ISSN 1330-3651 (Print), ISSN 1848-6339 (Online)
https://doi.org/10.17559/TV-20240802001896
Original scientific paper

A Hybrid Approach of DenseNet121 with Attention and Bi-LSTM for Yoga Pose Estimation

Aarthy K.*, A. Alice NITHYA

Abstract: This study presents an Automated Pose Recognition system using Enhanced Chicken Swarm Optimization with Deep Learning (APR-ECSODL), a cutting-edge solution for identifying and categorizing human postures from images and videos with high accuracy. The system is designed to integrate advanced AI techniques, providing an innovative approach to pose recognition that leverages several sophisticated machine learning models and algorithms to enhance performance. The pre-processing stage involves applying a Wiener Filter (WF) for effective noise removal, ensuring that the data is clean and ready for analysis. Dynamic Histogram Equalization (DHE) is then employed to enhance image contrast, improving the visibility of key features within the images. For segmentation, the YOLOv8 model is used to isolate relevant regions of interest, providing a precise input for the next phase. Feature extraction is conducted using OpenPose, a widely recognized tool for obtaining key human body points. This step is crucial for capturing detailed information about the postures. The classification of these poses is performed using a Self-Attention Based Gated Recurrent Unit (SA-GRU) model. This model enhances accuracy by incorporating self-attention mechanisms, allowing the system to focus on significant features within the data. Performance optimization is achieved through the Enhanced Chicken Swarm Optimization (ECSO) method, which fine-tunes the parameters of the system to ensure optimal results. The APR-ECSODL technique was rigorously tested on a posture image classification dataset from Kaggle, demonstrating its effectiveness in categorizing various poses. By integrating these cutting-edge deep learning and AI methodologies, the APR-ECSODL system sets a new standard in pose recognition, offering a robust tool for applications in fields such as fitness monitoring, rehabilitation, and human-computer interaction. This approach not only ensures accurate pose identification but also enhances practicing quality and helps prevent errors, making it a valuable asset in diverse domains..

Keywords: automated pose recognition; deep learning; enhanced chicken swarm optimization; self-attention based GRU; YOLOv8

1 INTRODUCTION

Yoga is a Century-old kind of workout learned by athletes, sick people, and physiotherapists. Yoga is an activity that improves health by emphasizing links between the body, mind, and spirit. Yoga is a well-known and highly regarded discipline for leading a healthy lifestyle. The impact of asana may be due to either an increase in human body activity and energy expenditure or the anticipation of health benefits [1].

Yoga at home offers independence and comfort as well. When people work from home, their working conditions cannot be ideal. For instance, they might spend a lot of time sitting in uncomfortable positions, which can result in poor posture. Yoga enables one to stretch, enhance blood circulation, correct posture, and prevent the harmful consequences of prolonged sitting for at least eight hours [2]. Even on days when it is impossible for a person to commute to the studio, yoga may be practiced at home through online mode. Many now find it challenging to set aside time for exercise in the gym because of the fast-paced, expanding way of life and the prevalence of working from home.

Though many prefer to practice yoga at home, it is a great challenge to practice it without any guidance, which might result in major injury. The research mainly concentrates on identifying correct yoga postures and classifying those using deep learning approaches and machine learning [3].

Despite numerous advancements in artificial intelligence and computer vision, detecting human body positions remains a complex task. [4]. Irrespective of the challenges faced, many people prefer to use online technologies and feel at ease doing so in their homes. This leads to a high demand for developing a robust and accurate yoga pose detection and identification model.

In this paper, our work makes two significant contributions. Firstly, we propose a novel and innovative hybrid model called "Hybrid DenseNet Attention worked along with Bidirectional Long Short-Term Memory (Bi-LSTM)." This new model leverages the strengths of both DenseNet Attention and Bi-LSTM, enhancing the accuracy and efficiency of yoga pose estimation compared to traditional approaches. Secondly, we conduct a thorough comparative analysis of several model architectures, including VGG16 [5], VGG19 [6], Resnet 50 [7], and Densenet201 [7], for pose estimation in the context of yoga poses. We evaluate and compare their performance to identify their strengths and weaknesses in different scenarios. By conducting the comparative analysis and introducing the new hybrid model, our paper presents valuable insights and advancements in the field of yoga pose estimation, offering promising avenues for further research and development.

The experimental results are tested on a yoga pose dataset available in Kaggle [8] for fifteen different yoga poses, and it has been found that the performance of densenet201 is better than other architectures.

2 LITERATURE REVIEW

Aman Upadhyay et. al., [9] proposed the Y PN-MSSD model (yoga PoseNet Mobile Single shot Detector) based on TFlite Movenet, a system that combines Mobile-Net SSD and Pose-Net. Pose-net and Mobile-Net SSD are used for feature point detection and Mobile-Net SSD layer was used for human detection in each frame . Here seven yoga asanas are identified and their overall accuracy is 99.88% using their proposed Y_PN-MSSD model. This model's accuracy was based on the Mobile-Net SSD and Pose-Net pose evaluation.

Here the model is framed into three stages. In the first stage, a process called data collection involves capturing yoga postures from four different users, along with utilizing a public dataset containing seven yoga poses. Moving on to the second stage, the key points are extracted by the model and established connections between the key points in human body. Finally, third stage, the model is applied in real-time to identify yoga poses, and if necessary, it provides feedback to correct the user's posture.

In [10], Used Blaze Pose algorithm for estimating the key points from the input images. To identify key points, they integrated regression, heat maps, and offset. At first heat maps are used to control lightweight embedding, and so it can be used by regression encoder. So there was a balance between the low-level and high-level features because of the skip connection. On the input image, the model locates and evaluates the key points and also records localization. By localizing maximum body joints the model achieved 91% accuracy.

In [11], they used a model called YoNet and compared their work model with state-of-the-art image classification models - ResNet, Inception Net, Inception ResNet, and Xception. By using the input they have extracted spatial and depth information and taken those into consideration for additional posture detection calculations. They used only 5 different poses in this work. The above work results showed the accuracy of 94.91% with 95.61% precision.

In [12], a study of different deep learning and machine learning Models is used to predict the precise stance when all of the body parts were visible. Some bodily parts overlap with others while doing challenging yoga poses, making them difficult to see or feel. It is identified that present methods are not able to anticipate concealed body parts. The majority of large-scale posture estimate datasets consist of common stances used every day, make it challenging for the current algorithms to adequately handle complex poses. They claimed that complicated yogic posture detection could not be achieved by current yoga posture recognition methods. Additionally, research into three-dimensional body posture detection may be beneficial in the given work and may aid with future problems.

In [13], a model was utilized with DenseNet201 and Vgg19 to identify yoga poses on the given 3 levels. Level 1 is sitting pose; level 2 is legs in front; and the last level is bound angle pose. The input images used by them were of two types: skeletal image and normal image. Open Pose used for feature extraction and accuracy of the given model is used 82% for different poses; it increased by 7%, 11% and 12% for level 1, 2 and 3 respectively when analyzed separately. [14] uses six yoga poses and performed feature extraction. They used multi person pose estimation model and multilayer perceptron for classification and felt tough to classify many poses due to less variation among the poses. They achieved an accuracy of 99.58% in this work.

They employed SVM [15] for classification and the open pose model for feature extraction on a publicly accessible dataset. The model used here creates a virtual "skeleton" in 2D or 3D dimensions using key locations on the user's body that it has identified. The Model's overall performance hinges on its open-pose architecture, which may not function in situations like overlap between people and overlap between frames. For this system, work has been done on a portable self-education tool and a real-time prediction system. According to the authors, multi-character pose estimation was entirely a problem to be addressed along with limitations due to background, lighting, overlapping figures, etc. But this provides plenty of scope for research in yoga pose estimation. The training accuracy of this model is 98% even though there was some problem in validation result.

In [16], yoga82 dataset was used. They performed feature extraction using CenterNet and OpenPose followed by classification using 1D CNN and RNN. When compared to other models, OpenPose performs better when using 1D Convolutional Neural Networks. The accuracy of this work is 99.04% for a single frame and 99.38% for 45 frames of videos.

In [17], Alpha pose model was used for feature extraction followed by random forest for classification on yoga 82 dataset. Whenever occlusion and complex poses were involved, the model's performance is reduced. So a model which is resilient across many camera angles that are used in applications like automated yoga trainer, is used. To accomplish this, either more cameras need to be added to the training set (their results show that adding more cameras to the training set improves test performance), or computer vision techniques need to be used so as to make the models do not depend on camera angle. According to the outcome by analyzing unexplored camera angles they need to still improve the classification of yoga poses from various camera perspectives. Here Level 1: 91.21%, level 2: 87.91%, level 3: 80.14%.

In [18], authors generated a dataset by capturing images using webcam for 10 different yoga poses, then used Roboflow software for preprocessing. Here they performed various preprocessing and data augmentation steps before pose classification using VGG16. Their model after performing preprocessing and data augmentation was able to achieve an accuracy of 93% compared with old CNN architecture which yielded 76% only.

3 PROPOSED METHODOLOGY

In this work, we have proposed a novel approach for pose estimation that combines three powerful components: DenseNet for feature extraction, an Attention model for enhanced performance, and Bidirectional Long Short-Term Memory (Bi-LSTM) for final classification. This hybrid architecture aims to address the challenges of accurate and robust pose estimation from sequential data. Fig. 1 represents the architecture details of the proposed hybrid DenseNet Attention with Bi- LSTM.

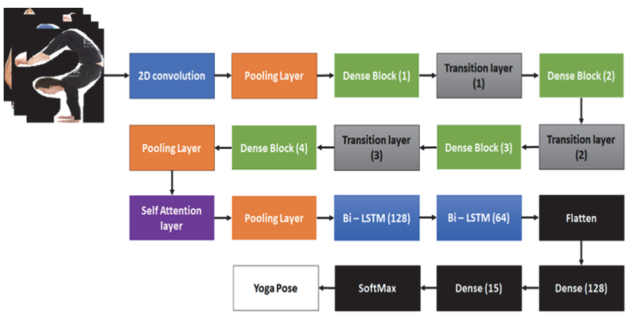

Figure 1 Proposed hybrid DenseNet attention with Bi-LSTM architecture

The details of DenseNet part was discussed in previous section.

Layers:

1. Convolution layer with Filter size is 64, kernel size is 7 x. Followed by convolution layer used batch normalization layer and activation layer.

2. MaxPooling layer with pool size 3 × 3, stride = 2, padding = same.
3. Dense block - 6 sets of convolution layers.
4. Transition layer has average pooling layer pool size 2 × 2.
5. Dense block - 12 sets of convolution layer.
6. Transition layer has average pooling layer pool size 2 × 2.
7. Dense block - 48 sets of convolution layer.
8. Transition layer has average pooling layer pool size 2 × 2.
9. Dense block - 32 sets of convolution layer.
10. Transition layer has average pooling layer pool size 2 × 2.
11. Self-attention layer.
12. MaxPooling layer with pool size 3 × 3, stride = 2, padding = same.
13. BiLSTM with 128 hidden layers.
14. BiLSTM with 64 hidden layers.
15. Fully connected layers.
16. Dense layer with 128 hidden layers.
17. Classification layer with softmax activation.

DenseNet for Feature Extraction: DenseNet is a convolutional neural network (CNN) architecture known for its dense connections between layers. Unlike traditional CNNs, where each layer takes input from the previous layer only, DenseNet allows for direct connections from every layer to every other layer. This dense connectivity encourages feature reuse and propagation of information throughout the network, leading to more efficient and meaningful feature representations. In our pose estimation framework, we leverage DenseNet to extract spatial features from the input pose data.

Feature extraction using OpenPose in the APR-ECSODL system enhances the system's ability to accurately detect complex poses, even in challenging conditions involving occlusions or overlapping structures. OpenPose's mechanisms such as Part Affinity Fields, multi-scale feature extraction, heatmaps, and confidence scoring equip APR-ECSODL with detailed and robust pose data, which is essential for precise pattern recognition and classification tasks. This high level of detail and robustness in key point extraction is fundamental to achieving accurate, real-time pose recognition in complex visual environments.

Attention Model for Enhanced Performance: To further enhance the performance of our pose estimation system, we incorporate an attention mechanism. Attention mechanisms enable models to prioritize the most relevant parts of input data while disregarding less informative regions. By directing the model's attention to important pose landmarks and body parts, we can improve feature extraction, especially in cases of occlusions or varying pose angles. This attention-based feature selection leads to more robust and discriminative representations, enabling our model to handle challenging pose variations effectively.

The combination of ECSO and chaotic PSO in the APR-ECSODL system offers a powerful synergy: ECSO provides structured exploration with adaptive hierarchy, while chaotic PSO enhances exploitation with randomized yet controlled dynamics. This hybrid approach ensures fast, stable, and high-precision optimization, making it ideal for feature selection, parameter tuning, and accurate pose recognition in the APR-ECSODL system. As a result, the overall model performance is significantly enhanced, with improved accuracy, stability, and efficiency across various pattern recognition and classification tasks.

Bidirectional Long Short-Term Memory (Bi-LSTM) for Final Classification: After feature extraction and attention-based feature selection, we pass the augmented feature sequence to a Bidirectional long-short-term memory (Bi-LSTM) network for final classification. LSTM is a specialized type of recurrent neural network (RNN) known for its exceptional ability to capture temporal dependencies in sequential data. By using the Bi-LSTM architecture, we allow the model to process the input sequence both forwards and backwards, enabling it to capture contextual information from past and future frames. This temporal modeling is crucial in pose estimation, as human body movements often exhibit complex temporal dynamics. The Bi-LSTM effectively encodes the temporal relationships between different pose landmarks, helping the model to better understand the sequential nature of pose sequences.

The overall architecture of our proposed method involves passing the input pose data through DenseNet for feature extraction. The attention mechanism is then applied to the extracted features to focus on relevant information. Finally, the attention-augmented feature sequence is fed into the Bi-LSTM network for classification. This multi-component hybrid architecture benefits from the robust feature extraction capabilities of DenseNet, attention-based feature selection, and the temporal modeling abilities of Bi-LSTM. As a result, it achieves improved accuracy and performance for pose estimation compared to conventional approaches that rely solely on individual components. An object detection model, variants of YOLO had been incorporated in segmentation layers, providing a lightweight option for applications where speed is prioritized over high precision in segmentation. YOLOv8 in the APR-ECSODL system enhances pose recognition by providing efficient, accurate, and real-time key point detection, allowing the system to assess the spatial orientation and alignment of objects in images effectively. This is key for achieving high-precision recognition of structures and patterns.

ResNet-50 is a popular variant of the Resnet (Residual Network) architecture. It is a deep CNN known for its exceptional performance on image classification. ResNet-50 introduces the concept of skip connections or identity mappings. By using this connection the information's are to flow directly from one layer to another and in between many layers are there which are bypassed. This helps in mitigating the problem of vanishing gradients and allows the network to learn more effectively, especially for deeper architectures. Its deep structure and skip connections contribute to its excellent performance. The deep structure of ResNet-50 makes it computationally expensive to train and deploy. Training the model requires significant computational resources and time, especially when working with large datasets. ResNet-50, like other deep neural network architectures, is prone to overfitting, especially if employing minimal training data. The parameters in the model are large numbers and can lead to over-parameterization, making it more vulnerable to overfitting. Regularization techniques and data augmentation are commonly used to address this issue.

4 RESULTS AND DISCUSSION

The Wiener Filter in APR-ECSODL improves the quality of the captured images by reducing noise, restoring signal clarity, and enhancing contrast, which is essential for the accuracy and reliability of pattern recognition in the system. True positives (TP_{events}) are accurately anticipated positive events, true negatives (TN_{events}) are accurately anticipated negative events, false positives (FP_{events}) are erroneously anticipated positive events, and false negatives (FN_{events}) are erroneously anticipated negative events, and these metrics are often employed in evaluating the performance of classification models.

1. *Accuracy*: Accuracy is a metric that measures the overall correctness of a classification model.

$$Accuracy = \frac{TP_{\text{events}} + TN_{\text{events}}}{TP_{\text{events}} + TN_{\text{events}} + FP_{\text{events}} + FN_{\text{events}}}$$

2. *Precision*: Precision is an indicator of the correctness of positive predictions made by a classification model. It quantifies the fraction of true positive predictions out of the total positive predictions.

$$Precision = \frac{TP_{\text{events}}}{TP_{\text{events}} + FP_{\text{events}}}$$

3. *Recall* (Sensitivity or True Positive Rate): Recall is an indicator of the model's ability to correctly recognize positive events. It calculates the fraction of TP_{events} predictions out of the actual positive events.

$$Recall = \frac{TP_{\text{events}}}{TP_{\text{events}} + FN_{\text{events}}}$$

4. $F1$ Score: A single statistic called the $F1$ score integrates recall and accuracy into a single quantity. The $F1$ score is the harmonic average of precision and recall, providing a balanced measure of model performance.

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

5. *Specificity*: Specificity is an estimation of the model's ability to correctly identify negative instances. It calculates the true negative rate, which is the fraction of correct negative predictions out of all actual negative instances. The specificity formula is

$$Specificity = TN_{\text{events}} / (TN_{\text{events}} + FP_{\text{events}})$$

6. Confusion matrix: A confusion matrix, also known as an error matrix, is a table that provides a summary of a classification model's performance by showing the counts of true positive (TP), true negative (TN), false positive (FP), and false negative (FN) predictions. It summarizes the performance of a classification model. It is an effective technique for assessing the precision and excellence of a categorization method.

Fig. 2 and Fig. 3 represent Training and Testing performance for DenseNet201, VGG16, VGG19, and ResNet50, respectively. Fig. 4 represents the testing confusion matrix for DenseNet201, VGG16, VGG19, ResNet50, and the proposed model. From the testing results, the proposed model attained increased *accuracy* by 1.22%, 0.80% increased *precision*, 1.22% increased *recall*, 1.222% increased $F1$ score, and 0.1% increased *Sensitivity* when compared with VGG16. The proposed model attained increased *accuracy* by 2.38%, 3.78% increased *precision*, 2.38% increased *recall*, 2.34% increased $F1$ score, and 0.1% increased *Sensitivity* when compared with ResNet50.

The proposed model attained increased *accuracy* by 3.98%, 3.8% increased *precision*, 3.9% increased *recall*, 3.99% increased $F1$ score, and 0.2% increased *Sensitivity* when compared with VGG19. The proposed model attained increased *accuracy* by 0.47%, 0.16% increased *precision*, 0.47% increased *recall*, 0.53% increased $F1$ score, and 0.05% increased *Sensitivity* when compared with DenseNet 201.

From the results, it clearly shows that the proposed model provides the best performance when compared to Densenet201, VGG16, VGG19, and ResNET50. Fig. 5 represents the testing ROC curve for DenseNet201, VGG16, VGG19, ResNet50, and the proposed model. The confusion matrix shows that in resnet.

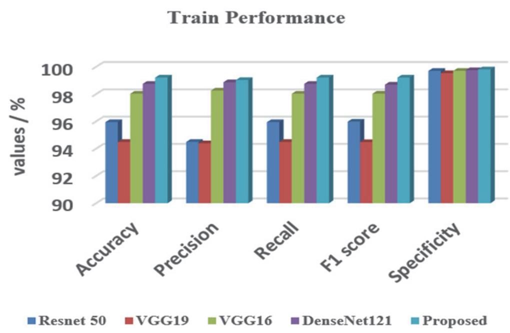

Figure 2 Training performance metrics

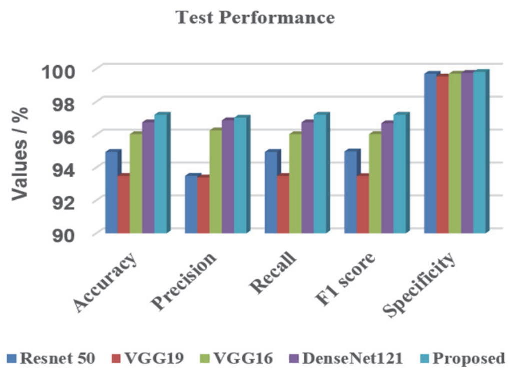

Figure 3 Testing performance metrics

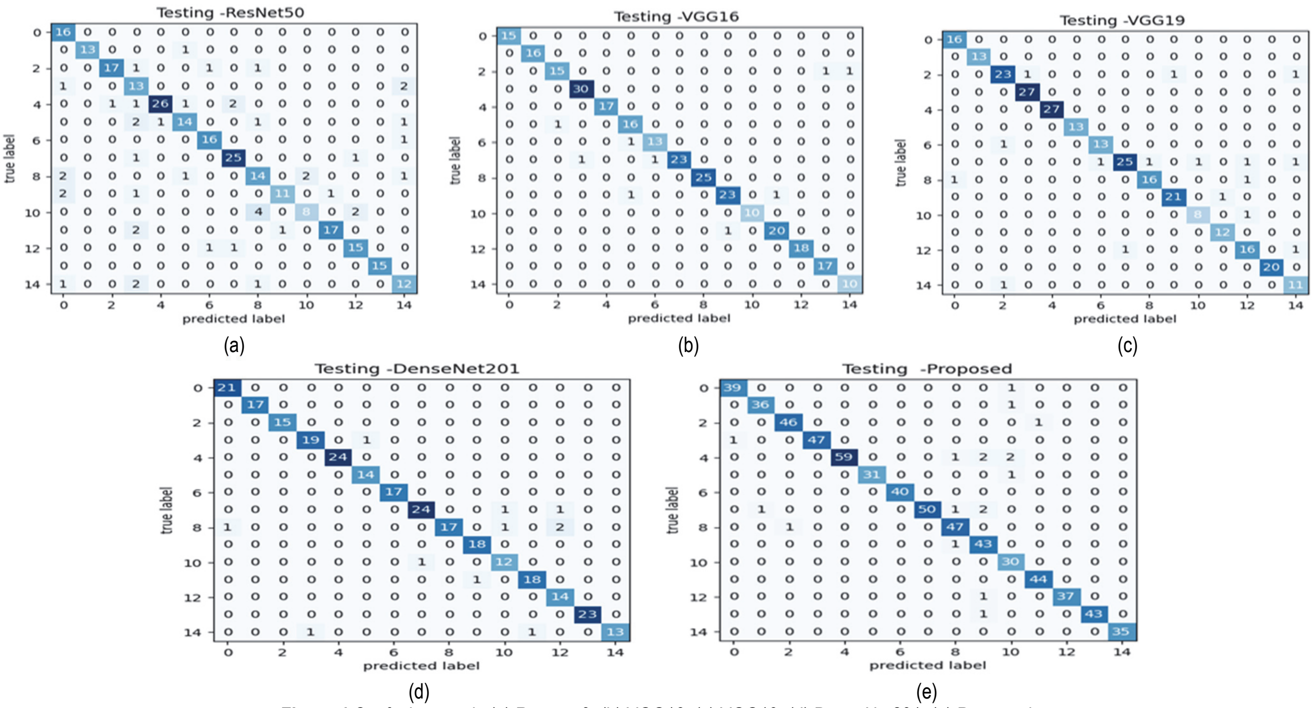

Figure 4 Confusion matrix (a) Resnet50, (b) VGG16, (c) VGG19, (d) DenseNet201, (e) Proposed

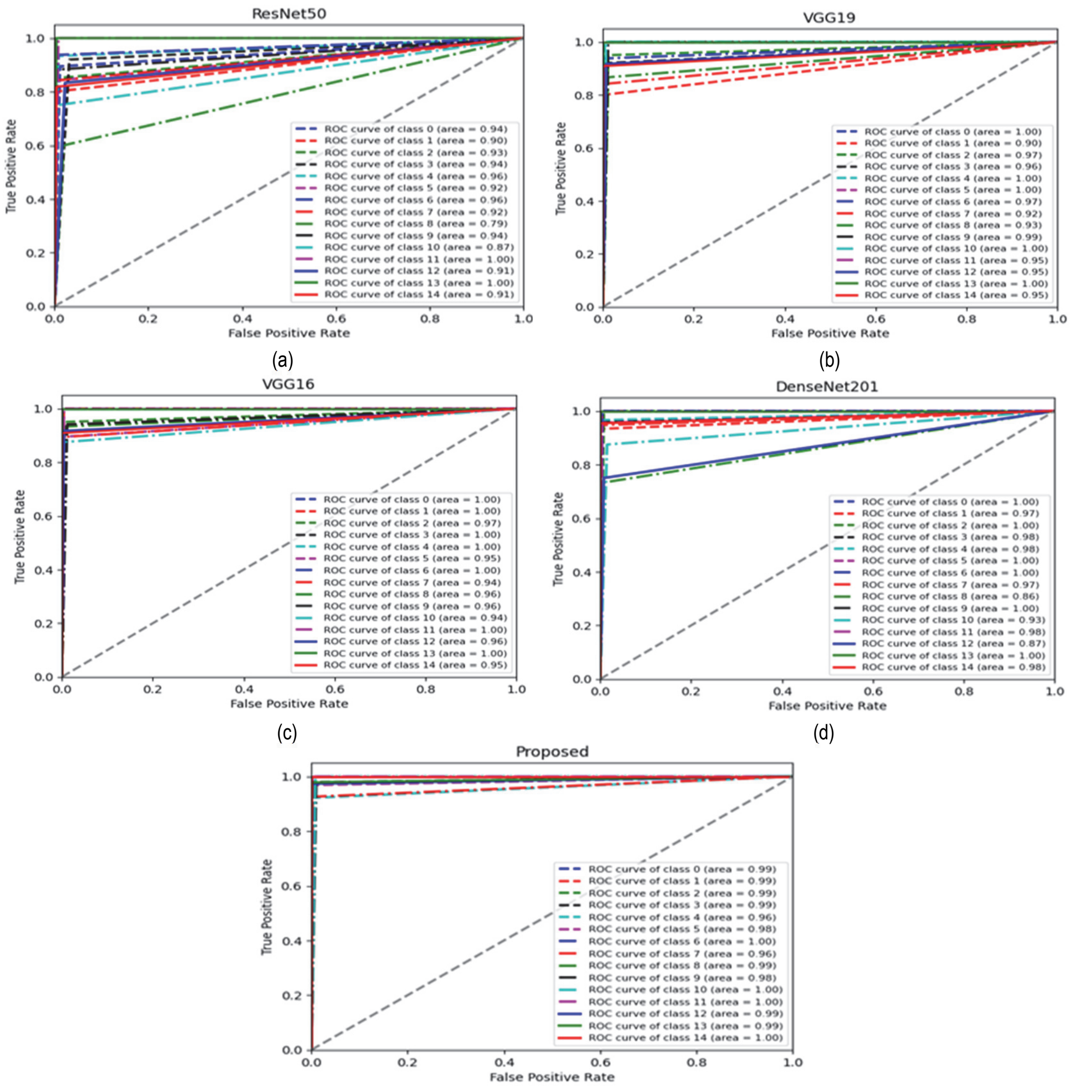

Figure 5 ROC curve for (a) Resnet50, (b) VGG19, (c) VGG16 (d) DenseNet201 (e)nProposed

The accuracy performance of the proposed work in comparison to other works is shown in Fig. 6. Prior literary works primarily concentrated on three to eight poses. In comparison to the other priority work, densenet201 work achieves the best performance. However, when compared to other works, our suggested model achieves an accuracy improvement of 4.26%.

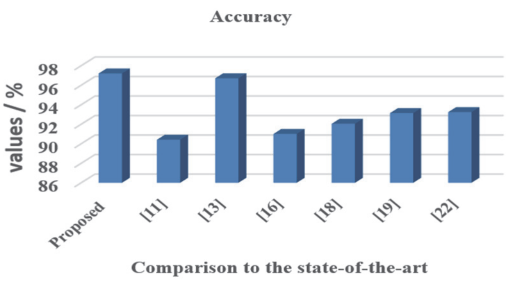

Figure 6 Comparison of the state of the art work with proposed model

5 CONCLUSION

In this study, we focus on the design and developmentof an AYPR-ECSODL technique. In the pre-processing stage, the WF is applied for noise removal, enhancing the image quality by reducing unwanted artefacts. Besides, the DHE is employed for contrast enhancement, ensuring that relevant features within the image are highlighted effectively. Besides, the segmentation step is executed through the YOLOv8 model. Feature extraction is accomplished via OpenPose, which extracts essential human body key points that serve as the foundation for further analysis. The classification stage employs a SA-GRU model. To optimize model performance, parameter tuning is performed using the ECSO method. The developed technique validated yoga posture image classification from the Kaggle dataset.

6 REFERENCES

[1] Ashraf, F. B., Islam, M. U., Kabir, M. R., & Uddin, J. (2023). Yonet: A neural network for yoga pose classification. *SN Computer Science, 4*(2), 198. https://doi.org/10.1007/s42979-022-01618-8

[2] Ghongane, A. (2022). Hierarchical Classification of Yoga Poses using Deep Learning Techniques. Dublin: National College of Ireland.

[3] Kundu, K. & Goswami, A. (2023). Classification of Yoga Poses Using Integration of Deep Learning and Machine Learning Techniques. *Proceedings of International Conference on Recent Trends in Computing: ICRTC 2022, Springer Nature Singapore*. 417-428. https://doi.org/10.1007/978-981-19-8825-7_36

[4] Jose, J. & Shailesh, S. (2021). Yoga asana identification: a deep learning approach. IOP Conference Series: *Materials Science and Engineering, 1110*(1), 012002. https://doi.org/10.1088/1757-899X/1110/1/012002

[5] Palanimeera, J. (2023). Integrating Bio Mechanics And Deep Learning For Yoga Pose Classification Leveraging Joint Angle Estimation.

[6] Bhosale, V., Nandeshwar, P., Bale, A., & Sankhe, J. (2022). Yoga pose detection and correction using posenet and KNN. *International Research Journal of Engineering and Technology, 9*(4), 1290-1293.

[7] JV, N. L. & Bagaria, C. K. (2021). Yoga Pose Classification using Resnet of Deep Learning Models. *i-Manag Journal of Computer Science, 9*(2), 29. https://doi.org/10.26634/jcom.9.2.18464

[8] Yoga Poses Dataset. Retrieved from Kaggle: https://www.kaggle.com/datasets/yoga-poses-dataset

[9] Upadhyay, A., Basha, N. K., & Ananthakrishnan, B. (2023). Deep learning-based yoga posture recognition using the Y_PN-MSSD model for yoga practitioners. *Healthcare, 11*(4), 609. https://doi.org/10.3390/healthcare11040609

[10] Jain, S., Rustagi, A., Saurav, S., Saini, R., & Singh, S. (2021). Three-dimensional CNN-inspired deep learning architecture for Yoga pose recognition in the real-world environment. *Neural Computing and Applications, 33*, 6427-6441. https://doi.org/10.1007/s00521-020-05405-5

[11] Garg, S., Saxena, A., & Gupta, R. (2022). Yoga pose classification: a CNN and MediaPipe inspired deep learning approach for real-world application. *Journal of Ambient Intelligence and Humanized Computing,14*, 16551–16562. https://doi.org/10.1007/s12652-022-03910-0

[12] Palanimeera, J. & Ponmozhi, K. (2023). YAP_LSTM: yoga asana prediction using pose estimation and long short-term memory. *Soft Computing*, *2023*. https://doi.org/10.1007/s00500-023-09044-5

[13] Bhaumik, U., Singh, K. K., Akbari, A. S., & Bajpai, M. K. (2022). A Deep Learning-Based Approach to Detect Correct Suryanamaskara Pose. *SN Computer Science, 3*(5), 337. https://doi.org/10.1007/s42979-022-01226-6

[14] Jadhav, R., Ligde, V., Malpani, R., Mane, P., & Borkar, S. (2023). Aasna: Kinematic Yoga Posture Detection And Correction System Using CNN. In ITM Web of Conferences, 56, 05007. https://doi.org/10.1051/itmconf/20235605007

[15] Yang, T., Zhou, S., Xu, A., Ye, J., & Yin, J. (2023). An Approach for Plant Leaf Image Segmentation Based on YOLOV8 and the Improved DEEPLABV3+. *Plants, 12*(19), 3438. https://doi.org/10.3390/plants12193438

[16] Nguyen, P. K., Nguyen, A. T., Doan, T. B., Trung, P. N., & Thi, N. D. (2022). Assessing Bicep Curl Exercises by Human Pose Application: A Preliminary Study. *International Conference on Soft Computing and Pattern Recognition*, *Springer Nature Switzerland*, 581-589. https://doi.org/10.1007/978-3-031-27524-1_55

[17] Choi, W., Kim, M. J., Yum, M. S., & Jeong, D. H. (2022). Deep convolutional gated recurrent unit combined with attention mechanism to classify pre-ictal from interictal EEG with minimized number of channels. *Journal of Personalized Medicine, 12*(5), 763. https://doi.org/10.3390/jpm12050763

[18] Wu, Y., Liu, J., Wang, L., An, Y., & Zhang, X. (2023). Distribution Network Reconfiguration Using Chaotic Particle Swarm Chicken Swarm Fusion Optimization Algorithm. *Energies, 16*(20), 7185. https://doi.org/10.3390/en16207185

Contact information:

Aarthy K.
(Corresponding Author)
Department of Computational Intelligence
SRM Institute of Science and Technology
Kattankulathur, Chennai - 603203
Tamil Nadu, India
ak3110@srmist.edu.in

A. Alice Nithya
Department of Computational Intelligence
SRM Institute of Science and Technology
Kattankulathur, Chennai - 603203
Tamil Nadu, India
alicenia@srmist.edu.in