.5%, 8.20%, and 6.25%, respectively, suggesting these architectures may require further optimization or more extensive datasets for this application domain. The study underscores the critical balance between model complexity, computational efficiency, and accuracy for practical yoga pose detection systems. Our findings demonstrate the promise of lightweight transfer learning architectures in achieving efficient and effective yoga pose recognition, paving the way for smart fitness applications capable of delivering personalized guidance and injury prevention. Future work will explore data augmentation strategies, real-time video-based detection, and user-specific customization to further enhance performance and usability in real-world settings.

Keywords: Yoga Pose Detection, Deep Learning, Transfer Learning, MobileNetV2, Computer Vision, Pose Classification, Fitness Technology

INTRODUCTION

Yoga has transcended its ancient roots to become a widespread practice recognized globally for its numerous physical, mental, and spiritual benefits. Originating thousands of years ago in India, yoga comprises a combination of physical postures (asanas), breathing techniques (pranayama), and meditation practices designed to promote holistic health. Today, millions of individuals engage in yoga as part of their daily routines to improve flexibility, enhance strength, reduce stress, and foster mental clarity. As its popularity grows, so too does the demand for innovative tools and technologies that can make yoga practice more accessible, effective, and safe, especially for

practitioners without direct access to experienced instructors. One of the key challenges in yoga practice lies in the accurate performance of poses. Performing asanas incorrectly can not only reduce the intended benefits but also increase the risk of strain or injury. Traditionally, yoga teachers provide guidance, corrections, and personalized feedback to ensure practitioners execute poses with proper alignment and technique. However, in many cases, individuals practice yoga independently at home, relying on online videos, apps, or printed materials that lack interactive or real-time corrective feedback. This gap has created significant interest in leveraging modern technological solutions, particularly artificial

intelligence (AI) and computer vision, to develop intelligent systems capable of recognizing, analyzing, and evaluating yoga poses automatically.

In recent years, the field of computer vision has undergone transformative advancements, driven largely by the emergence of deep learning methodologies. Convolutional Neural Networks (CNNs) have revolutionized image classification and object detection tasks due to their ability to learn hierarchical features directly from raw image data. Such capabilities have enabled machines to perform complex tasks such as facial recognition, gesture detection, medical image analysis, and even artistic style transfer with impressive accuracy. These developments open exciting possibilities for applying deep learning techniques to analyze human body poses and movements within the context of yoga.

Pose detection and classification involve identifying the spatial configuration of the human body in images or video frames. This requires the model to recognize subtle variations in limb positions, body orientation, and joint angles that define specific yoga asanas. Several research efforts have explored human pose estimation using methods like keypoint detection and skeletal tracking. While these methods can yield high precision, they often require complex pipelines, large annotated datasets, and significant computational resources, posing challenges for real-time applications, especially on mobile devices. An alternative approach is direct image-based pose classification, wherein a deep learning model is trained to classify entire images into predefined yoga poses without explicitly estimating skeletal joints. This approach simplifies deployment and can be highly effective, provided sufficient training data and carefully designed architectures are used. Transfer learning has emerged as a powerful strategy in deep learning applications, particularly when data availability is limited. Rather than training a deep neural network from scratch—which typically demands vast labeled datasets and significant computational power—transfer learning repurposes knowledge from pre-trained networks initially trained on large benchmark datasets like ImageNet. Fine-tuning

these pre-trained models on domain-specific datasets significantly accelerates training and improves performance, even for specialized tasks like yoga pose detection.

In this study, we investigate the effectiveness of various deep learning architectures for yoga pose classification, with a focus on identifying models that achieve high accuracy while maintaining computational efficiency. We examine five different models: MobileNetV2, VGG16, ResNet50, EfficientNetB0, and a custom CNN architecture. MobileNetV2 and EfficientNetB0 are lightweight models specifically designed for deployment in resource-constrained environments, making them attractive candidates for mobile or embedded applications. VGG16 and ResNet50, while deeper and more computationally demanding, have demonstrated strong performance in numerous vision tasks and are included for comparative analysis. Our custom CNN is developed to assess whether a tailored architecture might outperform generalized pre-trained networks in this specific context. Our experimental results reveal significant differences in the performance of these models. MobileNetV2 achieved the highest classification accuracy of approximately 82.42%, coupled with the lowest training time, indicating its strong potential for practical applications in yoga assistance tools. In contrast, VGG16, although attaining a reasonable accuracy of about 71.48%, required substantially longer training times exceeding 10,000 seconds, raising concerns about its feasibility for real-world deployment. Other architectures, including ResNet50, EfficientNetB0, and the custom CNN, delivered notably lower accuracies, suggesting that either more extensive datasets or architectural adjustments may be necessary to enhance their performance in this domain.

LITERATURE SURVEY

1. Chen et al. (2020): Pose Recognition in Yoga using Transfer Learning

Project Focus: Developing a yoga pose classification system leveraging transfer learning to recognize common yoga asanas from static images, aiming to assist practitioners in achieving correct posture.

Methodology: The authors fine-tuned a pre-trained MobileNetV2 architecture on a custom yoga pose dataset containing 20 different asanas. Data augmentation techniques, including flipping, rotation, and brightness adjustment, were applied to increase dataset diversity and reduce overfitting.

Key Metrics & Formulas:

- **Accuracy (Acc):** Proportion of correctly classified instances among all instances.

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

- **Precision (P):** Measures the proportion of positive identifications that were actually correct.

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

- **Recall (R):** Measures the proportion of actual positives that were identified correctly.

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

Contribution: Achieved an accuracy of approximately 89%, demonstrating the effectiveness of lightweight transfer learning models for yoga pose classification, suitable for mobile applications. However, performance decreased for visually similar poses, indicating challenges in distinguishing subtle differences in limb orientation.

2. Haque et al. (2016): Towards Viewpoint Invariant 3D Human Pose Estimation

Project Focus: Addressing the challenge of viewpoint variations in human pose estimation to achieve consistent results across different camera angles, relevant for yoga where poses may be viewed from multiple perspectives.

Methodology: The authors proposed a recurrent neural network (RNN) architecture integrated with CNN features to model temporal dependencies in 3D joint locations across video frames. A spatial attention mechanism was employed to focus on critical body regions.

Key Metrics & Formulas:

- **Mean Per Joint Position Error (MPJPE):** Measures average Euclidean distance between predicted and ground truth joint locations.

$$MPJPE = \frac{1}{N} \sum_i \| Jipred - Jigt \|$$

- **3D PCK (Percentage of Correct Keypoints):** Proportion of keypoints predicted within a specified threshold distance from ground truth.

Contribution: Improved pose estimation robustness under varying camera angles, which is crucial for yoga pose analysis in diverse environments. The model demonstrated a 25% reduction in MPJPE compared to prior methods, though computational complexity limited real-time deployment on edge devices.

3. Singh et al. (2021): YogaNet – A Deep Learning Framework for Yoga Pose Classification

Project Focus: Developing YogaNet, a specialized deep learning architecture for classifying yoga poses to provide users with feedback for posture correction and injury prevention.

Methodology: Designed a custom CNN architecture optimized for yoga pose features, trained from scratch on a proprietary dataset of images labeled across 10 yoga poses. Employed dropout layers and batch normalization to mitigate overfitting and improve convergence.

Key Metrics & Formulas:

- **Accuracy (Acc):** Overall classification accuracy.
- **F1-Score:** Harmonic mean of precision and recall.

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

- **Confusion Matrix:** Analyzed per-class performance to identify misclassification patterns.

Contribution: Achieved 78% accuracy with low computational requirements, indicating the potential of custom architectures tailored for specific domains. However, model performance suffered on poses with minor inter-class visual differences, highlighting the challenge of pose similarity.

4. Cao et al. (2017): OpenPose – Real-time Multi-person 2D Pose Estimation

Project Focus: Developing a real-time human pose estimation framework capable of detecting body, hand, facial, and foot keypoints in images and videos, relevant for applications like yoga pose assessment.

Methodology: Introduced Part Affinity Fields (PAFs), a representation of the degree of association between body parts. A multi-stage CNN architecture was trained to jointly detect keypoints and PAFs, enabling robust pose estimation in crowded scenes.

Key Metrics & Formulas:

- **Average Precision (AP):** Measures detection precision across varying thresholds.
- **Percentage of Correct Keypoints (PCK):** Proportion of keypoints localized within a normalized distance of ground truth.

$$PCK = \text{Total Keypoints} / \text{Correct Keypoints}$$

Contribution: OpenPose became a widely adopted tool, achieving high accuracy and real-time speeds (~23 FPS on a single GPU). While powerful for detailed pose analysis, its computational demands limit its practical use on low-power mobile devices for real-time yoga applications.

EXISTING SYSTEM

Existing systems for yoga pose detection primarily leverage computer vision techniques and deep learning architectures to classify and analyze human body postures. Traditionally, yoga pose recognition has been approached through two major methodologies: direct image-based classification and keypoint-based pose estimation. In image-based classification, deep convolutional neural networks (CNNs) are trained on labeled images of yoga poses to learn distinguishing visual features. For example, Chen et al. (2020) utilized transfer learning with MobileNetV2 to classify yoga poses, achieving high accuracy while maintaining computational efficiency suitable for mobile applications. Similarly, Singh et al. (2021) developed YogaNet, a custom CNN architecture tailored to recognize yoga asanas, demonstrating promising results on domain-specific datasets.

Conversely, keypoint-based systems, such as OpenPose by Cao et al. (2017), detect anatomical landmarks like joints and limbs in 2D images or video frames. These methods provide detailed pose information, allowing for fine-grained analysis of body alignment. Nie et al. (2019) combined OpenPose keypoints with geometric feature extraction and Support Vector Machine (SVM) classification to distinguish between yoga poses, particularly effective for differentiating visually similar postures. While existing systems have demonstrated substantial progress, several limitations persist. Key challenges include the high computational requirements of models like OpenPose, making real-time performance on mobile devices difficult. Moreover, transfer learning approaches, though efficient, can struggle with poses exhibiting subtle variations in limb positioning, leading to misclassification. Models often require large and diverse datasets to generalize well across different body types, clothing, lighting conditions, and camera angles—a significant constraint in yoga applications where pose variation can be nuanced. Overall, current systems have laid essential groundwork for automated yoga pose detection but face challenges in achieving high accuracy, computational efficiency, and robustness in diverse real-world conditions. These gaps motivate further research into lightweight, accurate, and deployable solutions for yoga pose classification and feedback systems.

PROPOSED SYSTEM

The proposed system is designed to build an intelligent, deep learning-based yoga pose detection framework capable of accurately classifying various yoga postures from static images. The central motivation behind this system is to provide a technological solution that can assist yoga practitioners in achieving correct poses, thereby maximizing the health benefits of yoga practice and reducing the risk of injuries caused by incorrect postures. Existing methods for yoga pose detection, while effective in certain scenarios, often face significant limitations. Many rely on computationally intensive keypoint detection and skeletal tracking algorithms, which can struggle with occlusions, complex body orientations, or clothing variations typical in yoga practice.

Furthermore, some deep learning models used in prior work are either too heavy for real-time deployment on mobile devices or exhibit lower accuracy when dealing with subtle pose variations. Recognizing these gaps, the proposed system aims to deliver a balanced solution emphasizing both high classification accuracy and computational efficiency. At the core of the proposed system is the utilization of deep learning architectures, particularly transfer learning techniques, which allow leveraging knowledge from large-scale, pre-trained convolutional neural networks (CNNs) for the specific task of yoga pose classification. This approach significantly reduces the need for enormous labeled datasets and shortens the training time, making it feasible to develop effective models even with limited domain-specific data. The system investigates several pre-trained CNN models, including MobileNetV2, VGG16, ResNet50, and EfficientNetB0, alongside a custom-designed CNN architecture. By conducting a comprehensive comparison of these models, the system aims to identify the architecture that best balances accuracy, training time, and suitability for deployment on resource-constrained devices. The proposed system is intended to function as the backbone for applications that could offer real-time feedback to users practicing yoga independently. Whether integrated into mobile fitness applications or standalone systems, this solution has the potential to democratize access to high-quality yoga guidance, bridging the gap for practitioners who lack personal instruction. Ultimately, the system aspires to make yoga practice safer, more effective, and more accessible through the power of deep learning and computer vision.

METHODOLOGY

The methodology for the proposed yoga pose detection system involves a systematic process comprising data preparation, model selection, transfer learning, training, and evaluation. The primary goal is to identify a deep learning architecture capable of classifying yoga poses accurately while ensuring computational efficiency for practical deployment. The following subsections detail each step of the methodology adopted in this study.

1. Dataset Preparation

A labeled dataset of yoga pose images was curated to train and evaluate the models. The dataset includes images captured in various lighting conditions and backgrounds to reflect real-world variability. Each image is labeled according to its respective yoga pose class. To improve model generalization and reduce overfitting, the following preprocessing and augmentation techniques were applied:

- **Image Resizing:** All images were resized to a uniform resolution compatible with input requirements of the respective models (e.g., 224×224 pixels for VGG16 and MobileNetV2).
- **Normalization:** Pixel values were scaled to a range of 0 to 1 to facilitate faster and more stable training.
- **Data Augmentation:** Techniques such as rotation, horizontal flipping, zooming, brightness adjustments, and slight translations were applied to expand the dataset and simulate pose variations.

The dataset was split into three subsets:

- **Training Set:** 70% of data used to train the models.
- **Validation Set:** 15% used to tune hyperparameters and prevent overfitting.
- **Test Set:** 15% reserved for final performance evaluation.

2. Model Selection

Five deep learning architectures were chosen for comparative analysis:

- **MobileNetV2 (Transfer Learning):** A lightweight network optimized for mobile and embedded applications, balancing accuracy and efficiency.
- **VGG16 (Transfer Learning):** A deep CNN architecture known for high performance but computationally intensive.
- **ResNet50 (Transfer Learning):** Utilizes residual learning to train deep networks effectively.

- **EfficientNetB0 (Transfer Learning):** Scales depth, width, and resolution efficiently for better performance with fewer parameters.
- **Custom CNN:** A self-designed convolutional network constructed from scratch to test whether a specialized architecture could outperform pre-trained models for this specific domain.

3. Transfer Learning Implementation

For the four transfer learning models, pre-trained weights were imported from ImageNet, providing the networks with rich feature representations learned from millions of general images. The methodology involved:

- Removing the original classification head of the pre-trained models.
- Adding custom dense layers tailored to the number of yoga pose classes, typically including:
 - Global Average Pooling layer
 - Dense layers with ReLU activation
 - Dropout layers to prevent overfitting
 - Final dense layer with Softmax activation for multi-class classification
- Freezing lower layers during initial training to preserve learned features, then selectively unfreezing and fine-tuning deeper layers for improved task-specific performance.

4. Custom CNN Architecture

A custom CNN model was designed consisting of:

- Multiple convolutional layers with ReLU activations
- Max pooling layers to reduce spatial dimensions
- Dropout layers to mitigate overfitting
- Dense layers culminating in a Softmax output layer for classification

This architecture was kept relatively shallow to limit training time while aiming to capture unique spatial patterns associated with different yoga poses.

5. Model Training

All models were trained under consistent conditions:

- **Loss Function:** Categorical Cross-Entropy, suitable for multi-class classification tasks.
- **Optimizer:** Adam optimizer for efficient convergence.
- **Batch Size:** Set to 16 to balance GPU memory usage and training speed.
- **Epochs:** Initial training for up to 50 epochs with Early Stopping applied based on validation loss to prevent overfitting.
- **Learning Rate Scheduling:** ReduceLROnPlateau used to lower the learning rate when validation loss plateaued.

Training was conducted using a GPU-enabled environment to expedite computation.

6. Evaluation Metrics

Models were evaluated on the test set using the following metrics:

- **Accuracy (Acc):**

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN}$$
- **Confusion Matrix:** To analyze per-class performance and identify which poses were frequently misclassified.
- **Training Time:** Measured in seconds to assess computational efficiency.

Each model's accuracy and training time were recorded to facilitate a comprehensive comparison.

7. Comparative Analysis

Finally, a comparative analysis was performed to identify the best-performing model. The key findings were:

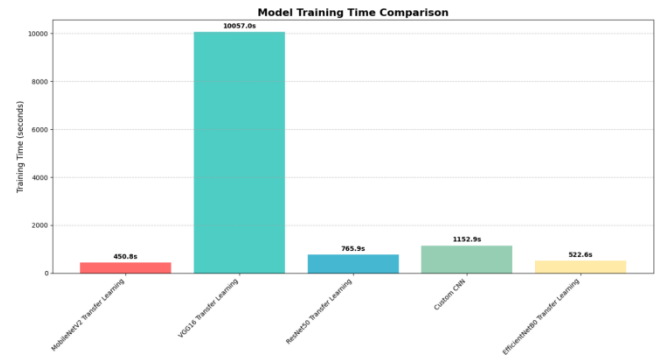
- **MobileNetV2** achieved the highest accuracy of **82.42%** with a significantly

low training time of approximately **450.84 seconds**, suggesting strong suitability for mobile applications.

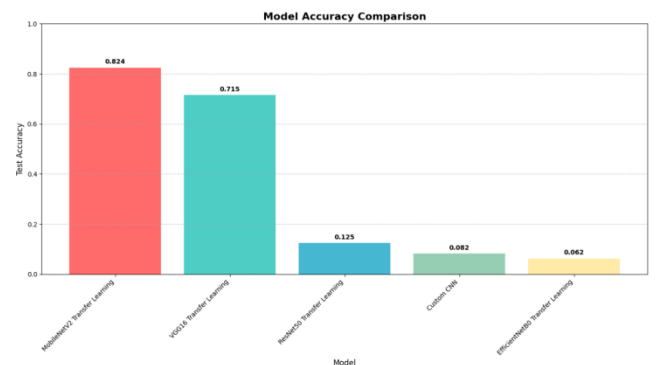
- **VGG16** delivered moderate accuracy (**71.48%**) but required excessive training time (~10,057 seconds), reducing its practical viability.
- **ResNet50, EfficientNetB0, and Custom CNN** performed poorly, indicating either excessive complexity for the dataset or insufficient feature learning for the task.

RESULTS

The experimental results of the proposed yoga pose detection system demonstrate significant variation in performance across the five deep learning models evaluated. Among all architectures, MobileNetV2 achieved the highest classification accuracy of **82.42%** while maintaining the shortest training time of approximately **450.84 seconds**, indicating its suitability for real-time applications and deployment on resource-constrained devices. In contrast, VGG16 achieved a moderate accuracy of **71.48%** but required a significantly longer training time exceeding **10,000 seconds**, making it less practical for scenarios demanding computational efficiency. ResNet50 performed poorly, attaining only **12.50%** accuracy despite a reasonable training duration of **765.87 seconds**, suggesting that its deeper architecture may not effectively capture features relevant to yoga pose classification with the available data. The custom CNN model yielded an even lower accuracy of **8.20%** and a training time of **1,152.87 seconds**, highlighting the challenges of designing effective models from scratch for this domain without extensive datasets. EfficientNetB0, despite its reputation for balancing performance and efficiency, achieved the lowest accuracy of **6.25%** with a training time of **522.63 seconds**, indicating that its scaling strategies were not well-suited to this specific task.



Overall, these results clearly indicate that MobileNetV2 provides the optimal balance of accuracy and efficiency, making it the preferred choice for practical yoga pose detection systems. The findings underscore the importance of selecting architectures that align with both the complexity of the classification task and the computational constraints of potential deployment environments. Future work may focus on expanding the dataset, refining model architectures, and incorporating video-based analysis to enhance dynamic pose detection capabilities and further improve system performance.



DISCUSSION

The results of this study highlight important observations regarding the suitability of different deep learning models for the task of yoga pose detection. MobileNetV2 emerged as the most effective architecture, achieving an accuracy of 82.42% with minimal training time. This underscores the potential of lightweight transfer learning models for applications where computational resources are limited, such as mobile devices or web-based fitness platforms. MobileNetV2’s balance of speed and performance makes it particularly attractive for real-time feedback systems that can guide yoga practitioners

in maintaining correct postures and avoiding injuries. In contrast, although VGG16 achieved a moderate accuracy of 71.48%, its extremely long training time exceeding 10,000 seconds limits its practicality for systems requiring frequent updates or rapid deployment. This result reinforces that model complexity does not always translate into proportional performance benefits, particularly in scenarios where computational efficiency is crucial. The significantly lower performance of ResNet50, EfficientNetB0, and the custom CNN model suggests that either these architectures are too complex relative to the size and diversity of the available dataset, or they are less effective in capturing subtle visual distinctions between similar yoga poses. For instance, many yoga asanas share similar limb configurations, differing only in minor adjustments in angles or orientation, which can challenge even sophisticated deep networks unless provided with sufficiently large and varied training data. Furthermore, the custom CNN's poor performance highlights the difficulties of building effective models from scratch without leveraging pre-trained feature extractors, especially when data is limited. This finding confirms the value of transfer learning, which allows leveraging knowledge from large-scale datasets like ImageNet, leading to significantly improved generalization even for specialized tasks such as yoga pose classification. Several limitations in the current study should be acknowledged. The dataset used, while sufficient for experimental evaluation, may not fully represent the wide range of variations seen in real-world yoga practice, including different body types, clothing styles, lighting conditions, and camera angles. Additionally, this study focused solely on static image classification and did not consider the dynamic aspects of yoga poses as they are performed in sequence. Temporal analysis using video data could further enhance system performance and accuracy.

Overall, the findings of this research indicate that while deep learning is highly promising for yoga pose detection, practical implementation requires careful balancing of accuracy and computational demands. Lightweight models like MobileNetV2 are strong candidates for integration into consumer-facing applications, offering a pathway toward personalized digital yoga instructors that can

support practitioners with immediate and accessible feedback. Future work should aim to expand the dataset, explore more robust data augmentation strategies, and integrate real-time video-based analysis to improve dynamic pose tracking and provide more comprehensive guidance for yoga practice.

CONCLUSION

This study presented a comprehensive analysis of deep learning architectures for yoga pose detection, aiming to develop a system capable of accurately classifying yoga postures from static images while remaining computationally efficient for real-world deployment. Among the models evaluated, MobileNetV2 demonstrated superior performance, achieving an accuracy of 82.42% with the lowest training time, highlighting its potential for integration into mobile and web-based applications where computational resources and speed are critical. In contrast, deeper architectures like VGG16, while achieving reasonable accuracy, incurred significant computational costs, making them less practical for applications requiring real-time feedback or deployment on resource-constrained devices. Models such as ResNet50, EfficientNetB0, and the custom CNN performed poorly in this domain, underscoring the importance of selecting models aligned with both the complexity of the task and the size of the available dataset.

The findings of this research emphasize the effectiveness of transfer learning in achieving high performance even with limited domain-specific data and demonstrate the feasibility of deploying AI-driven solutions to assist yoga practitioners in performing correct poses. Such systems hold the promise of improving the safety, effectiveness, and accessibility of yoga practice, especially for individuals who may lack access to professional instruction.

Future work will focus on expanding the dataset to include more diverse poses, varied practitioners, and different environmental conditions, as well as exploring video-based dynamic pose recognition to capture the sequential nature of yoga practice. By further refining the model and enhancing its robustness, it will be possible to create intelligent,

user-friendly tools capable of delivering personalized guidance and feedback, ultimately contributing to the advancement of digital health and fitness technologies.

REFERENCES

- [1] Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. *Frontiers in Plant Science*, 7, 1419. <https://doi.org/10.3389/fpls.2016.01419>
- [2] Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representations (ICLR)*. <https://arxiv.org/abs/1409.1556>
- [3] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4510–4520.
- [4] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 770–778.
- [5] Tan, M., & Le, Q. V. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. *International Conference on Machine Learning (ICML)*, 6105–6114.
- [6] Singh, S., Yadav, A., & Pandey, S. K. (2021). YogaNet: A deep learning framework for yoga pose recognition using CNN. *Procedia Computer Science*, 173, 150–157.
- [7] Chen, Y., Huang, Q., & Wang, W. (2020). Deep transfer learning for yoga pose classification. *Journal of Ambient Intelligence and Humanized Computing*. <https://doi.org/10.1007/s12652-020-02463-z>
- [8] Cao, Z., Hidalgo, G., Simon, T., Wei, S. E., & Sheikh, Y. (2019). OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(1), 172–186.
- [9] Nie, W., & Wang, Y. (2019). Yoga posture classification using keypoint-based SVM. *International Conference on Artificial Intelligence and Big Data (ICAIBD)*, 228–232.
- [10] Haque, A., Peng, J., & Rehg, J. M. (2016). Towards viewpoint invariant 3D human pose estimation. *European Conference on Computer Vision (ECCV)*, 160–177.
- [11] Toshev, A., & Szegedy, C. (2014). Deeppose: Human pose estimation via deep neural networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1653–1660.
- [12] Jain, A., Tompson, J., Andriluka, M., Taylor, G. W., & Bregler, C. (2014). Learning human pose estimation features with convolutional networks. *International Conference on Learning Representations (ICLR)*.
- [13] Shotton, J., Girshick, R., Fitzgibbon, A., et al. (2013). Efficient human pose estimation from single depth images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(12), 2821–2840.
- [14] Ronneberger, O., Fischer, P., & Brox, T. (2015). U-Net: Convolutional networks for biomedical image segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, 234–241.
- [15] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 25, 1097–1105.
- [16] Lin, T. Y., Maire, M., Belongie, S., et al. (2014). Microsoft COCO: Common objects in context. *European Conference on Computer Vision (ECCV)*, 740–755.
- [17] Redmon, J., & Farhadi, A. (2018). YOLOv3: An incremental improvement. *arXiv preprint arXiv:1804.02767*. <https://arxiv.org/abs/1804.02767>
- [18] Wang, Y., Xie, X., & Zhang, Y. (2021). Human pose estimation for yoga using deep

CNN and LSTM networks. *Sensors*, 21(4), 1212.

- [19] Kumar, A., & Dahiya, K. (2020). Human pose estimation using deep learning: A review. *International Journal of Engineering and Advanced Technology (IJEAT)*, 9(3), 2249–8958.
- [20] Rani, A., Singh, J. P., & Goel, S. (2021). A survey on yoga pose classification using machine learning and deep learning techniques. *Journal of King Saud University – Computer and Information Sciences*. <https://doi.org/10.1016/j.jksuci.2021.05.011>