Applied Computer Systems
ISSN 2255-8691 (online)
ISSN 2255-8683 (print)
2025, vol. 30, no. 1, pp. 75–84
https://doi.org/10.2478/acss-2025-0009
https://content.sciendo.com

An Approach using Skeleton-based Representations and Neural Networks for Yoga Pose Recognition

Hai Thanh Nguyen[1*], Nguyen Nhat Truong[2], Linh Thuy Thi Pham[3], Ngoc Huynh Pham[4]
[1–2]*College of Information and Communication Technology, Can Tho University, Can Tho, Vietnam*
[3]*Faculty of Information Technology, Can Tho University of Technology, Can Tho, Vietnam*
[4]*FPT University, FPT Polytechnic, Can Tho, Vietnam*

***Abstract* – Amid a rapidly developing era, people can inevitably have problems with stress, depression, pressure, or difficulty sleeping due to frequent overthinking. To overcome the above problems, yoga will be an excellent solution to help adjust thoughts and harmonize body and soul, helping us relax, relax the mind, and retain positive thoughts. Negative and evil auras will be pushed away, and the worldview will improve. Yoga practice has incorrectly caused many unwanted injuries for practitioners. Therefore, we present an approach grounded in skeleton-based feature extraction and neural networks to find a solution to the recognition of yoga postures, creating a premise for researching a smart virtual trainer that supports home workouts for users from input image data converted into skeleton data through MoveNet. The classification models were used to train recognition and classification of yoga poses. The models were trained and evaluated on a dataset of 3939 images of 10 yoga poses. Experimental results show that the proposed algorithms are entirely suitable for the classification task when achieving good results on different metrics such as Precision, Recall, F1-score, and Accuracy.**

***Keywords* – Classification, image classification, MoveNet, pose recognition, skeleton data, skeleton-based feature, yoga pose.**

I. Introduction

People's need for health and beauty care has increased in recent years. Sports training methods to meet that essential need, including sports activities, have become popular – activities such as walking, jogging, swimming, aerobics, cycling, yoga, etc. Exercise is perfect for health, but we need to understand the sport we are practicing to not waste time and effort because each sport has its technical requirements, and beginners will not necessarily practice correctly. These subjects help train the body and increase movement, but yoga also has unique movements and benefits compared to other exercise subjects. Yoga is an exercise associated with physical and mental strength. As reported in [1], when practicing yoga, we must perform this exercise correctly, especially in the correct posture. It has been observed that sometimes, due to lack of support or knowledge, people do not know the effective method to practice yoga and start practicing yoga without any guidance, falling over themselves in the process [2]. Exercising due to incorrect training methods leads to injury. Yoga should be done under the guidance of a trainer, but this is not something everyone can do. According to [3], of the 2434 people who completed the survey in the UK, the majority said that their physical and mental condition had improved positively through yoga practice. In addition, a yoga performance at My Dinh yard, Hanoi, with the participation of 5000 people[2] with the message "Forever healthy and beautiful with yoga" was organized and officially established a Vietnam Record with "The yoga performance with the largest number of participants at the same time in Vietnam" showed some of the influence of yoga today. In addition, data from Wellness Creatives[3] showed that the yoga industry has been considered a "gold mine" when the size of the yoga market in the US reached 14.5 billion USD, growing on average 9.8 % per year, and was expected to reach 215 billion USD by 2025.

With the vigorous development of computer science, especially computer vision and artificial intelligence (AI), many potential development directions exist for human life today. Action recognition is also a widespread area of research in computer vision. Its applications include surveillance systems, video analytics, robotics, and various systems involving human interaction and electronic devices. In particular, human posture recognition is also one of the problems that new technologies approach to solving smartly and effectively. Many research projects use different methods and technologies to recognise yoga poses. The authors [4] presented an overview of different yoga poses and ways to minimise or prevent injuries caused by incorrect yoga postures. At the same time, the study provided insight into current research challenges. It proposed using different convolutional networks such as VGG16, VGG19, and Inception V3 for effective pose recognition task yoga. Alternatively, research [5] compared the performance of its model with several state-of-the-art image classification models – ResNet, InceptionNet, InceptionResNet, and Xception,

[*] Corresponding author's e-mail: nthai.cit@ctu.edu.vn
Article received 2025-02-17; accepted 2025-05-06

[2] https://tienphong.vn/post -1559672.tpo
[3] https://nhipcaudautu.vn/phong-cach-song/mo-vang-yoga-3347264/

showing that the proposed model architecture outperforms other models' previous classification models. The proposed architecture extracted spatial and depth features from images individually and considered them for further computation in classification. In [6], yoga postures were classified based on 3D data. In particular, with a data set of 2129 images of 13 yoga poses, different classification models were tested, including KNN, SVM, and some popular deep neural networks such as AlexNet, VGGNet, and ResNet. The experimental results show that, with the collected data set, the classification model with the highest Performance is AlexNet (83.05 %).

The 3D CNN architecture [7] designed to classify yoga poses achieved a test recognition accuracy of 91.15 % on an in-house prepared yoga pose dataset consisting of 10 yoga poses. Besides, [8] also proposed a yoga pose classification method based on 3D data. Research classifying ten yoga poses using the LSTM [10] model achieved expected accuracy (92.34 %). Deepak Kumar and Anurag Sinha [9] proposed a Backing Vector Machine, Convolutional Neural Network (CNN), and CNN + LSTM models to identify and classify six popular yoga movements and achieve desirable results. Besides, Chhaihuoy Long, Eunhye Jo, and Yunyoung Nam [10] have developed a yoga self-training system using the Transfer Learning technique to predict yoga postures and confirm direction feedback. This group of authors has proposed the most optimal model among the six proposed models and achieved an accuracy of 98.43 % for yoga pose recognition. In [11], authors tested different machine learning techniques for classifying yoga poses. To solve that problem, they prepared a dataset of 5500 images for ten yoga poses and then applied a pose estimation algorithm to extract skeletal images in real time. The results showed that the Random Forest Classifier worked well for the collected data set. In [12], a group of students and associate professors at the Department of Information Technology, Vidyalankar Institute of Technology, Mumbai, India, proposed a KNN classification model using PoseNet to process input data to detect and correct yoga posture. From the proposed model, the expected accuracy was 98.51 %.

In [13], authors proposed yoga pose classification through the CNN deep learning method and used the MediaPipe library to extract skeletons for the evaluation process between training models with image data and skeleton data. This research achieved positive results; on image input data (without skeleton extraction), the VGG16 model achieved the highest accuracy (95.6 %), followed by InceptionV3, NASNetMobile, YogaConvo2d (the proposed model) (89.9 %), and finally, InceptionResNetV2. On the contrary, since the input data were images extracted from skeletons, the proposed model YogaConvo2d had an accuracy of 99.62 %, followed by VGG16, InceptionResNetV2, NASNetMobile, and InceptionV3. The research [14] used CenterNet and OpenPose for the task of converting images into skeleton data and 1D CNN and RNN for the task of classification of yoga poses. By comparing the experimental results, the performance of OpenPose with 1D CNN gave better results than other models. Besides, [15] research also used OpenPose to improve the accuracy and performance of posture estimation systems by applying OpenPose keypoint detection. With a data set of about 11 000 images collected, the proposed method produced superior results compared to existing models. By comparing the experimental results, the performance of OpenPose with 1D CNN provided better results than other models. Besides, [15] research also used OpenPose to improve the accuracy and performance of posture estimation systems by applying OpenPose keypoint detection. With a data set of about 11 000 images collected, the proposed method produced superior results compared to existing models.

The study [16] proposed a method to classify six yoga poses based on a data set collected from 15 volunteers. The authors used different machine learning algorithms for classification, including logistic regression, SVM, KNN, Random Forest, and naive Bayes. Besides, with the collected data, the study extracted skeleton data before putting it into the classification model and achieved desirable results with a random forest accuracy of 94 %. In another study [17], 12 yoga poses were classified according to skeletal data on four different models. The results show that the proposed KNN model has a high classification performance of 96 % and faster training time than the other models.

Most research on identifying yoga postures based on skeleton-extracted images has brought high results. In this article, we deploy the direction of training a recognition model with data. Skeletal input is extracted from the image, providing the most accurate yoga pose recognition results. We aim to build a website system that combines the FNN1D and Conv1D models to allow practitioners to easily access and perform yoga movements more accurately, making practicing this subject more convenient. Our contributions to this topic include the following:

- By applying advances in machine learning algorithms, through data preprocessing with MoveNet, the project has deployed and evaluated the FNN1D and Conv1D models for the classification task and achieved desirable results. With FNN1D, the accuracy on the test set is 0.97676 and Conv1D is 0.98135.
- Integrating models into the website makes it easy for users to access and provides good support for yoga pose recognition. It creates a premise to expand the system into a virtual yoga instructor to support self-practice for people who want to practice this subject.

The main structure of this article is presented below. Section II explains related concepts, presents the framework model of the algorithm proposed in this article, and designs the algorithm used in standard yoga pose recognition. Section III-B verifies the algorithm through experiments, and Section IV concludes the article.

II. Methodology

In this section, we present the main techniques of the system for recognising yoga postures in skeleton data with the execution sequence of the recognition model presented in Fig. 1.

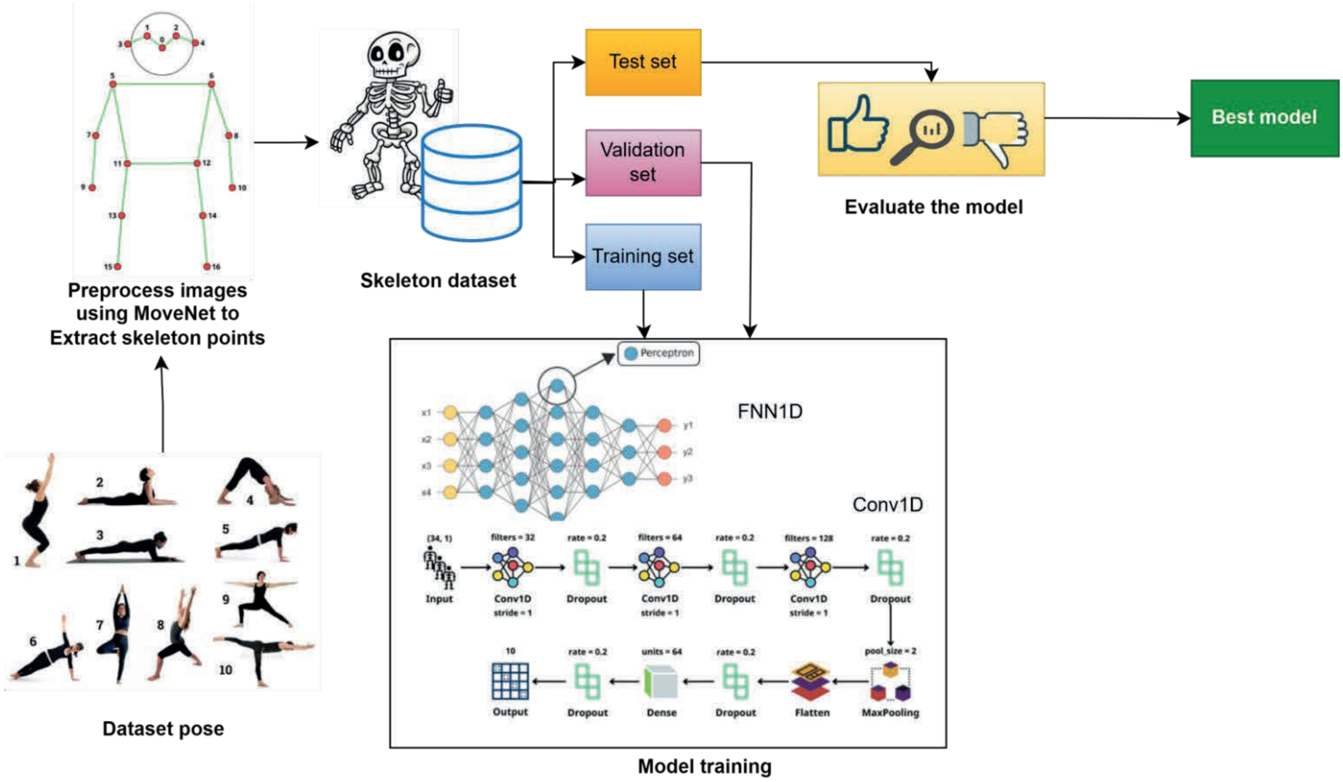
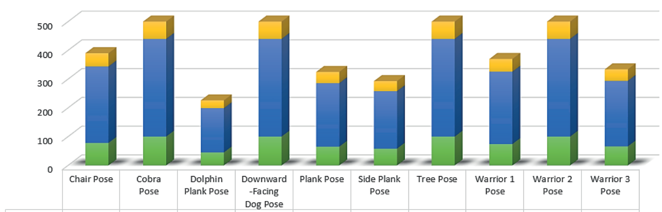


Fig. 1. The overall architecture for yoga pose recognition.

A. Description of the Data Set and Data Preprocessing

We crawled data from various sources, including Yoga-82 [18], Kaggle[4] and the Internet. From that data, we selected an appropriate data set. We obtained a total of 3939 images of 10 famous and simple yoga poses for beginners (including chair pose, cobra pose, dolphin plank pose, downward-facing dog pose, plank pose, side plank pose, tree pose, warrior one pose, warrior two pose, and warrior three poses). Images of each yoga pose are collected from many sources, so the sizes of the images of yoga poses vary; the average size of the data set was about 460 574 pixels. After that, we divided the data with the training set that represented 68 %, the validation set accounting for 12 %, and the test set accounting for 20 %, see Fig. 2.



	Chair Pose	Cobra Pose	Dolphin Plank Pose	Downward-Facing Dog Pose	Plank Pose	Side Plank Pose	Tree Pose	Warrior 1 Pose	Warrior 2 Pose	Warrior 3 Pose
Validation	46	60	27	60	39	35	60	44	60	40
Training	266	340	154	340	221	200	340	252	340	228
Test set	78	100	45	100	65	58	100	74	100	66

Test set Training Validation

Fig. 2. The sample distribution of each yoga pose.

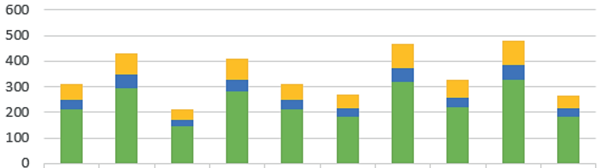
We focused on training the model on skeleton data, so we preprocessed the image data into skeleton data through MoveNet. The MoveNet model [19] is a super-fast and accurate deep learning model developed by Google Research to extract information about the skeleton (including 17 essential points on the body) in images or videos. This number of 17 points is determined based on the trade-off between model detail and performance to meet the actual requirements of the application. The model is designed to track and recognise human skeletal points in real time. TF Hub offers this model in two variants: Lightning and Thunder[5]. The Lightning variant is designed for applications that require low latency, while the Thunder variant is made for applications that require high precision. Taking advantage of the MoveNet model's power, we proceed to use it to preprocess input image data. Specifically, we use the Thunder variant to increase the accuracy in extracting skeletons from the input image after applying MoveNet to obtain a feature vector storing the extracted skeletons. Each feature vector stores the coordinates of 17 points on the human body in the image that will be used as input to the classification model.

After extracting the skeleton from the input image data (.jpg and .png), we obtain a file (.csv) consisting of 54 attribute columns, including 51 attribute columns storing 17 skeletal points on the body and three attribute columns storing the file name, layer number, and layer name, the results of the

[4] https://www.kaggle.com/datasets/niharika41298/yoga-poses-dataset

[5] https://analyticsindiamag.com/inside-movenet-googles-latest-pose-detection-model/

extracted skeleton data are shown in Fig. 3. As during the skeleton extraction process, MoveNet ignores invalid images (images without colour and images with no skeleton detected), the number of elements after extraction may be less than the original.



	Chair Pose	Cobra Pose	Dolphin Plank Pose	Downward-Facing Dog	Plank Pose	Side Plank Pose	Tree Pose	Warrior 1 Pose	Warrior 2 Pose	Warrior 3 Pose
Test set	61	84	40	80	63	55	95	70	97	52
Validation set	37	52	26	49	37	32	56	39	58	32
Training set	212	295	144	280	211	182	318	219	325	182

Training set Validation set Test set

Fig. 3. Skeletal data for each yoga pose is divided into each dataset.

B. Machine Learning Techniques for Classification Tasks

The main task in yoga pose recognition on skeletal data is to classify the label or name of the yoga pose, also known as classification. Many machine learning architectures are currently built for classification and have achieved unexpected results. We built popular architectures such as Feedforward Neural Networks and Convolutional Neural Networks in this topic. As the input data is a skeleton feature vector, the architectures are built with input as skeleton data, a 1D vector of 34 elements, and output as yoga pose labels (includes ten classes: chair pose, cobra pose, dolphin plank pose, downward-facing dog pose, plank pose, side plank pose, tree pose, warrior one pose, warrior two pose, and warrior three pose). While building the Feedforward Neural Network and Convolutional Neural Network models, it is better to build from simple to more complex models and evaluate the results of the models on the ACC measure. From the results obtained, it is necessary to compare to find the model with the best results (in the project, Convolutional Neural Network). At the same time, the proposed models should be used to run experiments on 2D data through data augmentation to compare the effectiveness of data preprocessing by means of the MoveNet extraction model skeleton.

Feedforward Neural Network architecture: Feedforward Neural Network (FNN) is a type of neural network architecture in machine learning and artificial intelligence [20]. A feedforward neural network consists of a series of fully connected layers, where each neuron in the previous layer connects to all the neurons in the following layer. Feedforward neural networks are structure-independent, meaning they do not make any particular assumptions about the network input, so they are often used in many machine learning applications.

In this study, we applied the Feedforward Neural Network model on skeleton data with the input being a 1D vector (34 elements), followed by 4 Dense layers (or fully-connected layers) with the numbers of neurons of 64, 128, 128, 256, respectively for dense layers, using the non-linear activation function ReLU, along with a dropout layer of 0.2 after each Dense layer and an output size of 10 representing the ten layers (chair pose, cobra pose, dolphin plank pose, downward-facing dog pose, plank pose, side plank pose, tree pose, warrior one pose, warrior two pose and warrior three pose). The activation function is widely used in classification problems through the Adam optimizer [21], with learning rate = 0.0005, using the 'categorical cross-entropy' loss function to calculate the real loss between the predicted label and the actual label. The architecture is illustrated in Fig. 4a.

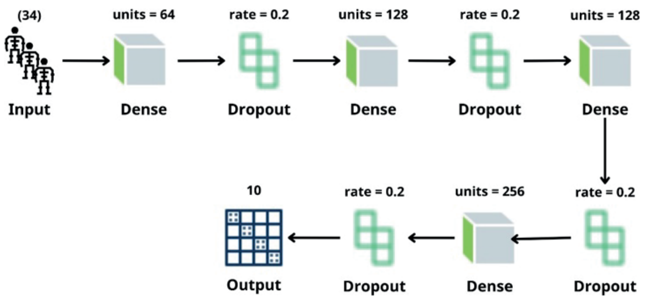
Convolutional Neural Network architecture: We used a CNN architecture on 1D data (depicted in Fig. 4b) with the input size of the model being a 34-element feature vector. This CNN architecture [22] consists of three convolutional layers (Conv1D) followed by MaxPooling layers, flattening layer, and fully connected layer. The convolutional layers in this study contain 32, 64, and 128 filters, respectively, with each filter size being 3, a stride being 1, and a ReLU activation function. The MaxPooling layer is used with a size of 2, followed by a flattening layer and a fully connected layer of 64 units. Next, the output is the Dense layer, with each neuron receiving input from all neurons of the previous layer. The activation function used in this layer is softmax with a value of 10 representing ten classes to be classified. Adam optimization function was with learning rate = 0.0005, using the "categorical cross-entropy" loss function. The model is built with a total number of about 171 thousand parameters.

III. Experiments

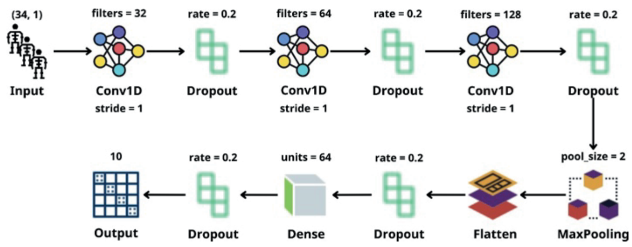
A. Settings and Hyperparameter Evaluation

Our research experimented on the Python programming language platform using open-source libraries to support machine learning models. Training models with image input data were performed on the Google Colab environment. The model training process with skeleton data and the recognition model used for the web system was carried out on Windows 11 64bit operating system, with configuration Intel(R) Core(TM) i5-1135G7 2.40 GHz, and RAM 16 GB.

The website system was built and installed according to the client-server model. In particular, the localhost computer was considered a server, where training was performed, recognition models, and storing databases were incorporated; the models were trained, and the requests sent from the client were processed and then returned (responses) to the user. Client is the one that sends requests to the server and receives returned results (responses). The system's website was built with supporting languages, libraries, and frameworks: HTML, CSS, Javascript, and Bootstrap to create a beautiful, intuitive, and easy-to-use web interface. As for the backend, a Python Framework called Flask was used to create an API that allowed integration to run machine learning models, which recognised yoga poses and returned recognition results trained and stored on the server. On the client side, to use the system, users need a web browser: Google Chrome, Microsoft Edge, Coc Coc, Firefox, or any browser that can connect to the address. Regarding evaluation parameters, we used the metrics to evaluate the training effectiveness of the machine learning model: Precision, Recall, F1-score, and Accuracy (ACC), and used the confusion matrix to evaluate the performance of the model – layered shape.

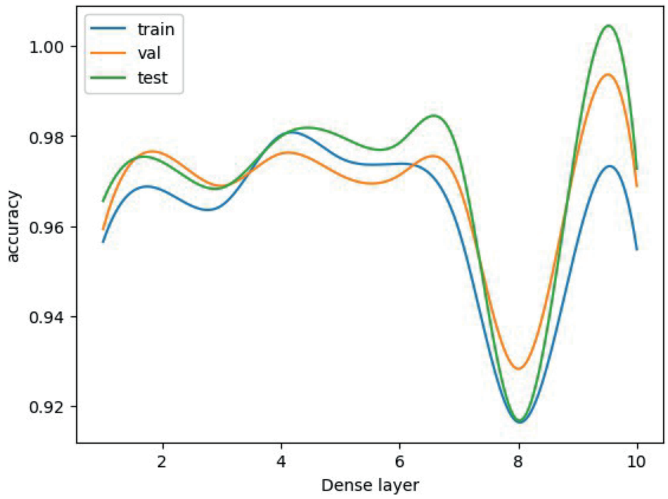


(a) FNN

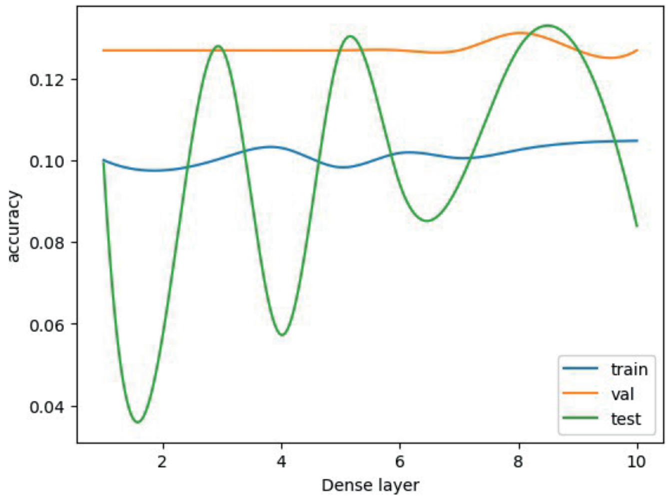


(b) CNN

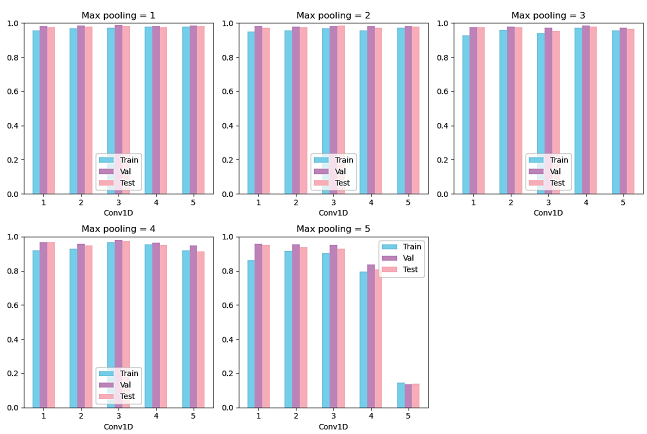
Fig. 4. Some illustrations of yoga pose classification on the skeleton.



(a) FNN

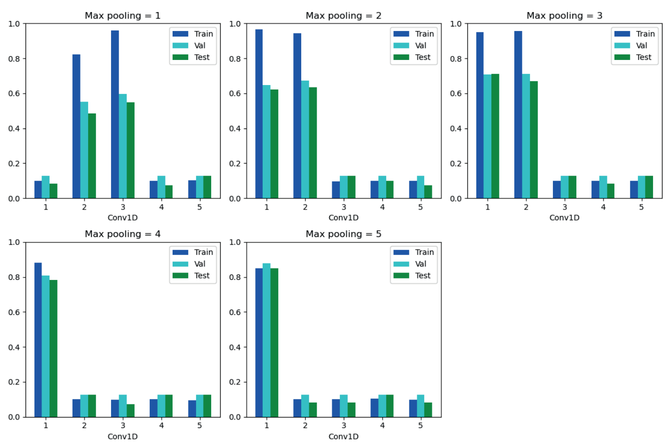


(a) FNN



(b) CNN

Fig. 5. Performance of various configurations on the skeleton.



(b) CNN

Fig. 6. Performance of various configurations on the images.

B. Results and Discussion

For the FNN architecture, we built models with different Dense layers, ranging from 1 to 10, to find the optimal model. The experimental results are shown in Fig. 5a. For CNN architecture, we built the model with the change of 1D convolutional layer (Conv1D) and Maxpooling layer according to the formula: $(n \times Conv1D) + (m \times Maxpooling)$ in which each Maxpooling layer will change n Conv1D layers before it (from 1 to 5), and change m Maxpooling layers of the model (from 1 to 5). The optimal model in this topic is 1 Maxpooling layer and 3 Conv1D layers; the results are shown in Fig. 5b.

Besides, to compare the effectiveness of preprocessing skeleton data, we trained FNN and CNN models with image data by applying data augmentation techniques on the training set (with each class having 400 samples); the results of training FNN and CNN models with input image data of size 150×150×3 (colour image) are shown in Figs. 6a and 6b. Applying the method of preprocessing image data into skeleton data contributed to extracting features of yoga pose images, thereby increasing the recognition efficiency of the classification model. We found the optimal classification models on skeleton data, respectively, the Feedforward Neural Network model with 4 Dense layers (FNN1D-4) and the Convolutional Neural Network model with 1 Maxpooling layer and 3 Conv1D layers (Conv1D-1-3). To increase the reliability of the classification model, we experimentally ran the FNN1D-4 and Conv1D-1-3 models to collect the average accuracy over five runs. The results are that the FNN1D-4 model achieved an overall accuracy of 0.97676, standard deviation of 0.00447, and the Conv1D-1-3 model achieved an overall accuracy of 0.98135, standard deviation of 0.00144 (see Fig. 7). We also evaluated the Precision, Recall, and F1-score results of the classes over five runs with a training epoch of 80, batch_size = 32 and used EarlyStopping to increase performance and save time in model training. With monitor = “val_accuracy” and patience = 10, the results obtained across the layers of the FNN1D-4 and Conv1D-1-3 models are relatively high and reliable. Besides, Conv1D-1-3 model performs better than FNN1D-4.

Figure 8a shows the accuracy change on the training and validation sets of the FNN1D-4 model over each epoch. Through five runs with FNN1D-4 models, the accuracy reaches its maximum on the validation set at epochs 27, 30, 21, 18, 19 with an accuracy of 0.97368, 0.97847, 0.96890, 0.96651, 0.97129, and on the training set the accuracy is 0.97128, 0.98015, 0.96706, 0.96199, 0.96706. Also, we can observe that the loss is stable from epoch 15 as shown in Figure 8b.

Figure 9a shows the accuracy change on the training set and validation set of the Conv1D-1-3 model over each epoch. Through five runs, we see that the accuracy reaches its maximum at epochs 21, 27, 20, 14, 19 with an accuracy of 0.98804, 0.98804, 0.98804, 0.98565, 0, 98804 on the training set, and on the validation set the accuracy is 0.97804, 0.97804, 0.97762, 0.97002, 0.98057. Figure 9b shows the change in the loss function in the training set and the validation set of the Conv1D-1-3 model over each epoch. Over five runs, the model’s loss fluctuated around 0.25, showing that the model was relatively stable and did not lose too much. The loss fluctuates strongly from epoch 0 to 5 and stabilizes after that.

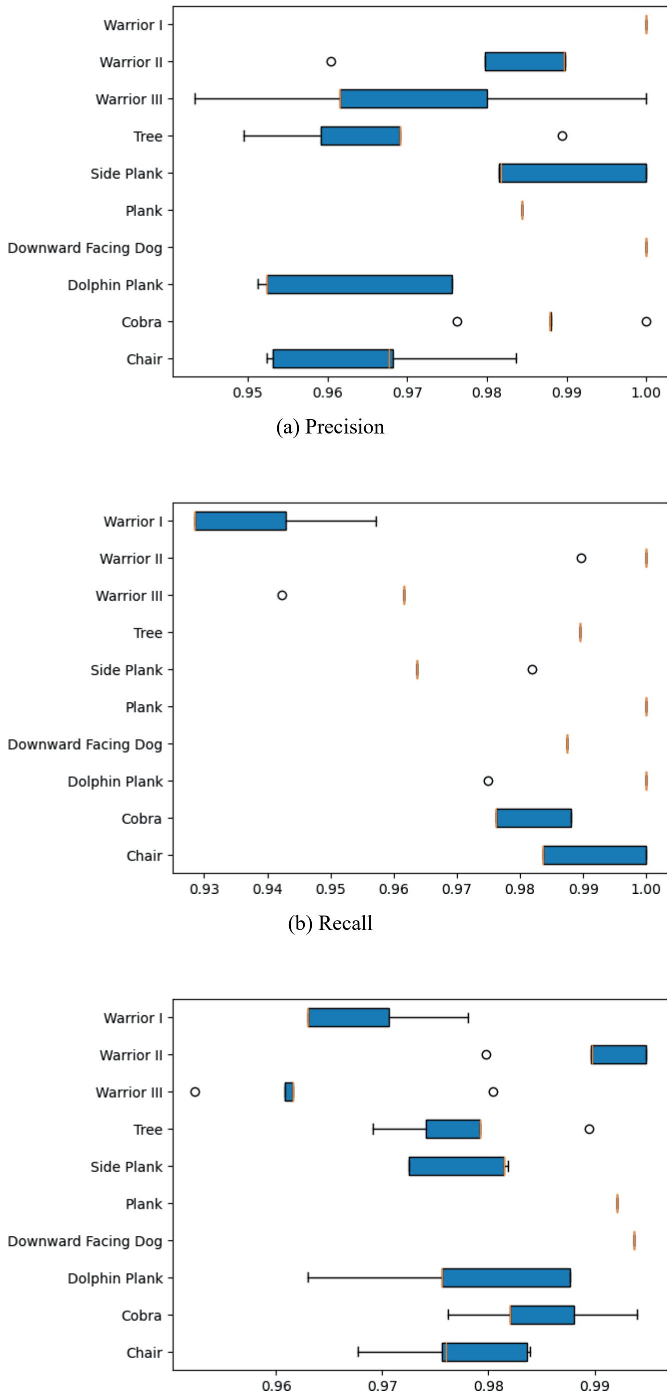
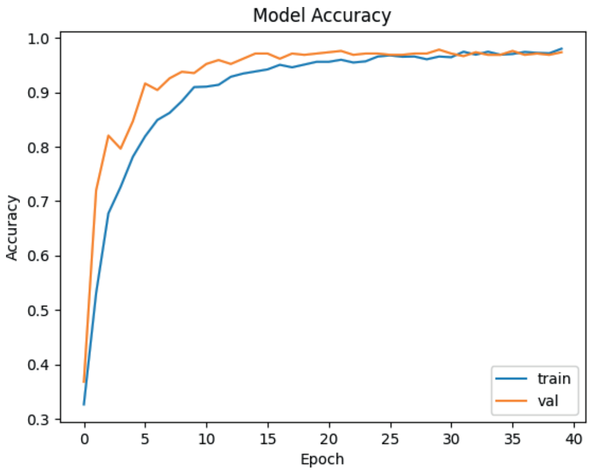
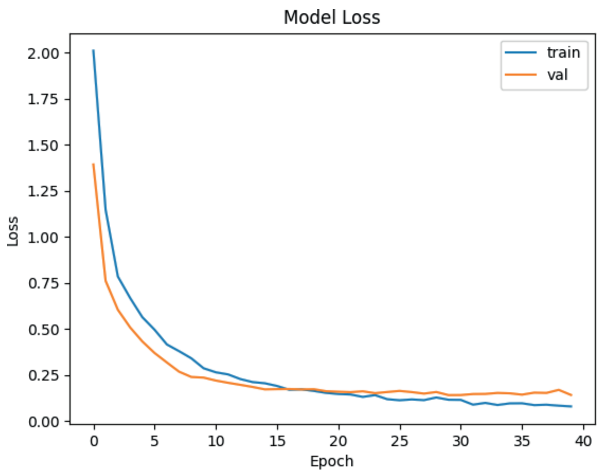


Fig. 7. Average results of the Conv1D-1-3.

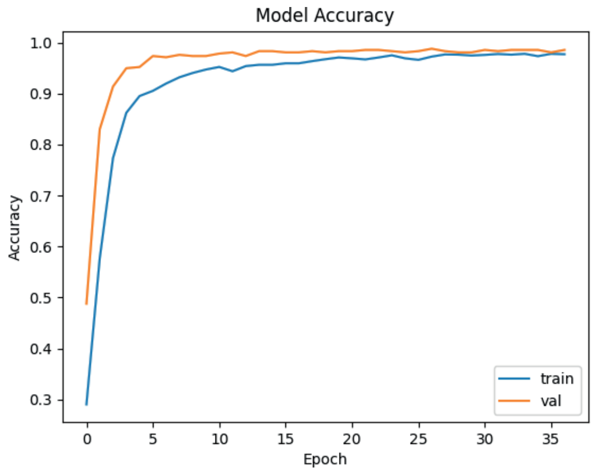


(a) Accuracy

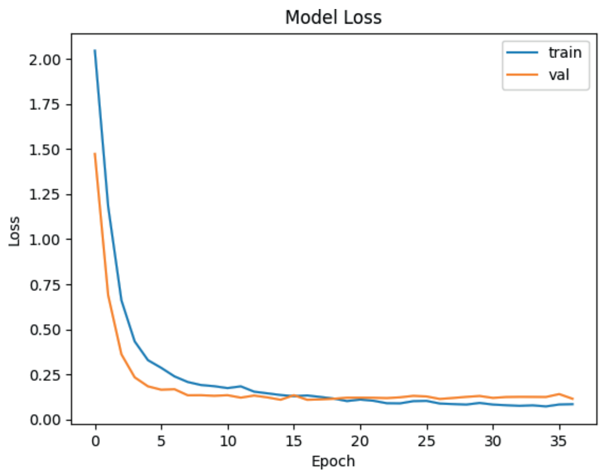


(b) The loss

Fig. 8. Accuracy and the loss through each epoch of the FNN1D-4 model.



(a) Accuracy

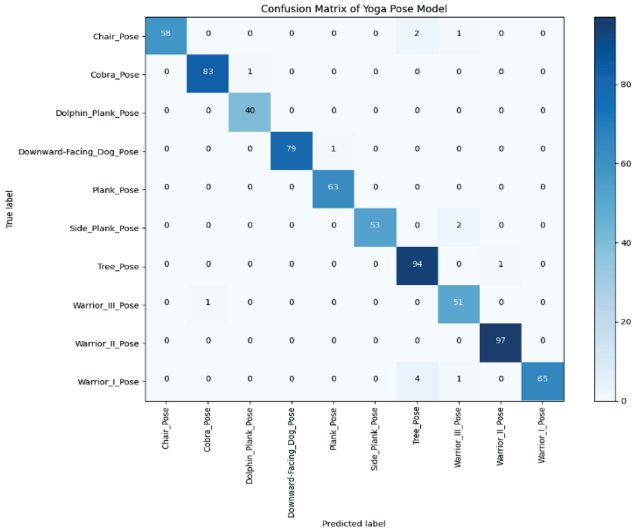


(b) The loss

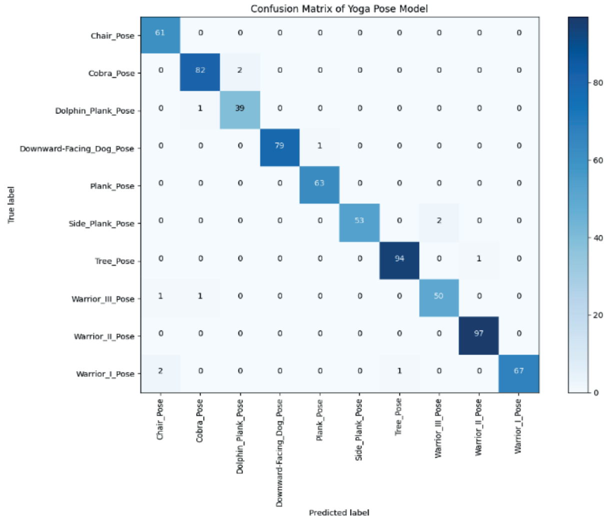
Fig. 9. Accuracy and the loss through each epoch of the Conv1D-1-3 model.

Then, we evaluated the classification model through five training times using the confusion matrix of the FNN1D-4 model. Figure 10a shows the very good prediction results of the FNN1D-4 model across each class. Besides, to visually see the accuracy of the classification model, the project evaluated the model through five training times using the standardized confusion matrix of the FNN1D-4 model. From the results of evaluating the standardized confusion matrix of the FNN1D-4 model through Figure 10b, it can be seen that with the skeleton data set, the FNN1D-4 model gives prediction results that are very good across classes. Most classes achieved accuracy above 92 %.

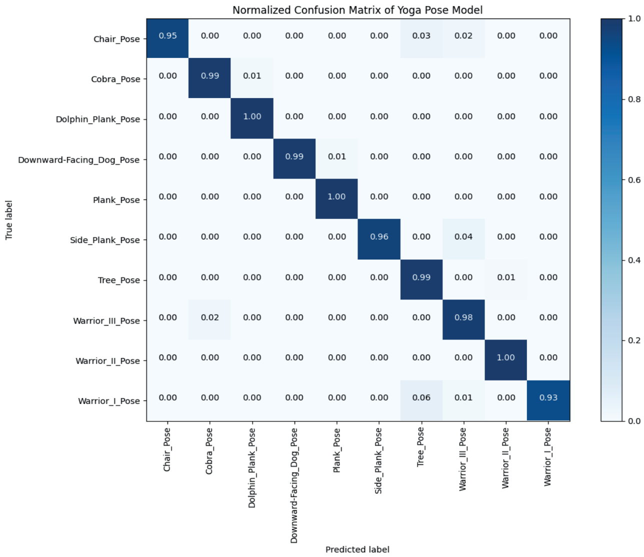
We illustrated the classification model over five training times through the confusion matrix of the Conv1D-1-3 model. Figure 11a shows that the Conv1D-1-3 model has better prediction results than the FNN1D-4 model in each class. Besides, to visually see the accuracy of the Conv1D-1-3 model, we evaluated the results of predicting classes through five training times through the standardized confusion matrix. With the results of predicting each class through the standardized confusion matrix of the Conv1D-1-3 model (Fig. 11b), we comment that with the skeleton data set, the Conv1D-1-3 model gives excellent prediction results across classes. Most of the classes achieved an accuracy of more than 93 %, in which the Planck Pose and Warrior II Pose classes almost achieved absolute accuracy with the test data set. The confusion matrix and normalized confusion matrix results of the FNN1D-4 and Conv1D-1-3 models above show that the classification model effectively predicts all. In all classes, there is no “rote learning” situation.



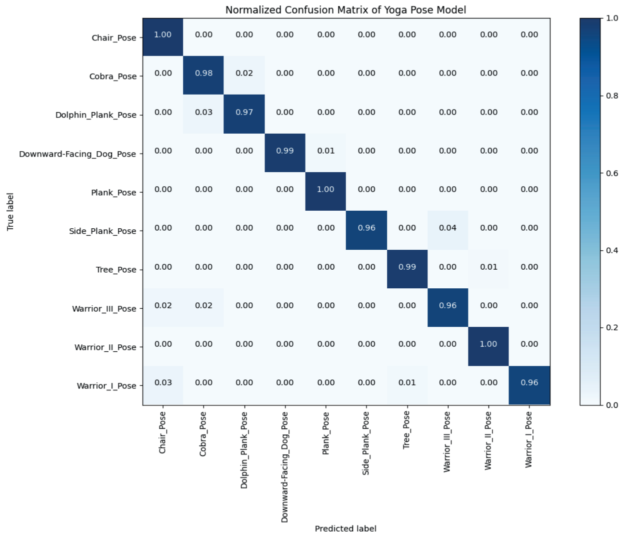
(a) Confusion matrix of FNN1D-4 Model



(a) Confusion matrix of Conv1D-1-3 Model



(b) ConFNN1D-4 model via the normalized



(b) Conv1D-1-3 model via the normalized

Fig. 10. Confusion matrices of ConFNN1D-4.

Fig. 11. Confusion matrices of Conv1D-1-3.

C. Comparison with the Previous Study

In addition to the built FNN1D-4 and Conv1D-1-3 models, we conducted a classification on the “Yoga-82” [23] through five runs and compared with other models. Variants of the DenseNet-201 model have the variant parameters (18.27M, 18.27M, and 22.59M). Figure 12 shows how to divide the classifications of the Yoga-82 data set, in which the data set has 28 478 yoga posture images divided into three levels with the number of classes being 6, 20, and 82, respectively. Specifically, Level 3 is the leaf node of the diagram with 82 subclasses, Level 2 includes 20 subclasses, and Level 1 includes six subclasses. We used the Yoga-82 dataset to evaluate the performance of the FNN1D-4 and Conv1D- 1-3 models. When loading the Yoga-82 data set, due to the URL of some yoga pose images being corrupted, the number of images collected from the Yoga-82 data set was reduced, and 20 541 images were downloaded. Figures 13a and 13b show the average accuracy results over five runs of the FNN1D-4 model and the Conv1D-1-3 model on the data set Yoga-82 of subclasses 6, 20 and 82. The results of the accuracy comparison (depicted in Fig. 14) show that the model we built can save memory (FNN1D-4 has 62 666 parameters, and Conv1D-1-3 has 171 018 parameters) and is more optimal than the models proposed in the study [23].

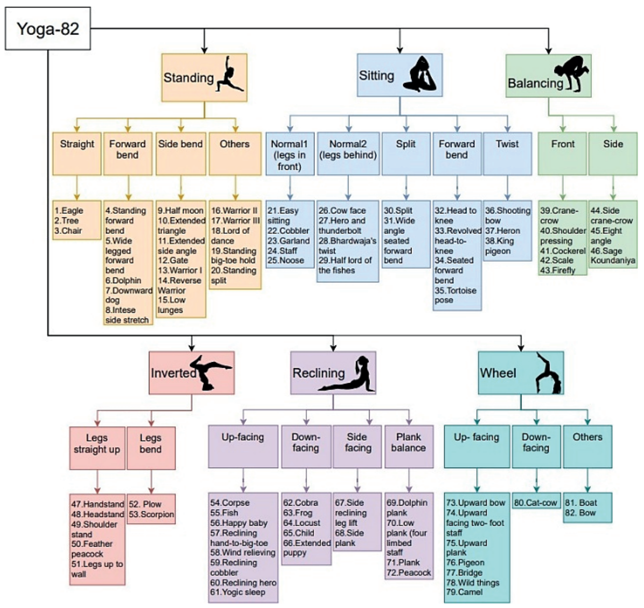
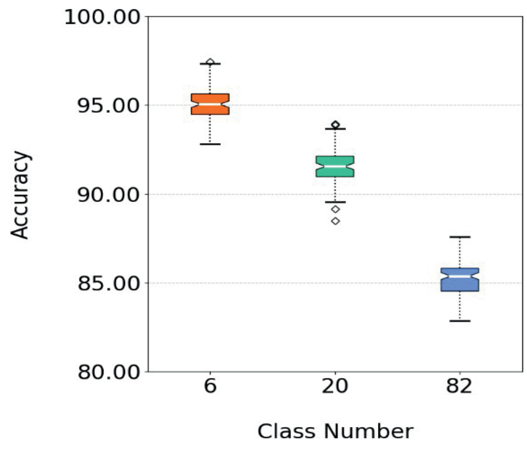
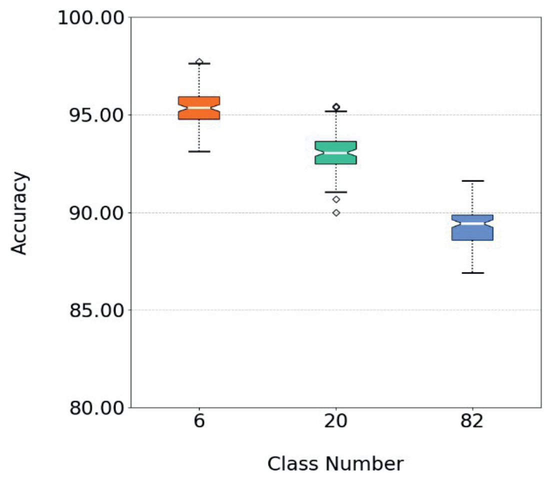


Fig. 12. An illustration of the label structure from Yoga-82 dataset [23].



(a) FNN1D-4



(b) Conv1D-1-3

Fig. 13. The performance on the Yoga-82 dataset.

IV. Conclusion

Yoga is a classical system of practice originating from India. It combines physical exercise, meditation, and breathing techniques to achieve balance and harmony between mind and body. Practicing yoga brings many health benefits, such as improving body shape and maintaining a balanced body shape. It also helps improve emotions and reduce anxiety and stress.

There are many different types of yoga, each with its approach and goals, and yoga can be practiced at many levels, suitable for everyone from beginners to experienced people. After the research process of applying machine learning models to identify yoga postures on skeletal data, we successfully built machine learning models FNN and CNN, then performed training and evaluation on the image and skeleton datasets. Hence, choosing the most suitable approach for recognition results in high accuracy and short running time. With experimental results over five runs, the FNN1D-4 model achieved an accuracy of 0.97676 ± 0.00447, and the Conv1D-1-3 model achieved an accuracy of 0.98135 ± 0.00144. The classification model was built with several parameters and achieved higher accuracy than the DenseNet-2001 models in the research [23]. At the same time, we built a website that integrates trained models. It provides user management, model management, and recognition history management functions to help identify yoga poses quickly and conveniently.

Conflict of Interest Statement

All authors declare that they have no conflicts of interest.

Availability of Data

The data supporting this research findings are available at [18].

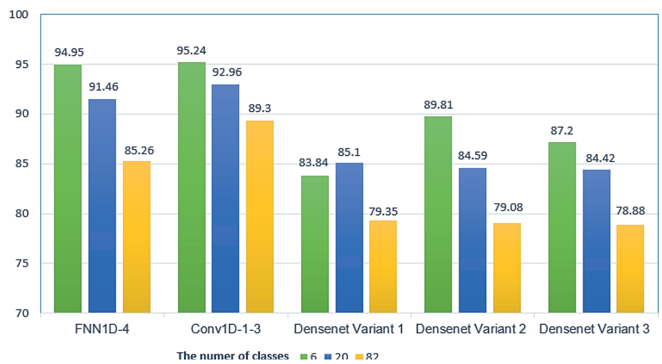


Fig. 14. Comparison with the previous study, including Densenet-201 Variant 1, Variant 2, Variant 3 of the study [18].

References

[1] R. Govindaraj, S. Karmani, S. Varambally, and B. Gangadhar, “Yoga and physical exercise – a review and comparison,” *International Review of Psychiatry*, vol. 28, no. 3, pp. 242–253, Apr. 2016. https://doi.org/10.3109/09540261.2016.1160878

[2] C. Brems, L. Justice, K. Sulenes, L. Girasa, J. Ray, M. Davis, J. Freitas, M. Shean, and D. Colgan, “Improving access to yoga: Barriers to and motivators for practice among health professions students,” *Adv. Mind Body Med.*, vol. 29, no. 3, pp. 6–13, 2015. https://www.academia.edu/19663263/Improving_access_to_yoga_Barriers_and_motivators_for_practice_among_health_professions_students

[3] T. Cartwright, H. Mason, A. Porter, and K. Pilkington, “Yoga practice in the UK: A cross-sectional survey of motivation, health benefits and behaviours,” *BMJ open*, vol. 10, no. 1, 2020, Art. no. e031848. https://doi.org/10.1136/bmjopen-2019-031848

[4] S. Goel, S. Mohanty, and S. Markanday, “Classification of yoga pose using pre-trained CNN models and machine learning classifiers,” in *2022 IEEE International Conference on Current Development in Engineering and Technology (CCET)*, Bhopal, India, Dec. 2022, pp. 1–6. https://doi.org/10.1109/CCET56606.2022.10080048

[5] F. B. Ashraf, M. U. Islam, M. R. Kabir, and J. Uddin, “YoNet: A neural network for yoga pose classification,” *SN Computer Science*, vol. 4, Feb. 2023, Art. no. 198. https://doi.org/10.1007/s42979-022-01618-8

[6] C. Nagalakshmi and S. Mukherjee, “Classification of yoga asanas from a single image by learning the 3D view of human poses,” in *Digital Techniques for Heritage Presentation and Preservation*, Springer, Mar. 2021, pp. 37–49. https://doi.org/10.1007/978-3-030-57907-4_3

[7] S. Jain, A. Rustagi, S. Saurav, R. Saini, and S. Singh, “Three- dimensional CNN-inspired deep learning architecture for yoga pose recognition in the real-world environment,” *Neural Computing and Applications*, vol. 33, pp. 6427–6441, 2021. https://doi.org/10.1007/s00521-020-05405-5

[8] R. P. Srivastava, L. S. Umrao, and R. S. Yadav, “Real-time yoga pose classification with 3-D pose estimation model with LSTM,” *Multimedia Tools and Applications*, vol. 83, pp. 33019–33030, Sep. 2023. https://doi.org/10.1007/s11042-023-17036-8

[9] D. Kumar and A. Sinha, *Yoga Pose Detection and Classification Using Deep Learning*. LAP LAMBERT Academic Publishing London, 2020.

[10] C. Long, E. Jo, and Y. Nam, “Development of a yoga posture coaching system using an interactive display based on transfer learning,” *The Journal of Supercomputing*, vol. 78, pp. 5269–5284, Mar. 2022. https://doi.org/10.1007/s11227-021-04076-w

[11] Y. Agrawal, Y. Shah, and A. Sharma, “Implementation of machine learning technique for identification of yoga poses,” in *2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT)*, Gwalior, India, Apr. 2020, pp. 40–43. https://doi.org/10.1109/CSNT48778.2020.9115758

[12] V. Bhosale, P. Nandeshwar, A. Bale, and J. Sankhe, “Yoga pose detection and correction using Posenet and KNN,” *International Research Journal of Engineering and Technology*, vol. 9, no. 4, pp. 1290–1293, 2022.

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

[14] S. Abarna, V. Rathikarani, and P. Dhanalakshmi, “Skeleton pose estimation features based classification of yoga asana using deep learning techniques,” *International Journal of Mechanical Engineering*, 2022.

[15] J. Z. Tan, C. P. Lee, K. M. Lim, and J. Y. Lim, “Yoga pose estimation with machine learning,” in *2023 11th International Conference on Information and Communication Technology (ICoICT)*, Melaka, Malaysia, Aug. 2023, pp. 260–265. https://doi.org/10.1109/ICoICT58202.2023.10262445

[16] U. Bahukhandi and S. Gupta, “Yoga pose detection and classification using machine learning techniques,” *Int. Re.s J. Mod. Eng. Technol. Sci.*, vol. 3, no. 12, pp. 13–15, 2021.

[17] J. Palanimeera and K. Ponmozhi, “Classification of yoga pose using machine learning techniques,” *Materials Today: Proceedings*, vol. 37, pp. 2930–2933, 2021. https://doi.org/10.1016/j.matpr.2020.08.700

[18] M. Verma, S. Kumawat, Y. Nakashima, and S. Raman, “Yoga-82: A new dataset for fine-grained classification of human poses,” in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Seattle, WA, USA, Jun. 2020, pp. 4472–4479, https://doi.org/10.1109/CVPRW50498.2020.00527

[19] B. Jo and S. Kim, “Comparative analysis of OpenPose, PoseNet, and MoveNet models for pose estimation in mobile devices,” *Traitement du Signal*, vol. 39, pp. 119–124, Feb. 2022. https://doi.org/10.18280/ts.390111

[20] G. Arulampalam and A. Bouzerdoum, “A generalized feedforward neural network architecture for classification and regression,” *Neural Networks*, vol. 16, no. 5–6, pp. 561–568, June–July 2003. https://doi.org/10.1016/S0893-6080(03)00116-3

[21] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, Dec. 2014. https://doi.org/10.48550/arXiv.1412.6980

[22] D. Bhatt, C. Patel, H. Talsania, J. Patel, R. Vaghela, S. Pandya, K. Modi, and H. Ghayvat, “CNN variants for computer vision: History, architecture, application, challenges and future scope,” *Electronics*, vol. 10, no. 20, Oct. 2021, Art. no. 2470. https://doi.org/10.3390/electronics10202470

[23] M. Verma, S. Kumawat, Y. Nakashima, and S. Raman, “Yoga-82: a new dataset for fine-grained classification of human poses,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, Seattle, WA, USA, Jun. 2020, pp. 1038–1039. https://doi.org/10.1109/CVPRW50498.2020.00527

Hai Thanh Nguyen is an Associate Professor at the College of Information and Communication Technology, Can Tho University, Vietnam. He received his Master degree in Computer Science and Engineering from National Chiao Tung University, Taiwan, and his PhD degree in Computer Science from Sorbonne University, France. His current research includes bioinformatics, healthcare systems, recommendation systems, and applications using computer vision, machine learning, and data analysis.
E-mail: nthai.cit@ctu.edu.vn
ORCID iD: https://orcid.org/0000-0002-1386-1390

Nguyen Nhat Truong completed a specialized training programme in Information Technology at the Information and Communication Technology College, Can Tho University, Vietnam. He is interested in artificial intelligence and machine learning.
E-mail: itnntruong@gmail.com

Linh Thuy Thi Pham is a Lecturer at the Faculty of Information Technology, Can Tho University of Technology. Her research interests include computational intelligence, artificial intelligence, biomedical engineering, image processing, and computer vision.
E-mail: pttlinh@ctuet.edu.vn
ORCID iD: https://orcid.org/0009-0005-7338-0520

Ngoc Huynh Pham is currently a Lecturer at FPT University – FPT Polytechnic Can Tho, Vietnam. She has a deep passion and interest in information technology, especially in areas such as applications based on machine learning and computer vision.
Email: ngocph11@fpt.edu.vn,
ORCID iD: https://orcid.org/0000-0001-6023-3189