Computers, Materials & Continua
DOI: 10.32604/cmc.2023.034654
Article

Squirrel Search Optimization with Deep Convolutional Neural Network for Human Pose Estimation

K. Ishwarya and A. Alice Nithya*

Department of Computational Intelligence, SRM Institute of Science and Technology, Kattankulathur, 603203, Tamilnadu, India
*Corresponding Author: A. Alice Nithya. Email: alicenia@srmist.edu.in
Received: 23 July 2022; Accepted: 14 October 2022

Abstract: Human pose estimation (HPE) is a procedure for determining the structure of the body pose and it is considered a challenging issue in the computer vision (CV) communities. HPE finds its applications in several fields namely activity recognition and human-computer interface. Despite the benefits of HPE, it is still a challenging process due to the variations in visual appearances, lighting, occlusions, dimensionality, etc. To resolve these issues, this paper presents a squirrel search optimization with a deep convolutional neural network for HPE (SSDCNN-HPE) technique. The major intention of the SSDCNN-HPE technique is to identify the human pose accurately and efficiently. Primarily, the video frame conversion process is performed and pre-processing takes place via bilateral filtering-based noise removal process. Then, the EfficientNet model is applied to identify the body points of a person with no problem constraints. Besides, the hyperparameter tuning of the EfficientNet model takes place by the use of the squirrel search algorithm (SSA). In the final stage, the multiclass support vector machine (M-SVM) technique was utilized for the identification and classification of human poses. The design of bilateral filtering followed by SSA based EfficientNet model for HPE depicts the novelty of the work. To demonstrate the enhanced outcomes of the SSDCNN-HPE approach, a series of simulations are executed. The experimental results reported the betterment of the SSDCNN-HPE system over the recent existing techniques in terms of different measures.

Keywords: Parameter tuning; human pose estimation; deep learning; squirrel search algorithm; activity recognition

1 Introduction

The human pose estimation task that is established for years, and aims to attain the posture of the human body from sensor input. The vision-based approach is frequently utilized for providing a solution with the help of cameras [1]. Also, HPE accomplishes rapid development by using deep learning techniques. The major development includes a well-developed network with rich datasets for feeding networks, further practical exploration of the body, and model greater estimation capability.

Even though there are several researchers have worked in HPE, it still, lacks a study to review the current Deep Learning (DL)-based achievement [2]. As the basic computer vision task, HPE is a significant study area and is employed in different applications like activity or action/recognition, human tracking, action detection, Virtual reality, Movies, and animation, Video surveillance, Human-computer communication, Self-driving, Sports motion analysis, medical assistance, and so on [3,4]. The aim is to automatically discover the human body parts from videos or images. Even though considerable development has been accomplished over the last decade, building a fast, efficient, and accurate scheme for the detection of action in unseen videos remain a challenge because of some difficulties, for example, change in occlusions, camera viewpoint, speed of motion, background, and so on [5].

To resolve the issue of HPE, various techniques were introduced in the survey. The handcrafted features were utilized in earlier studies [6]. This handcrafted feature includes Edgelet and HOG (Histogram of Oriented Gradient) are inadequate in defining the precise location of body parts. On the other hand, the DL-based method can extract adequate features from metadata. This method has produced outstanding results and outperformed the non-deep advanced method [7] by a big margin. Even though the usage of DL in the HPE field is comparatively new, many remarkable studies have been carried out. Implementing human activity recognition with deep learning by replicating human neural networks. But it is recognized that human visual recognition doesn't emphasize the whole scene [8]. Humans sequentially emphasize distinct portions of the scene to extract relevant information. Mostly, the present computer vision algorithm doesn't utilize an attention model and aren't actively investigated in different areas of video or image [9,10]. With the current increase in Deep Neural Networks (DNNs), the attention-based model has been proven to accomplish promising outcomes in several challenging tasks including machine translation, games, and subtitle generation. Several models use Convolutional Neural Network (CNN)-based Long Short Term Memory (LSTM) and show better outcomes in the training sequence. The visual attention mechanism is categorized into soft and hard attention models [11]. This study introduces a squirrel search optimization with a deep convolutional neural network for HPE (SSDCNN-HPE) technique to identify the human pose accurately and efficiently. Primarily, the video frame conversion process is performed and pre-processing takes place via bilateral filtering-based noise removal process. Then, the EfficientNet model is applied to identify the body points of a person with no problem constraints. Besides, the hyperparameter tuning of the EfficientNet model takes place by the use of SSA. In the final stage, the multiclass support vector machine (M-SVM) model is employed for the identification and classification of human poses. To demonstrate the improved outcomes of the SSDCNN-HPE algorithm, a series of simulations are implemented.

The structure of the work is presented: Section 2 provides a detailed literature survey. Section 3 briefly explains Human Activity Recognition (HAR) and its usage in diverse settings. Section 4 analyses the results. At long last Section 5 comes to an end and poses a few outstanding issues for further study.

2 Literature Review

Gao et al. [12] presented a multi-scales fusion architecture-based hourglass network for the pose evaluation that could efficiently attain comprehensive data of distinct resolutions. In the procedure of extracting distinct resolution features, the network continuously complements the higher resolution feature. The entire network is arranged by sub-network. To apply in constrained memory space better, we only utilize a 2-phase stacked network. Omran et al. [13] presented a technique called

Neural Body Fitting (NBF) which incorporates a statistical body method within a CNN. It leveraged robust top-down body model constraint and consistent bottom-up semantic body part segmentation. Yang et al. [14] developed an online technique to learn the pose dynamics that are independent of pose detection in existing fame, and therefore might serve a strong estimation even in difficult scenarios comprising occlusion. It forecasts the equivalent poses in the frame for the tracklet and takes as input the past pose tracklet. Zou et al. [15] proposed a Modulated Graph Convolutional Network (GCN) for 3D HPE. It comprises two major elements: affinity and weight modulations. Weight modulation learns distinct modulation vectors for distinct nodes thus the feature transformation of distinct nodes is disentangled while maintaining a smaller model size. Zhang et al. [16] introduced an effective architecture for HPE with two portions, an effective head, and backbone. By executing a neural structure search technique, the study reduces computation cost with negligible accuracy degradation and modifies the backbone network model for HPE. Choi et al. [17] presented a mobile-friendly method, MobileHumanPose, for real-time three-dimensional HPE from single Red, Green, Blue (RGB) images. This technique comprises a parametric activation function, the skip concatenation stimulated by U-Net, and the adapted MobileNetV2 backbone. Particularly, the skip concatenation infrastructure enhanced performance by propagating rich features with negligible computation power. Zhang et al. [18] proposed AdaFuse, an adoptive multiview fusion model that could improve the feature in occluded view by leveraging visible view. The mainstream of AdaFuse is for defining the point-point correspondence among 2 views that are efficiently resolved by examining the sparsity of heatmap illustration. Zhong et al. [19] present a lower computation-cost deep supervision pyramid architecture named DSPNet. Next, presented a deep supervision pyramid network for improving the multiscale gaining capacity of Multiset Regression Analysis (MSRA) Simple Baseline while not bringing any surge in the number of parameters.

3 The Proposed Model

In this study, a novel squirrel search optimization with a deep convolutional neural network for HPE (SSDCNN-HPE) system has been developed for effectual estimation of human poses. The SSDCNN-HPE technique majorly focused on the detection and classification of human poses. The SSDCNN-HPE technique encompasses a series of processes namely pre-processing, EfficientNet-based feature extraction, SSA-based hyperparameter tuning, and Modified SVM (M-SVM) based classification. Fig. 1 demonstrates the block diagram of the SSDCNN-HPE technique.

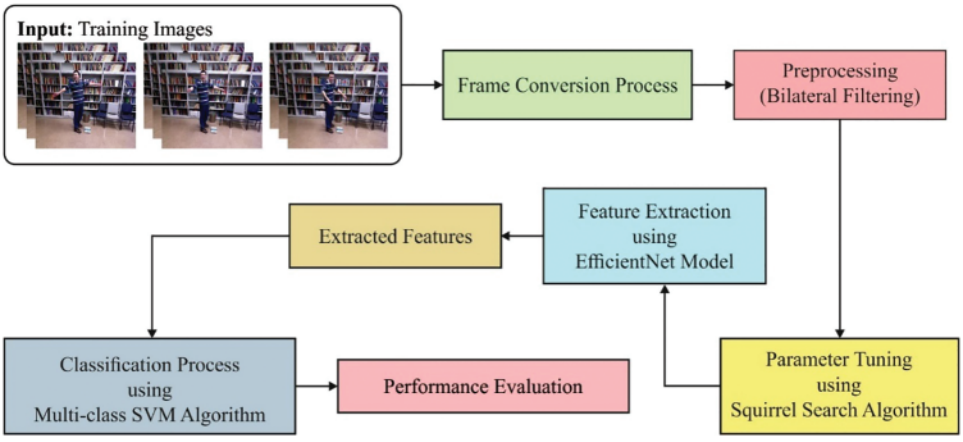

Figure 1: Block diagram of SSDCNN-HPE technique

3.1 Pre-processing

Initially, the input video is converted into frames and the bilateral filtering (BF) technique is applied for noise removal. Consider F represents a multi-channel frame and W indicates a sliding window of finite size $n \times n$. Assume pixels W can be defined by Cartesian Coordinates, denoted as $\mathrm{i} = (i_1,\ i_2) \in Y^2$ location of a pixel $\mathrm{F_i}$ in W where $Y = \{0, 1, \ldots,\ n-1\}$. The BF technique substitutes the middle pixel of every filter window via a weighted average of the nearby color pixels [19,20]. The weight function can be defined for smoothening the regions of identical colors by maintaining the edges together by deeply weighting the pixels which are spatially close and photometrically comparable to the intermediate pixel. Next, the weight $\mathcal{W}\left(\mathrm{F_i},\ \mathrm{F_j}\right)$ corresponds to pixels $\mathrm{F_j}$ based on $\mathrm{F_i}$ is the product of two elements namely spatial and photometrical, as given below.

$$\mathcal{W}\left(\mathrm{F_i},\ \mathrm{F_j}\right) = \mathcal{W}_s\left(\mathrm{F_i}\ ,\ \mathrm{F_j}\right)\mathcal{W}_p\left(\mathrm{F_i},\ \mathrm{F_j}\right) \tag{1}$$

where the spatial element $\mathcal{W}_s\left(\mathrm{F_i}\ ,\ \mathrm{F_j}\right)$ can be represented as follows.

$$\mathcal{W}_s\left(F_\mathrm{i}, F_\mathrm{j}\right) = e^{-\frac{||\mathrm{i}-\mathrm{j}||_2^2}{2\sigma_S^2}} \tag{2}$$

and photometrical element $\mathcal{W}_p\left(\mathrm{F_i},\ \mathrm{F_j}\right)$ can be defined using Eq. (3):

$$\mathcal{W}_p\left(\mathrm{F_i},\ \mathrm{F_j}\right) = e^{-\frac{\Delta E_{Lab}\left(\mathrm{F_i},\mathrm{F_j}\right)^2}{2\sigma_p^2}} \tag{3}$$

where $\Delta E_{Lab} = \left[(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2\right]^{\frac{1}{2}}$ signifies the perceptual color error from the $L^*a^*b^*$ color space, and σ_s,$\sigma_p > 0$. The color vector outcome $\widetilde{Fi}$ of the filter can be determined by normalized weight and is defined as follows.

$$\overline{\mathrm{F_i}} = \frac{\sum_{F_J \in w} \mathcal{W}\left(\mathrm{F_i}, \mathrm{F_j}\right)\mathrm{F_j}}{\sum_{F_J \in w} \mathcal{W}\left(\mathrm{F_i}, \mathrm{F_j}\right)} \tag{4}$$

3.2 Feature Extraction Using EfficientNet Model

During the feature extraction process, the pre-processed frames are passed into the EfficientNet model to derive feature vectors. The feature extracting is performed after the pre-processing stage from the facial expression detection technique. It signifies the data retrieving in the image in a lower dimension space. A DL-based method named deep convolution neural network (DCNN) is deployed that comprises hidden, input, and output layers. The hidden layer is composed of a group of neurons which comprises of fully connected layer, convolution layer, number of ReLU, pooling layer, and normalization layer. A convolutional lay is an initial layer existing in the model for extracting the feature in the image. It comprises a group of filters or kernels that is convolved individually with the input image passing its results to the subsequent layer. The results of the convolutional process produce convolved features in a reduction dimension than the input image. EfficientNet is a family of Recurrent Neural Networks (RNNs) where the EfficientNet model scales effectively in terms of layer width, input resolution, layer depth, and a grouping of this factor [21]. Fig. 2 illustrates the framework of EfficientNet. As well, the EfficientNet model utilizes an efficient and simple compound scaling technique to scale the baseline ConvNet, while preserving efficacy. Generally, the EfficientNet is efficient and more accurate when compared to the present CNN's like ImageNet, AlexNet, MobileNet, and GoogleNet. The major component of the network comprises of MBConv along with excitation optimization and compression. The inverted bottleneck MBConv, the

major component for EfficientNet, comprises layers that extend and compress the channel, therefore direct connection among bottlenecks is utilized for connecting fewer channels when compared to the expanded layer.

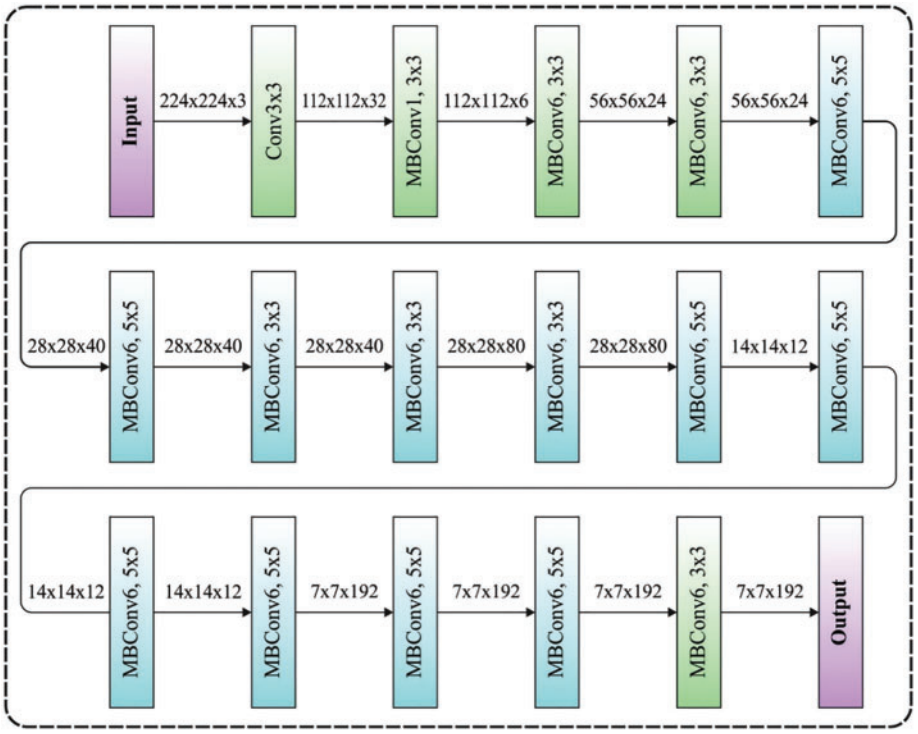

Figure 2: EfficientNet structure

In comparison with the conventional convolution layer, this structure has deeply separable convolution that reduces the calculation by the factor of near k with traditional layers [22]. This technique to determines the scaling factor, whereby α, β, γ are constants defined by a grid search. Likewise, the user-determined factor, ψ, control the presented resource for model scaling, whereas α, β, α γ, β determines that the further resources are assigned for network resolution, width, and depth, correspondingly. An EfficientNet-B0 contains the subsequent stages: considering twice as several available resources, executing the grid search with $\psi = 1$ and the optimum values of α, β, γ.

$$depth: \ d = \alpha^{\psi} \tag{5}$$

$$width: \ w = \beta^{\psi} \tag{6}$$

$$resolution: \ r = \gamma^{\psi} \tag{7}$$

$$\alpha \geq 1, \beta \geq 1, \gamma \geq 1 \tag{8}$$

3.3 Squirrel Search Algorithm Based Hyperparameter Tuning

For optimal hyperparameter tuning of the EfficientNet model, the Squirrel Search Algorithm is applied to it. SSA is presented by Marques et al. [23]. This technique inspires the dynamic foraging performance of the southern squirrel. During the SSA, it can be considered that there is only one hickory nuts tree (HNT), many oak trees (OT), and one of the normal trees (NT) from the forest, with

all the trees maintaining a squirrel. In SSA, Eq. (9) has been utilized for generating a primary place with uniform arbitrary distributions. During this order, the population was separated into 3 groups such as optimum solution, suboptimal solution, and general solution.

$$FS_i = FS_L + U(0,1) \times (FS_U - FS_L) \tag{9}$$

Eq. (9), FS_i implies the place of i^{th} squirrel that is n dimension vector, $U(0,\ 1)$ refers the uniform distribution across in zero and one, FS_U signifies the upper limit of squirrel searches and FS_L represents the lower limit.

Location update formula

During the iterative procedure, the place upgraded equation was separated into 3 kinds, likewise as Eqs. (10)–(12). One squirrel on oak is moved from the direction of HNT. The squirrels on ordinary trees are separated into 2 batches oak and HNT.

Case1. The squirrel on OT heading nearby the HNT.

$$FS_{at}^{t+1} = \begin{cases} FS_{at}^{t} + d_g \times G_c \times \left(FS_{ht}^{t} - FS_{at}^{t}\right) & R_1 \geq P_{dp} \\ Random\ location & \text{otherwise} \end{cases}. \tag{10}$$

Case2. The squirrel on NT heading nearby the OT.

$$FS_{nt}^{t+1} = \begin{cases} FS_{nt}^{t} + d_g \times G_c \times \left(FS_{at}^{t} - FS_{nt}^{t}\right) & R_2 \geq P_{dp} \\ Randomlocation & \text{otherwise} \end{cases}. \tag{11}$$

Case3. The squirrel on NT heading nearby the HNT.

$$FS_{nt}^{t+1} = \begin{cases} FS_{nt}^{t} + d_g \times G_c \times \left(FS_{ht}^{t} - FS_{nt}^{t}\right) & R_3 \geq P_{dp} \\ Randomlocation & \text{otherwise} \end{cases}. \tag{12}$$

In Eqs. (2)–(4), P_{dp} implies the predator occurrence probability. R_1, R_2, and R_3 represents the arbitrary numbers zero and one. G_c represents the gliding constant [24]. d_g refers the gliding distance of squirrels that jump amongst trees. For preventing excessive disturbance in being established as to formula, the normal gliding distance has been scaled for obtaining a perturbation range of d_g across [0.5, 1.11]. In all the iterations, d_g refers to the sliding change that greatly enhances the local search capability of the technique. $FS_{ht}^{t}, FS_{at}^{t}, FS_{nt}^{t}$ stands for the places in which a flying squirrel attains the NT, HNT, and OT correspondingly, with t being the count of iterations.

Seasonal constant

As squirrel mobility alters with seasons, seasonal monitored criteria are presented in this technique. These criteria are abandoned local optimum solutions in techniques to a particular extent. Initially, it can compute the seasonal constant based on Eq. (13).

$$S_c^t = \sqrt{\sum_{k=1}^{d} \left(FS_{at,k}^{t} - FS_{ht,k}\right)^2} \quad (t = 1, 2, 3) \tag{13}$$

Afterward, to check the seasonal monitored criteria $S_c^t \geq S_{\min}$, The computation of $S_{\min}$ is as:

$$S_{\min} = \frac{10E-6}{(365)^{t/(t_{\max}/2.5)}} \tag{14}$$

If the seasonal recognition states are fulfilled, the population of squirrels is transferred, and the equation as:

$$FS_{nt}^{new} = FS_L + Levy\gamma\ (n) \times (FS_U - FS_L) \tag{15}$$

In Eq. (15), $Levy\gamma\ (n)$ is computed as the subsequent equation:

$$Levy\gamma\ (x) = 0.01 \times \frac{r_a \times \sigma}{|r_b|^{1/\beta}} \tag{16}$$

$$\sigma = \left(\frac{\Gamma\ (1+\beta) \times \ \sin\ \left(\frac{\pi \times \beta}{2}\right)}{\Gamma\ \left(\frac{\beta+1}{2}\right) \times \beta \times 2\left(\frac{\beta-1}{2}\right)} \right)^{1/\beta} \tag{17}$$

$$\Gamma\ (x) = (x-1)\,! \tag{18}$$

where r_a, r_b are 2 normally distributed arbitrary numbers from zero and one, and β refers the constant 1.5.

3.4 M-SVM Based Classification

In the final stage, the M-SVM model is employed to determine the appropriate class labels. The M-SVM has been applied to estimate the poses. The execution of M-SVM is complete both experimental and conceptual. It implements the classifier by mapping the input vector to high dimension space and structuring a hyperplane that separates the data from the high dimension space from an optimum method. The M-SVM has been selected as it is a superior count of classes which is classification [25], as related to SVM that is restricted to only 2 kinds of classes. During this procedure, the trained set is utilized for training the M-SVM method and the testing set is utilized for testing the classifier accuracy performance. The testing set was demonstrated in (19) as:

$$X = \{(x_{it} y_i)\}_{i=1}^{l} \ where\ \chi_i \in R^n \ and\ y_i \in \{1_t 2_t 3_t\ .c\} \tag{19}$$

4 Performance Validation

Parameter Settings

Training the model is a CPU-intensive exercise and cannot be performed in a normal server without proper GPU kits. Once the model is trained, it can be saved along with the weight as an h5 file and can be loaded into a client workstation.

Server Configuration:

Processor–Intel i7–5930–6 core, 2.2 GHz. 15 MB Cache

RAM–64 GB DDR4

Hard Disk–500 GB

GPU–NVidia (16 GB V-RAM)

This section examines the experimental validation of the SSDCNN-HPE model on benchmark datasets such as UCF-11 [26], NW-UCLA [27], and Penn Action [28] datasets.

Fig. 3 demonstrates the input images and the corresponding keypoints generated on each frame on the UCF-Sports Action dataset. The first and third row represents the original video frames and the key points generated at each frame are offered in the second and fourth rows.

Figure 3: UCF-Sports Action Dataset (1st and 3rd Row Images/2nd and 4th Row Key Points)

Fig. 4 illustrates the few sample images on the NW-UCLA dataset. The results of the SSDCNN-HPE model are examined and compared with existing techniques. UCF-11 is a Youtube Action data set containing 11 activities like tennis, basketball shooting, walking diving, golf swing, horseback riding, soccer, juggling, volleyball, trampoline, and cycling. The clip has a frame rate of 29.97 FPS (Frames Per Second), and all videos have only one linked activity. It can be utilized 975 videos to train and 625 videos to test. The NW-UCLA data set is the most RGB-D-based dataset. It contains 1494 videos in 10 activity classes: pick up with one hand, pick up with 2 hands, throw away the trash, stand up, walk, sit down, put on, take off, throw, and move. All the activities are executed 1–6 times per learning. The data set is offered with RGB and depth corresponding to expected 3D skeleton sequences. The Penn Action dataset contains 2326 videos containing 15 activities including baseball pitching, bench press, and guitar striking.

The accuracy outcome analysis of the SSDCNN-HPE technique under the UCF-Sports Action dataset is portrayed in Fig. 5. The results demonstrated that the SSDCNN-HPE approach has been able to improve validation accuracy related to training accuracy. It is also observable that the accuracy values get saturated with the count of epochs. The loss outcome analysis of the SSDCNN-HPE technique under the UCF-Sports Action dataset is portrayed in Fig. 6. The figure revealed that the SSDCNN-HPE system has denoted the lower validation loss on the training loss. It is additionally noticed that the loss values get saturated with the count of epochs.

Figure 4: Sample Images (NW-UCLA Dataset)

For showcasing the enhanced performance of the SSDCNN-HPE technique, a comparative accuracy analysis of the UCF Sports Action dataset is made in Table 1 and Fig. 7. From the experimental results, it could be noticed that the SVM and DT techniques have resulted in reduced accuracy values of 0.780 and 0.780 correspondingly. At the same time, the CNN model has tried to showcase slightly enhanced outcomes with an accuracy of 0.830. Followed by, the APAR-MMSHF technique has resulted in a moderately increased accuracy of 0.890. However, the SSDCNN-HPE model has accomplished higher performance with increased accuracy of 0.915.

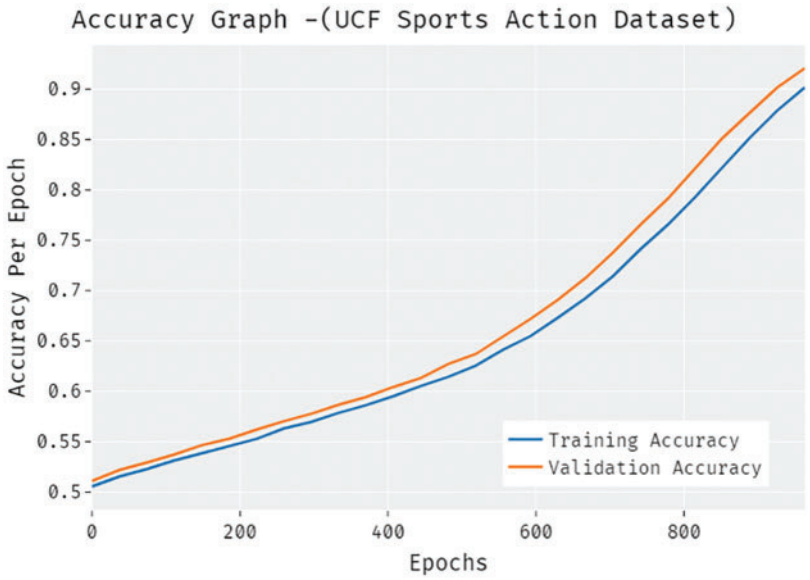

Figure 5: Accuracy analysis of SSDCNN-HPE technique under UCF-Sports Action dataset

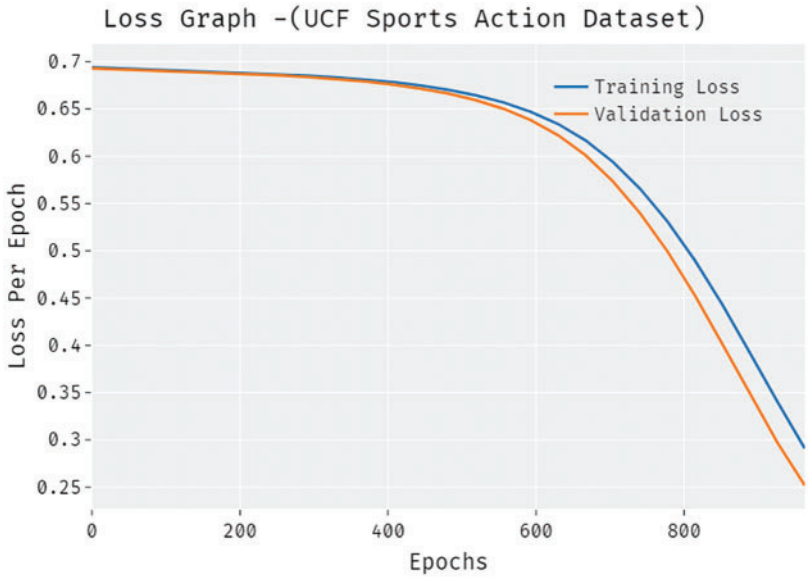

Figure 6: Loss analysis of SSDCNN-HPE technique under UCF-Sports Action dataset

Table 1: Comparative analysis of SSDCNN-HPE technique with recent approaches on UCF Sports Action dataset

Methods	Accuracy
SVM Algorithm	0.780

(Continued)

Table 1: Continued

Methods	Accuracy
DT Algorithm	0.780
CNN Algorithm	0.830
APAR-MMSHF	0.890
SSDCNN-HPE	0.915

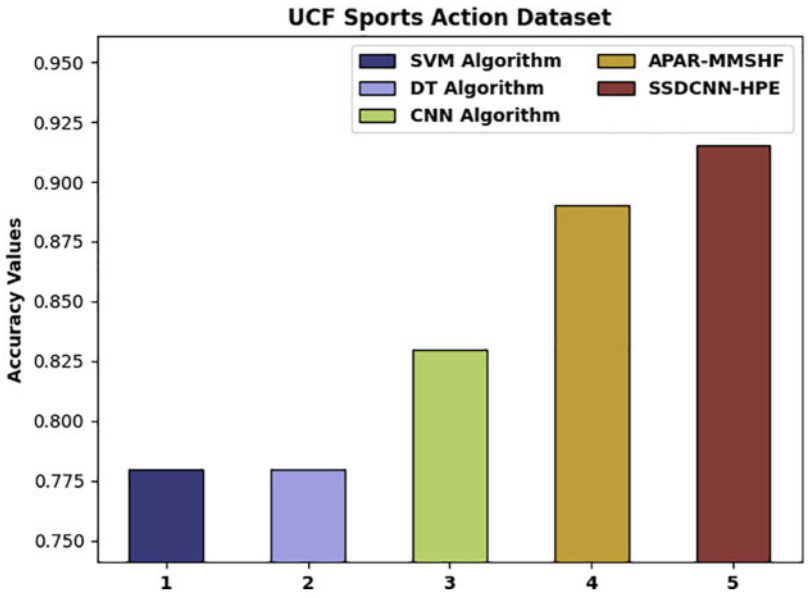

Figure 7: Comparative analysis of SSDCNN-HPE technique on UCF Sports Action dataset

A brief Region of Curve (ROC) examination of the SSDCNN-HPE model with existing models is made on the UCF Sports Action dataset in Fig. 8. The figure portrayed that the SVM approach has obtained ineffective outcomes with a lesser ROC of 94.9908. In line with this, the CNN model has significantly increased performance with an ROC of 95.1723. Next, the DT model has accomplished slightly enhanced outcomes with a ROC of 95.1723. Though the APAR-MMSHF technique has resulted in a reasonable ROC of 96.3978, the presented SSDCNN-HPE model has achieved better results with a maximum ROC of 98.1736.

The accuracy outcome analysis of the SSDCNN-HPE technique under the NW-UCLA dataset is portrayed in Fig. 9. The results demonstrated that the SSDCNN-HPE technique has capable of improved validation accuracy compared to training accuracy. It is also observable that the accuracy values get saturated with the count of epochs. The loss outcome analysis of the SSDCNN-HPE technique under the NW-UCLA dataset is depicted in Fig. 10. The figure revealed that the SSDCNN-HPE approach has denoted the minimum validation loss on the training loss. It is additionally noticed that the loss values get saturated with the count of epochs.

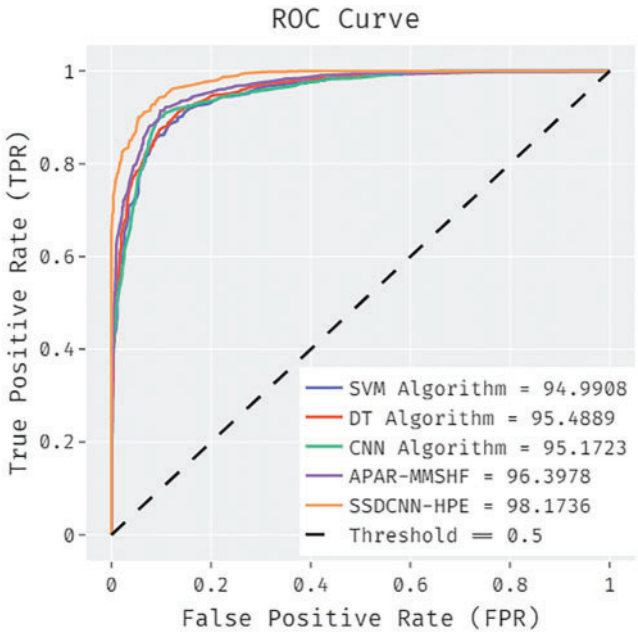

Figure 8: ROC analysis of SSDCNN-HPE technique under UCF-Sports Action dataset

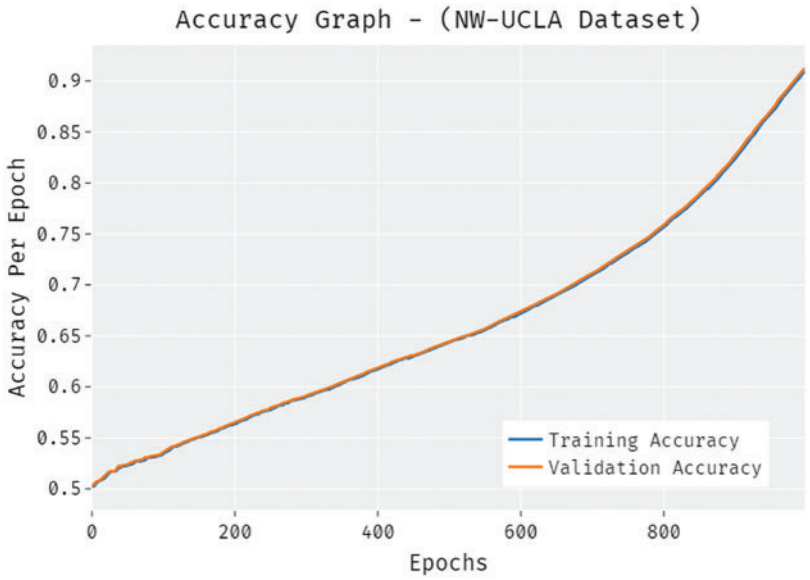

Figure 9: Accuracy analysis of SSDCNN-HPE technique under NW-UCLA dataset

For showcasing the enhanced performance of the SSDCNN-HPE technique, a comparative accuracy analysis of the NW-UCLA dataset is made in Table 2 and Fig. 11 [29,30]. From the experimental outcomes, it could be noticed that the ETESPT-HAR and PCSTA-HAR methods have resulted in lesser accuracy values of 0.750 and 0.850 correspondingly. Besides, the BVA and BEP model has tried to showcase somewhat enhanced outcomes with the accuracy of 0.875 and 0.880 correspondingly. Similarly, the ARC-VAPS methodology has resulted in a moderately increased accuracy of 0.880. At

last, the SSDCNN-HPE technique has accomplished superior performance with enhanced accuracy of 0.912.

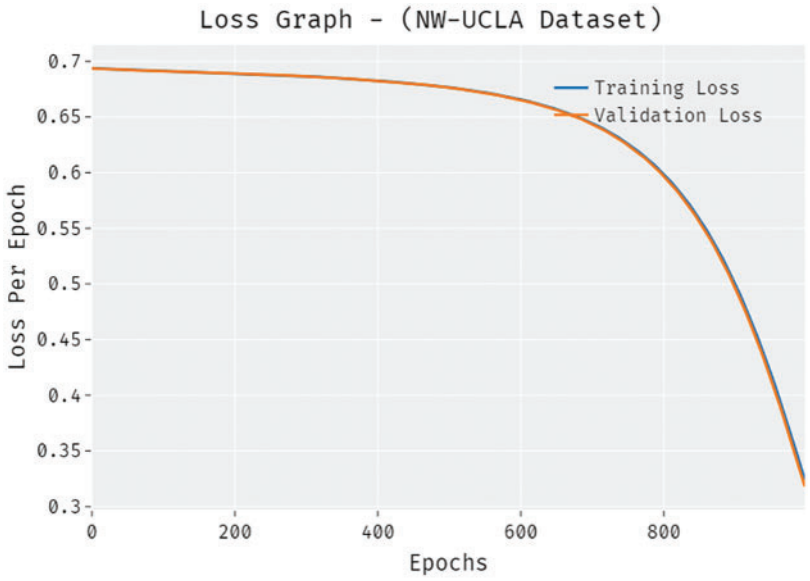

Figure 10: Loss analysis of SSDCNN-HPE technique under NW-UCLA dataset

Table 2: Comparative analysis of SSDCNN-HPE technique with recent approaches on NW-UCLA dataset

Methods	Accuracy
ETESPT-HAR	0.750
PCSTA-HAR	0.850
BVA	0.875
BEP	0.880
ARC-VAPS	0.880
SSDCNN-HPE	0.912

A detailed ROC examination of the SSDCNN-HPE approach with existing techniques is made on the NW-UCLA dataset in Fig. 12. The figure portrayed that the ETESPT-HAR model has obtained ineffective outcomes with a lower ROC of 87.1306. In addition, the PCSTA-HAR model has resulted in somewhat enhanced performance with a ROC of 87.3779. Afterward, the BVA and BEP techniques accomplished slightly enhanced outcomes with the ROC of 87.6185 and 88.0692. Moreover, the ARC-VAPS technique has resulted in a reasonable ROC of 88.6639, the presented SSDCNN-HPE approach has accomplished better results with a superior ROC of 89.1923. The accuracy outcome analysis of the SSDCNN-HPE technique under the Penn action dataset is portrayed in Fig. 13. The results demonstrated that the SSDCNN-HPE system has accomplished improved validation accuracy related to training accuracy. It is also observable that the accuracy values get saturated with the count of epochs. The loss outcome analysis of the SSDCNN-HPE technique under the Penn action dataset is depicted in Fig. 14. The figure exposed that the SSDCNN-HPE methodology has denoted the

decreased validation loss on the training loss. It can be additionally detected that the loss values get saturated with the count of epochs.

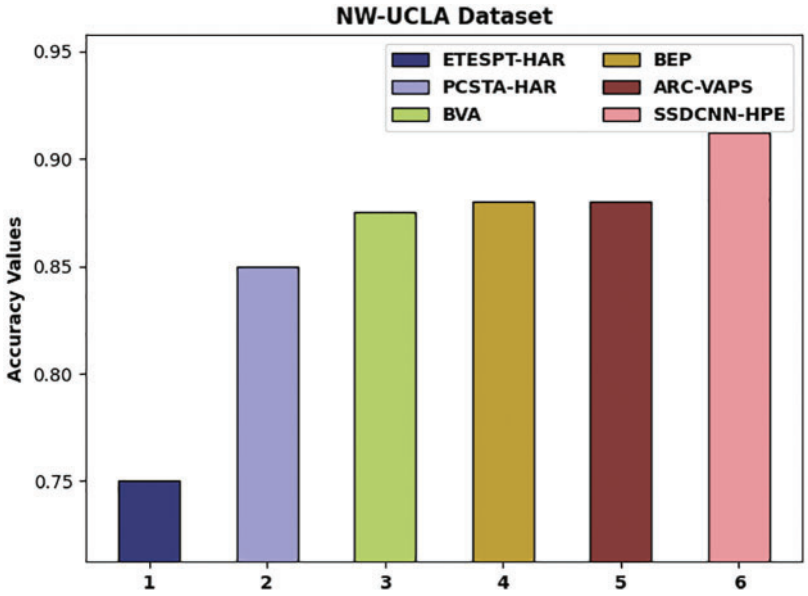

Figure 11: Comparative analysis of SSDCNN-HPE technique on NW-UCLA dataset

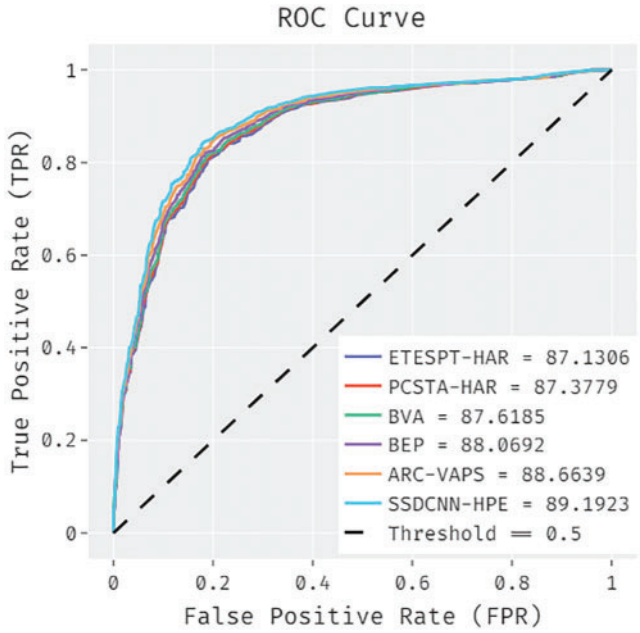

Figure 12: ROC analysis of SSDCNN-HPE technique under NW-UCLA dataset

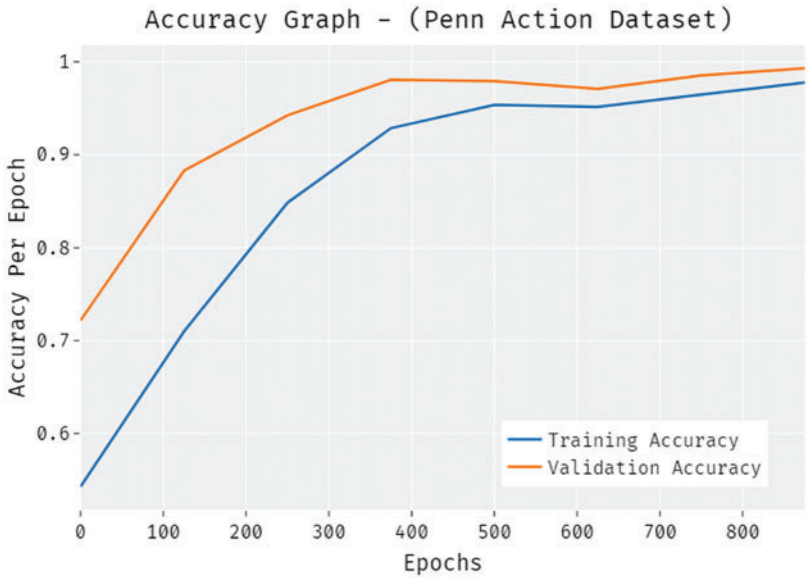

Figure 13: Accuracy analysis of SSDCNN-HPE technique under Penn action dataset

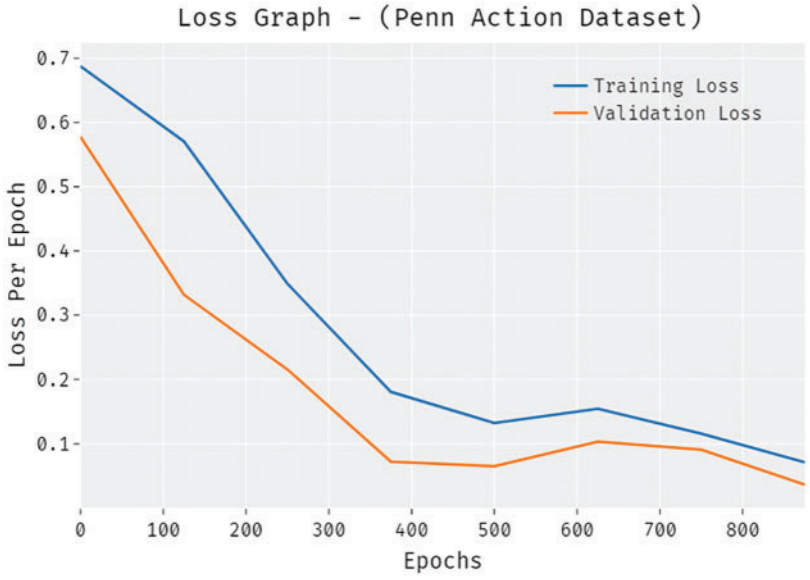

Figure 14: Loss analysis of SSDCNN-HPE technique under Penn action dataset

To demonstrate the enhanced performance of the SSDCNN-HPE model, a comparative accuracy analysis of the Penn action dataset is made in Table 3 and Fig. 15 [31–35]. From the experimental results, it could be noticed that the JAR-PSV and PAAP models have resulted in minimal accuracy values of 0.860 and 0.790 correspondingly. Simultaneously, the BJG-3D Deep Conv. and BEP model have tried to showcase slightly enhanced outcomes with the accuracy of 0.980 and 0.980 correspondingly. Furthermore, the ARC-VAPS technique has resulted in a moderately improved accuracy of 0.990. Finally, the SSDCNN-HPE methodology has accomplished maximum performance with higher accuracy of 0.993.

Table 3: Comparative analysis of SSDCNN-HPE technique with recent approaches on Penn action dataset

Methods	Accuracy
JAR-PSV	0.860
PAAP	0.790
BJG-3D Deep Conv.	0.980
BEP	0.980
ARC-VAPS	0.990
SSDCNN-HPE	0.993

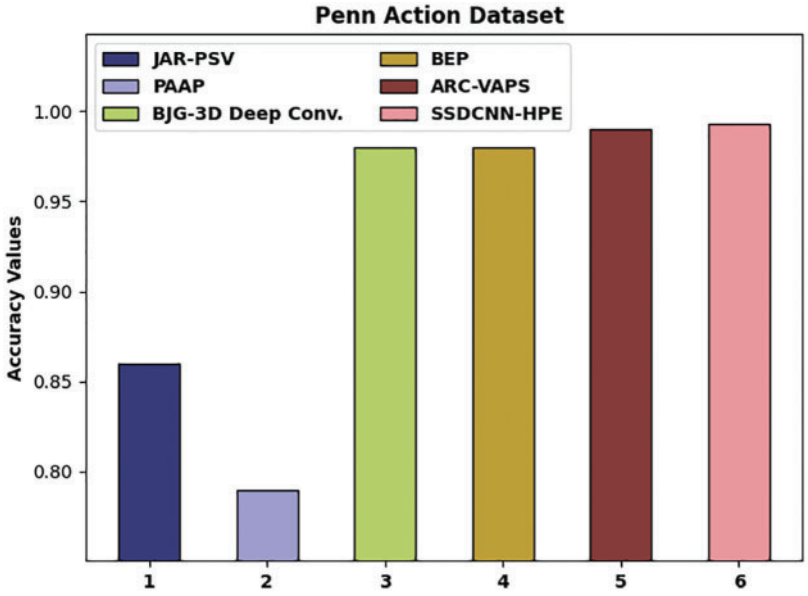

Figure 15: Comparative analysis of SSDCNN-HPE technique on Penn action dataset

A brief ROC examination of the SSDCNN-HPE system with existing algorithms is made on the Penn action dataset in Fig. 16. The figure demonstrated that the JAR-PSV technique has obtained ineffective outcomes with a lower ROC of 92.0425. Likewise, the PAAP model has resulted in somewhat increased performance with an ROC of 93.1237. Next to that, the BJG-3D Deep Conv. and BEP methods have accomplished somewhat improved outcomes with the ROC of 93.8484 and 94.7399. In addition, the ARC-VAPS technique has resulted in a reasonable ROC of 94.3397, the presented SSDCNN-HPE approach has accomplished better results with a higher ROC of 95.1543.

After examining the above-mentioned tables and figures, it can be obvious that the SSDCNN-HPE technique has showcased effective capability in pose estimation. Therefore, the SSDCNN-HPE model can be utilized as an effective approach for HPA.

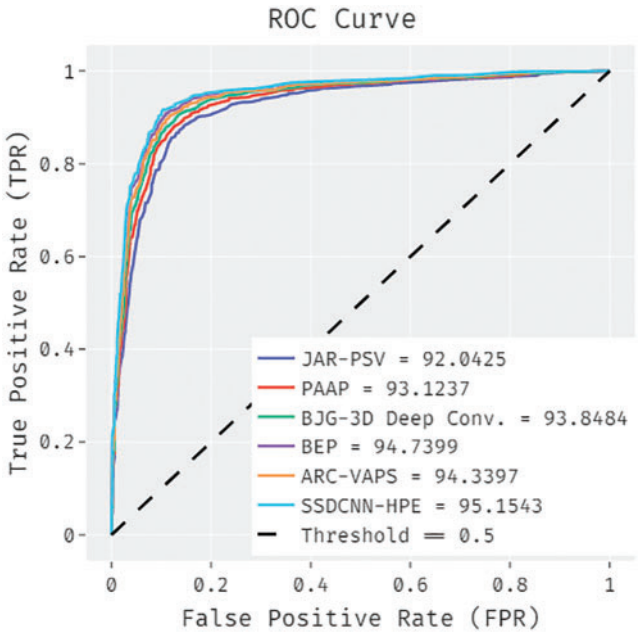

Figure 16: ROC analysis of SSDCNN-HPE technique under Penn action dataset

5 Conclusion

In this study, a novel SSDCNN-HPE approach has been developed for the effectual estimation of human poses. The SSDCNN-HPE technique majorly focused on the detection and classification of human poses. The SSDCNN-HPE technique encompasses a series of processes namely pre-processing, EfficientNet-based feature extraction, SSA-based hyperparameter tuning, and M-SVM-based classification. The utilization of SSA helps to appropriately tune the hyperparameters of the EfficientNet model and it results in improved pose estimation performance. To demonstrate the enhanced outcomes of the SSDCNN-HPE system, a series of simulations are executed. The experimental results reported the betterment of the SSDCNN-HPE algorithm on the recent existing techniques in terms of different measures. Therefore, the SSDCNN-HPE technique was employed as an effectual tool for HPE. In the future, hybrid DL models can be used for the classification process rather than the M-SVM model.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. Martinez, R. Hossain, J. Romero and J. J. Little, "A simple yet effective baseline for 3d human pose estimation," in *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, Venice, Italy, pp. 2659–2668, 2017.

[2] C. Ionescu, F. Li and C. Sminchisescu, "Latent structured models for human pose estimation," in *2011 Int. Conf. on Computer Vision*, Barcelona, Spain, pp. 2220–2227, 2011.

[3] M. Andriluka, L. Pishchulin, P. Gehler and B. Schiele, "2D human pose estimation: New benchmark and state of the art analysis," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Columbus, OH, USA, pp. 3686–3693, 2014.

[4] A. A. Nithyaand and C. Lakshmi, "Enhancing iris recognition framework using feature selection and BPNN," *Cluster Computing*, vol. 22, no. 5, pp. 12363–12372, 2019.
[5] A. Toshev and C. Szegedy, "DeepPose: Human pose estimation via deep neural networks," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Columbus, OH, USA, pp. 1653–1660, 2014.
[6] B. Xiao, H. Wu and Y. Wei, "Simple baselines for human pose estimation and tracking," in *Proc. European Conf. on Computer Vision (ECCV)*, Munich, Germany, pp. 466–481, 2018.
[7] T. L. Munea, Y. Z. Jembre, H. T. Weldegebriel, L. Chen, C. Huang *et al.,* "The progress of human pose estimation: A survey and taxonomy of models applied in 2D human pose estimation," *IEEE Access*, vol. 8, pp. 133330–133348, 2020.
[8] W. Ouyang, X. Chu and X. Wang, "Multi-source deep learning for human pose estimation," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Columbus, OH, USA, pp. 2337–2344, 2014.
[9] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook *et al.,* "Efficient human pose estimation from single depth images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 12, pp. 2821–2840, 2013.
[10] F. Zhang, X. Zhu, H. Dai, M. Ye and C. Zhu, "Distribution-aware coordinate representation for human pose estimation," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, pp. 7091–7100, 2020.
[11] D. C. Luvizon, D. Picard and H. Tabia, "Multi-task deep learning for real-time 3D human pose estimation and action recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 8, pp. 2752–2764, 2020.
[12] B. Gao, K. Ma, H. Bi, L. Wang and C. Wu, "Learning high resolution reservation for human pose estimation," *Multimedia Tools and Applications*, vol. 80, no. 19, pp. 29251–29265, 2021.
[13] M. Omran, C. Lassner, G. Pons-Moll, P. Gehler and B. Schiele, "Neural body fitting: Unifying deep learning and model based human pose and shape estimation," in *Proc. Int. Conf. on 3D Vision (3DV)*, Verona, Italy, pp. 484–494, 2018.
[14] Y. Yang, Z. Ren, H. Li, C. Zhou, X. Wang *et al.,* "Learning dynamics via graph neural networks for human pose estimation and tracking," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, pp. 8070–8080, 2021.
[15] Z. Zou and W. Tang, "Modulated graph convolutional network for 3D human pose estimation," in *Proc. IEEE/CVF Int. Conf. on Computer Vision (ICCV)*, Montreal, QC, Canada, pp. 11457–11467, 2021.
[16] W. Zhang, J. Fang, X. Wang and W. Liu, "Efficientpose: Efficient human pose estimation with neural architecture search," *Computational Visual Media*, vol. 7, no. 3, pp. 335–347, 2021.
[17] S. Choi, S. Choi and C. Kim, "Mobilehumanpose: Toward real-time 3D human pose estimation in mobile devices," in *Proc. IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW)*, Nashville, TN, USA, pp. 2328–2338, 2021.
[18] Z. Zhang, C. Wang, W. Qiu, W. Qin and W. Zeng, "Adafuse: Adaptive multiview fusion for accurate human pose estimation in the wild," *International Journal of Computer Vision*, vol. 129, no. 3, pp. 703–718, 2021.
[19] F. Zhong, M. Li, K. Zhang, J. Hu and L. Liu, "DSPNet: A low computational-cost network for human pose estimation," *Neurocomputing*, vol. 423, pp. 327–335, 2021.
[20] K. Ishwarya and A. Alice Nithya, "Performance-enhanced real-time lifestyle tracking model based on human activity recognition (PERT-HAR) model through smartphones," *The Journal of Supercomputing*, vol. 78, no. 4, pp. 5241–5268, 2022.
[21] H. Chang and W. C. Chu, "Double bilateral filtering for image noise removal," in *2009 WRI World Congress on Computer Science and Information Engineering*, Los Angeles, CA, USA, pp. 451–455, 2009.
[22] C. Chen, S. Chandra, Y. Hanand and H. Seo, "Deep learning-based thermal image analysis for pavement defect detection and classification considering complex pavement conditions," *Remote Sensing*, vol. 14, no. 1, pp. 106–123, 2022.
[23] G. Marques, D. Agarwal and I. D. L. T. Díez, "Automated medical diagnosis of COVID-19 through EfficientNet convolutional neural network," *Applied Soft Computing*, vol. 96, pp. 106691–106712, 2020.

[24] M. Jain, V. Singh, and A. Rani, "A novel natureinspired algorithm for optimization: Squirrel search algorithm," *Swarm and Evolutionary Computation*, vol. 44, pp. 148–175, 2019.
[25] K. Panimalar, S. Kanmani, V. Nithya, A. Subalakshmiand and S. Vishvetha, "Data aggregation using squirrel search algorithm in wireless sensor networks," *International Journal of Scientific Research in Science and Technology*, vol. 9, no. 1, pp. 628–639, 2021.
[26] S. Ibrahim, N. A. Zulkifli, N. Sabri, A. A. Shari and M. R. M. Noordin, "Rice grain classification using multi-class support vector machine (SVM)," *IAES International Journal of Artificial Intelligence*, vol. 8, no. 3, pp. 215–232, 2019.
[27] M. D. Rodriguez, J. Ahmed and M. Shah, "Action MACH: A spatio-temporal maximum average correlation height filter for action recognition," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, AK, USA, pp. 24–26, 2008.
[28] J. Zhang, W. Li, P. O. Ogunbona, P. Wang and C. Tang, "RGB-D-based action recognition datasets: A survey," *Pattern Recognition*, vol. 60, pp. 86–105, 2016.
[29] W. Zhang, M. Zhu and K. G. Derpanis, "From actemes to action: A strongly-supervised representation for detailed action understanding," in *Proc. IEEE Int. Conf. on Computer Vision*, Sydney, Australia, pp. 2248–2255, 2013.
[30] S. Song, C. Lan, J. Xing, W. Zeng and J. Liu, "Spatio-temporal attention-based LSTM networks for 3D action recognition and detection," *IEEE Transactions on Image Processing*, vol. 27, no. 7, pp. 3459–3471, 2018.
[31] L. Lu, H. Di, Y. Lu, L. Zhang and S. Wang, "Spatio-temporal attention mechanisms based model for collective activity recognition," *Signal Processing: Image IEEE Transactions on Image processingCommunication*, vol. 74, pp. 162–174, 2019.
[32] B. XiaohanNie, C. Xiong and S. C. Zhu, "Joint action recognition and pose estimation from video," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, Boston, MA, USA, pp. 1293–1301, 2015.
[33] U. Iqbal, M. Garbade and J. Gall, "Pose for action-action for pose," in *Proc. 12th IEEE Int. Conf. on Automatic Face & Gesture Recognition (FG 2017)*, Washington, DC, USA, pp. 438–445, 2017.
[34] C. Cao, Y. Zhang, C. Zhang and H. Lu, "Body joint guided 3-D deep convolutional descriptors for action recognition," *IEEE Transactions on Cybernetics*, vol. 48, no. 3, pp. 1095–1108, 2018.
[35] J. Kimand and D. Lee, "Activity recognition with combination of deeply learned visual attention and pose estimation," *Applied Sciences*, vol. 11, no. 9, pp. 4153–4172, 2021.