Received: 1 July 2022 | Revised: 3 October 2022 | Accepted: 7 October 2022

IET Image Processing

DOI: 10.1049/ipr2.12667

ORIGINAL RESEARCH

Fall detection based on OpenPose and MobileNetV2 network

Mengqi Gao | **Jiangjiao Li** | **Dazheng Zhou** | **Yumin Zhi** | **Mingliang Zhang** | **Bin Li**

School of Mathematics and Statistics, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China

Correspondence
Bin Li, School of Mathematics and Statistics, Qilu University of Technology (Shandong Academy of Sciences), Jinan, China.
Email: ribbenlee@126.com

Funding information
Natural Science Foundation of Shandong Province, Grant/Award Number: ZR2020MF097; National Natural Science Foundation of China, Grant/Award Number: 61973185; Colleges and Universities Twenty Terms Foundation of Jinan City, Grant/Award Number: 2021GXRC100; Development Plan of Young Innovation Team in Colleges and Universities of Shandong Province, Grant/Award Number: 2019KJN011

Abstract
The proposed fall detection approach is aimed at building a support system for the elders. In this work, a method based on human pose estimation and lightweight neural network is used to detect falls. First, the OpenPose is used to extract human keypoints and label them in the images. After that, the modified MobileNetV2 network is used to detect falls by integrating both human keypoint information and pose information in the original images. The above operation can use the original image information to correct the deviation in the keypoint labeling process. Through experiments, the accuracy of the proposed method is 98.6% and 99.75% on the UR and Le2i datasets, which is higher than the listed comparison methods.

1 | INTRODUCTION

A series of injuries from accidental falls are the second leading cause of accidental death among older adults. Accidental fall detection in the elderly living alone can reduce the risk of death and injury [1]. In recent years, researches on fall detection are mainly divided into three types, which are based on wearable device, environmental sensor and computer vision, respectively.

Fall detection can be abstracted as a classification problem, and the proposal of convolution neural network (CNN) architecture such as Alex-net [2], VGG [3], Google-net [4], ResNet [5] and PCAnet [6] have laid a certain foundation for subsequent researchers to improve the performance of their frameworks. The current researches on fall detection based on computer vision algorithms are mainly divided into two ideas in terms of feature extraction: (1) Taking the change of human body contour as the input features. Head position, body aspect ratio parameters and Motion History Image (MHI) were used to represent human posture information for detecting falls [7]. What's more, the authors used body proportion, acceleration and deflection as key features of the human body to detect falls [8]. And the fall motion vector modeling was used for falling detection [9]. (2) Using a deep learning algorithm to extract the spatiotemporal features of the detected target. Chen extracted the skeleton information of the human body by OpenPose and the falls were identified through three critical parameters [10]. The proposed method [11] utilized 3D CNN to extract spatio-temporal information from videos and images to detect stumbles. In short, the features extracted based on the deep learning could make the detection algorithm more robust and generalized.

There are many researches on feature extraction using deep learning algorithms. For example, researchers detected the falls after processing the acquired depth images [12], but the depth image resolution decreased as the distance of the target from the camera increased, and eventually it was difficult to complete background subtraction and segmentation of the depth images due to the low resolution. A method [13] combined multi-sensor, YOLOV3 and Lite Flow Net algorithms to detect falls, but the application conditions of the Lite Flow Net algorithm had certain limitations of light stability and small motion. At present, human pose estimation is widely used in fall detection. The authors learned from the idea of single-stage target detection and proposed PPN (Pose Proposal Network) algorithm

[14] to detect human pose. The CMU team proposes the OpenPose algorithm [15] based on convolutional neural networks and supervised learning. In the literature [16], the authors detected falls based on YOLOv5s and combined with lightweight OpenPose. In [17], the authors applied the AlphaPose algorithm to extract human key points in video images as features to detect falls. The above examples further confirm the necessity of using CNN to extract features.

In the fall detection application scenario, while ensuring accuracy, the algorithm with a fast operation speed is more applicable, so we adopt the MobileNetV2 [18] with fewer parameters as the classification algorithm. The current keypoint detection algorithms are divided into two categories: Top-Down and Bottom-Up detection algorithms. The Bottom-Up detection algorithm detects key points firstly and then associates the key points with a human body, which is faster than Top-Down algorithms. Therefore, considering both the detection speed and accuracy, this paper proposes a new fall detection algorithm based on the Bottom-Up algorithm OpenPose and MobileNetV2.

At present, there are also studies that implement the fall detection algorithm in the edge computing platform and integrate it into the smart home environment [19–21]. The method proposed in [22] deploys the network model to the edge computing platform, reports fall events through WiFi, and performs fall detection while protecting user privacy. The article [23] proposes a PEFDM method based on human gesture recognition and uses an edge computing architecture to detect falls, aiming to eliminate the privacy problem brought by pictures. Embedding algorithms into smart home environments has gradually become a trend.

In this article, the keypoints and torso information of the human body are marked based on retaining the original image information and the feature enhancement operation is performed, then the deep learning algorithm of MobileNetV2 is used for feature extraction and finally detected falls. In this way, the problem of inaccurate detection results caused by the deviation of key point labeling and only using RGB pictures with inconspicuous features can be avoided at the same time. The main contributions are as follows:

(1) Keypoints are marked using openpose, and the marked images are used as input to the classification network for detecting falls.
(2) We propose a modified MobileNetV2 with a fully connected layer and a softmax function to preserve more information (to avoid too much information loss due to rapid dimensionality drop) for the fall detection task, in order to adapt to the fall detection task.
(3) To improve the accuracy of labeling human keypoints in dark environments, we highlighted the column of overly dark frames in the UR dataset.

The remainder of this paper is organized as follows. Section 1 introduces the current state of the field of study. Section 2 presents the proposed fall detection algorithm. Section 3 shows the experimental results of our proposal on the publicly available datasets. Section 4 summarizes the main conclusions.

2 | METHOD

In this section, we introduce the proposed fall detection algorithm. Since OpenPose can quickly mark human key point information and MobileNetV2 has a small parameter volume without reducing the accuracy of the algorithm, we conduct investigation based on the advantages of the two algorithms and propose a new fall detection algorithm. The proposed algorithm contains two parts: features enhancement and fall detection.

2.1 | Features enhancement

In the existing researches, many researchers have used OpenPose to extract human keypoints to represent human pose features for fall detection. However, due to the influence of factors such as the original light in the video image data, there may be some bias in the key point extraction, so the accuracy of the results may be affected for the algorithm that detects falls using only key point information. In order to solve the above problems and improve the effectiveness of the feature extraction stage, the OpenPose is used to extract human key point information, which can make the key point information annotated in the original image and then the annotated images are used as the input of the MobileNetV2 for detecting falls. Through this process, when the key point extraction is biased, the pose features of the original image can also be used as a reference in the subsequent classification detection stage, avoiding the detection error caused by relying only on the biased key point to detect falls.

Using the method of OpenPose in [10], the key point information is extracted from the input image frame column, and the key points are labeled in the original image, as shown in Figure 1. First, the preprocessed images are inputted into the first 10 layers of VGG-19 to extract features. Second, the position confidence map and part affinity fields domain of the limbs are predicted and then, the network model parameters according to the loss function, at the same time associate the body parts with the human body are optimized, and finally the image with the key points of the human body is marked in the original image.

We start from the torso to find the body part points and connect to the human body. For example, the two ends of the forearm must be the shoulders and the elbows. First, search all the connection graphs of the shoulders and elbows. Because the information of the PAFs calculated by the previous network is used as the support, so we can quickly find the big arms that belong to the same person. Then, we separately find other torsos, such as the forearm, calf etc., and combine these torsos to form the human frame. In order to distinguish it from the human body in the picture, we use colorful line segments and solid dots to mark the human torso. The discussion on whether marker color and size affect the detection results is analyzed in Section 3.3.

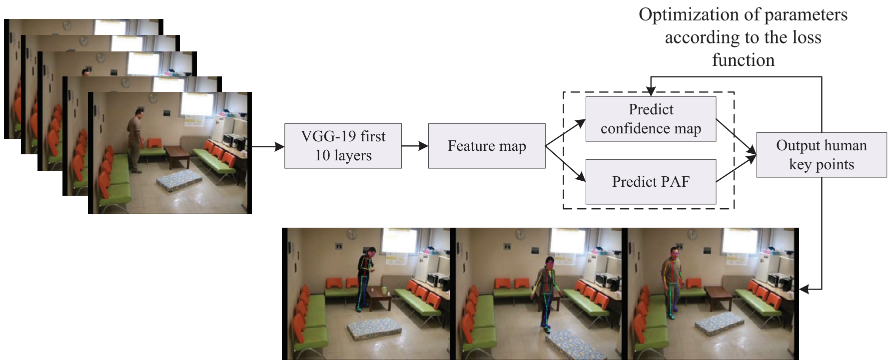

FIGURE 1 Flowchart of marking key points in the human body using Openpose

2.2 | Fall detection

This section focuses on the improved method of fall detection based on the lightweight neural networks of MobileNetV2, which can achieve better detection accuracy while keeping the number of parameters small, so this paper fine-tunes the framework based on MobileNetV2 and uses it for the fall classification detection task. The details are described in the article [18], and here we only describe our modifications to the framework and how they can be applied in the fall to detection scenario.

In response to the backbone selection problem of the proposed algorithm, we conducted a simple detection experiment using the original image on the fall dataset with MobileNetV2, Efficientnet, and EfficientnetV2. The detection accuracy of the above three algorithms on the Le2i dataset was 98.5%, 93.5%, and 94.92%, and the detection accuracy on the UR was 96.3%, 95.93%, and 96%. Although the performance of the Efficientnet series of networks is powerful, in the fall detection task, the experimental results show that the fall detection accuracy of MobileNetV2 is higher, so we use MobileNetV2 as the backbone network of the proposed algorithm. The specific experiments and analysis are shown in Section 3.2.

The network framework of MobileNetV2 outputs directly through the fully connected layer and excludes the classifier. In order to meet the needs of the classification task in fall detection, we added a fully connected layer (to avoid information loss caused by rapid dimensionality reduction) and a softmax classifier behind the original framework. In addition, we found that MobileNetV2 lacks an attention mechanism, so we added a CBAM (Convolutional Block Attention Module)[24] attention mechanism at the beginning of the network to improve the learning ability of the shallow network for essential targets. The modified network framework is shown in Figure 2.

In Figure 2, the network starts with a convolutional layer, followed by CBAM attention framework and seven bottlenecks stacked with blocks of different times. Based on the original framework of MobileNetV2, to slow down the drop speed of the feature dimension, we add a fully connected layer at the network end of the MobileNetV2 and finally use the softmax function to calculate the probability that the picture frame belongs to the fall or not.

The cross-entropy loss function is adopted to estimate the model loss, while the Adam algorithm is used to optimize the parameters for improving the model generalization performance. The cross-entropy loss function as shown in equation 1.

$$Loss = -\frac{1}{m}\sum_{q}\left[y\_t \cdot \ln(y\_p) + (1 - y\_t)\ln(1 - y\_p)\right], \quad (1)$$

where m is the number of samples, y_t is the real category, y_p is the prediction category, q is the sample.

The features are extracted using deep separable convolution and then downscaled by average pooling and fully connected layers. Finally the behavior in the picture frame is output by the softmax function as fall/not fall and its probability of belonging to that category.

2.3 | Overall framework

The overall flow of the proposed algorithm in this paper is shown in Figure 3. Step1: input the preprocessed image into the OpenPose network. Step2: extract features. Step3: predict the position confidence map and part affinity fields domain of the limbs, and optimize the network model parameters according to the loss function. Step4: associate the body parts with the human body, and then mark the image with the key points of the human body. These four steps use the Openpose algorithm to label the human key points in the picture. Step5: Input images with key point information into MobileNetV2, then extract features using deep convolution module. Step6: Use the Adam algorithm to optimize model parameters according to the loss function and finally detect falls.

3 | EXPERIMENTS

3.1 | Experimental conditions

In terms of hardware, the experiment runs on a server with Intel(R) Core (TM) i7-9700 CPU @ 3.00GHz 3.00 GHz processor and NVIDIA GeForce RTX 2080. In terms of software,

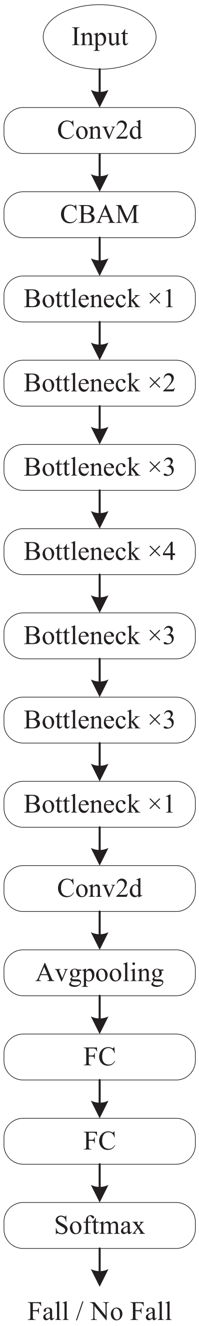

FIGURE 2 A process framework diagram of the feature extraction and classification part of the proposed algorithm

the operating system is Windows10, the development language is Python3.7 and the deep learning library is Tensorflow2.2.0. The algorithm proposed in this paper has experimented on the public datasets Le2i [25] and UR [26]. The Le2i dataset contains a total of 191 human motion videos in 5 different environments with an image resolution of 320 × 240. The UR dataset contains 70 (30 falls + 40 activities of daily living) sequences, and each video stream is stored in a separate compressed file as a sequence of PNG (Portable Network Graphic Format) images. The resolution is 640 × 480.

3.2 | Selection of backbone networks

In response to the backbone selection problem of the proposed algorithm, we conducted a simple detection experiment using the original image on the fall dataset with MobileNetV2, Efficientnet, and EfficientnetV2. In Figure 4, the detection accuracy of the above three algorithms on the Le2i dataset was 98.5%, 93.5%, and 94.92%, respectively, and in Figure 5 the detection accuracy on the UR was 96.3%, 95.93%, and 96%, respectively. Although the performance of the Efficientnet series of networks is powerful, in the fall detection task, the experimental results show that the fall detection accuracy of MobileNetV2 is higher, so we use MobileNetV2 as the backbone network of the proposed algorithm.

3.3 | Experimental of labeling key points

In the process of labeling key points, in order to clearly distinguish them from the human body and the background, we use colored labels instead of black and white lines, and the colors are randomly set. And we use solid dots to label the joint points. To observe whether marker color and size used for marking has an effect on the detection results, we conducted 20 trials on the Le2i and UR datasets, respectively, and visualized the accuracy in the form of a line graph.

Figure 6 is the accuracy change curve of 20 experiments on the Le2i dataset, and Figure 5 is the accuracy curve of 20 experiments on the UR dataset. It can be seen from Figure 4, that the accuracy of the 20 experiments does not change by more than 0.0003, and the rounding accuracy remains around 98.6%. As in Figure 7, the accuracy of the 20 experiments does not change by more than 0.0002, and the rounding accuracy remains around 99.75%. This shows that although the color of the line changes randomly when marking key points, The color and size of solid dots and lines used to mark the human body will not affect the detection results.

3.4 | Ablation experiments

To demonstrate that the improvements made by the proposed algorithms were practical, we performed a series of ablation experiments on the proposed algorithm in the Le2i and UR datasets, respectively. The experimental results of the algorithm on the Le2i dataset are shown in Figure 8. We first verify that the annotated image will have a favorable effect on the inspection results. After we used Openpose to annotate the keypoints of the image, the detection accuracy of the algorithm increased from 95.8% to 97.78%. This shows that the annotated picture can enhance the posture characteristics of the human body and improve the accuracy of the algorithm detection. Then, to strengthen the focus on crucial information in the shallow part of the network, we added the CBAM attention mechanism at the beginning of MobilenetV2's network. After adding attention, the algorithm's detection accuracy increased from 97.78% to

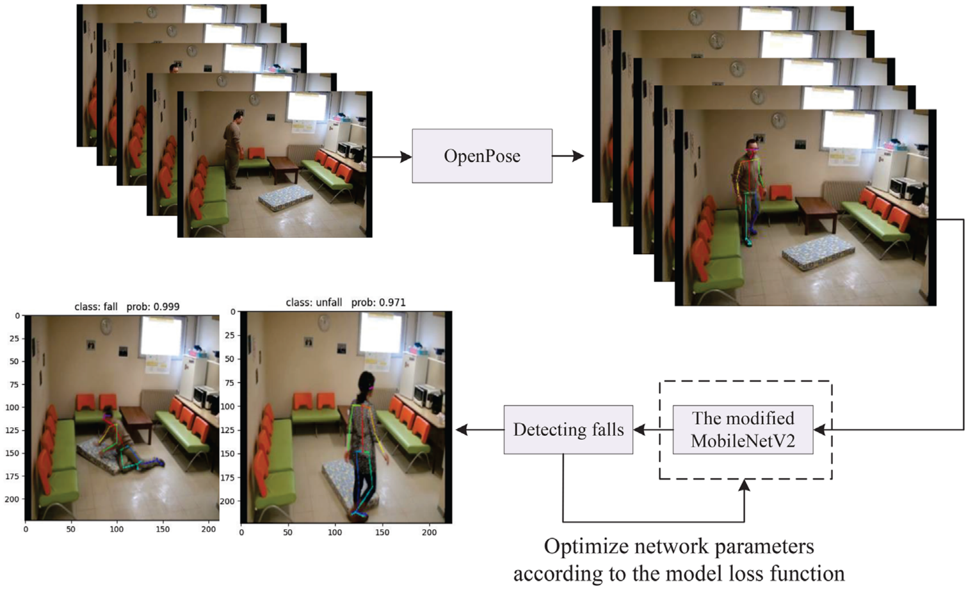

FIGURE 3 Overall flow of the proposed algorithm

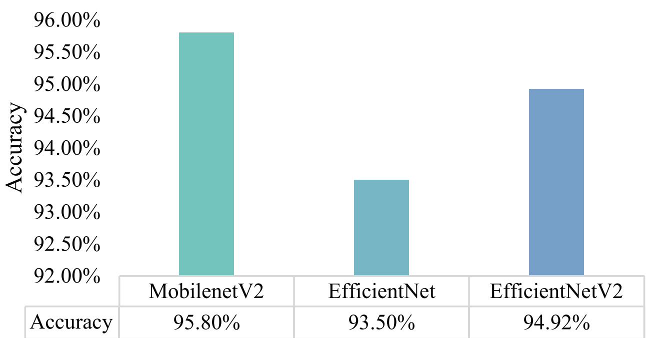

FIGURE 4 Detection results of three algorithms on Le2i

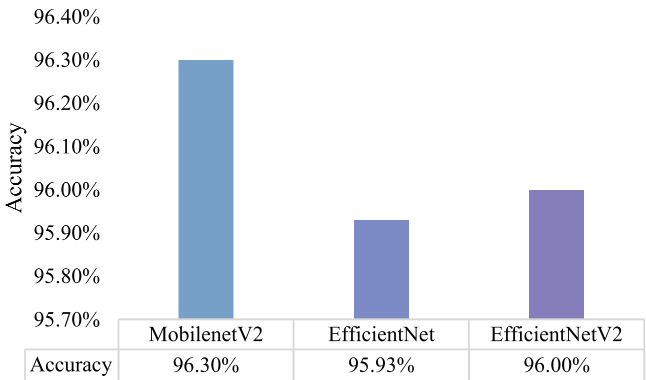

FIGURE 5 Detection results of three algorithms on UR

98.2%. Adding attention mechanisms increases the algorithm's focus on critical features, thereby improving algorithm performance. When training a network, there are rare situations where the accuracy of network training does not change for a long time. After many attempts, we found that the network was out-

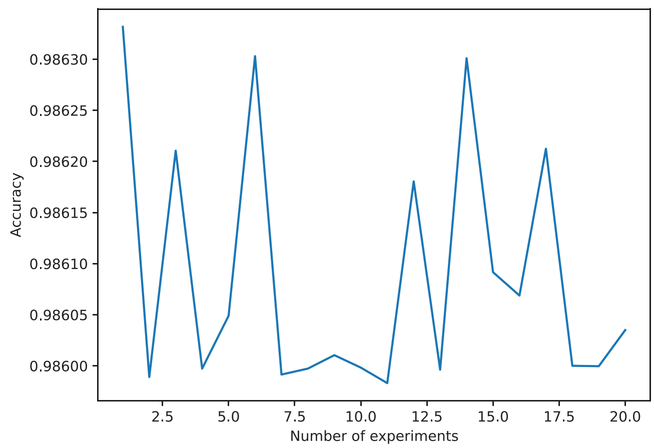

FIGURE 6 Conducted 20 experiments on the Le2i dataset, line graph of the change in accuracy

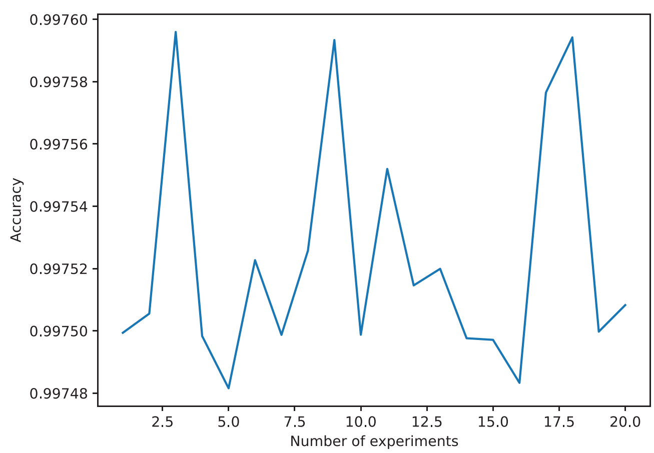

FIGURE 7 Conducted 20 experiments on the UR dataset, line graph of the change in accuracy

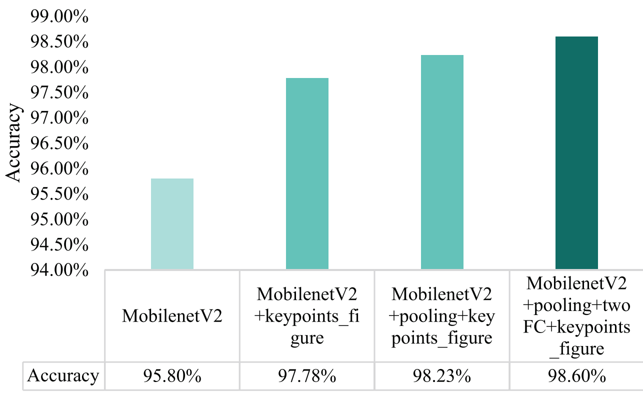

FIGURE 8 Ablation experiment of the proposed algorithm on the Le2i

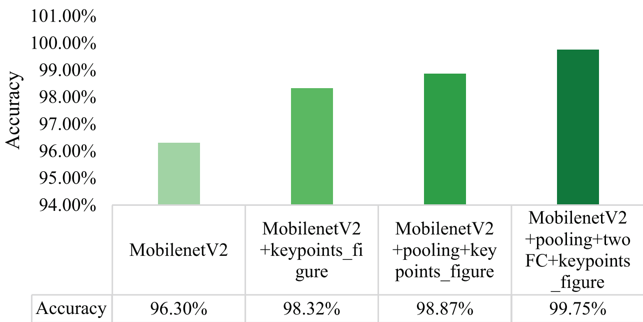

FIGURE 9 Ablation experiment of the proposed algorithm on the UR

put only through a fully connected layer, and the rapid decline in dimension led to excessive loss of data features. To avoid too much information loss caused by the sudden decrease in data dimensions, we modified the classification part of the network and added a fully connected layer. And there was no training stagnation problem after the modification, and the detection accuracy of the network increased from 98.2% to 98.6%.

The experimental results of the algorithm on the UR dataset are shown in Figure 9. First, using Openpose to annotate the keypoints of the image, the results show that the detection accuracy of the algorithm increased from 96.3% to 98.32%. Then, add the cam attention mechanism at the beginning of the MobilenetV2 network. After adding the attention mechanism, the detection accuracy of the algorithm increased from 98.32% to 98.87%. Finally, the fully connected layer is added, and the detection accuracy of the algorithm reaches 99.75%. All of the above proves that our improvement has an effect.

TABLE 1 Corresponding positions of key points (*L* stands for left; *R* stands for right)

Number	Key point	Number	Key point	Number	Key point
0	Noe	9	Hip(R)	17	Ear(R)
1	Nek	10	Knee(R)	18	Ear(L)
2	Shoulder(R)	11	Ankle(R)	19	BigToe(L)
3	Reibow	12	Hip(L)	20	SmallToe(L)
4	Wrist(R)	13	Knee(L)	21	Heel(L)
5	Shoulder(L)	14	Ankle(L)	22	BigToe(R)
6	Elbow(L)	15	Eye(R)	23	SmallToe(R)
7	Wrist(L)	16	Eye(L)	24	Heel(R)
8	MidHip				

3.5 | Experimental on the Le2i dataset

First, the dataset is processed, and the videos in the Le2i dataset are saved as picture data. According to every 0.2 s, the video data is converted into a picture, and a frame of a picture is saved every 5 frames, and some processed images are shown in Figure 10. The training data and the testing data are divided according to 7:3, and then the OpenPose algorithm is used to extract 25 key points of the human body according to the corresponding positions of the key points in Table 1, and next the key points are marked in the original image, as shown in Figure 11. The pictures of point and torso information can make the human features more distinct from the background and the posture information more clearly without increasing the complexity of the pictures. This operation is equivalent to enhancing the features of the original data so that the detection result is more accurate than only using the RGB original images.

The output of the OpenPose network is used as the input of MobileNetV2 to detect falls. Figure 12 shows the detection results of the algorithm on the Le2i dataset. The proposed algorithm can more accurately detect whether there is a fall behavior in each frame of pictures in the test data set. To verify the effect of OpenPose for feature enhancement, we directly feed RGB images without OpenPose processing to MobileNetV2 to detect falls and compare the result curves of the proposed method. At the same time, a comparison with the other three existing methods is also carried out to demonstrate the effectiveness of the proposed method.

The accuracy and convergence curves of the comparison experiment are shown in Figures 13 and 14. It can be seen from Figure 13 that the proposed method has excellent accuracy

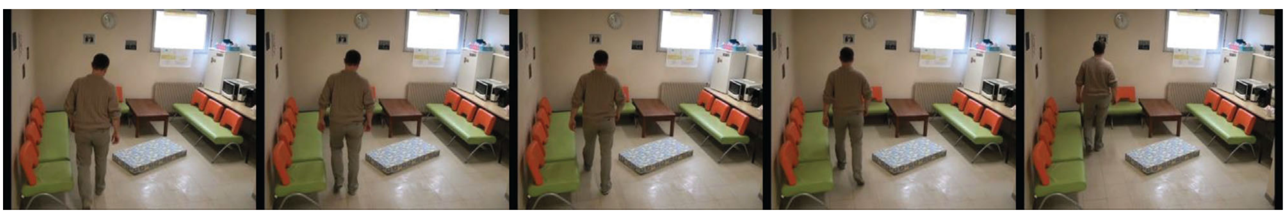

FIGURE 10 The image after processing the Le2i video

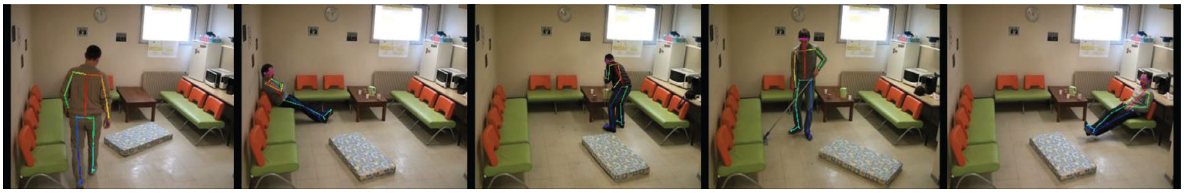

FIGURE 11 The image after labeling key points

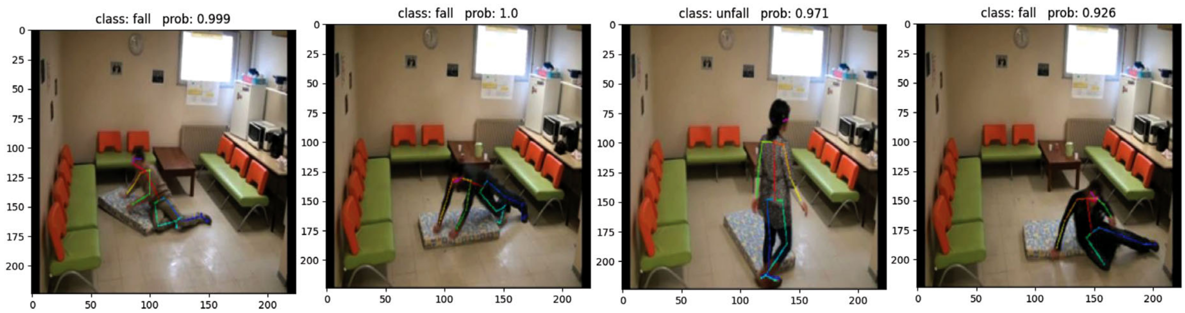

FIGURE 12 The detection results of the algorithm on the Le2i dataset

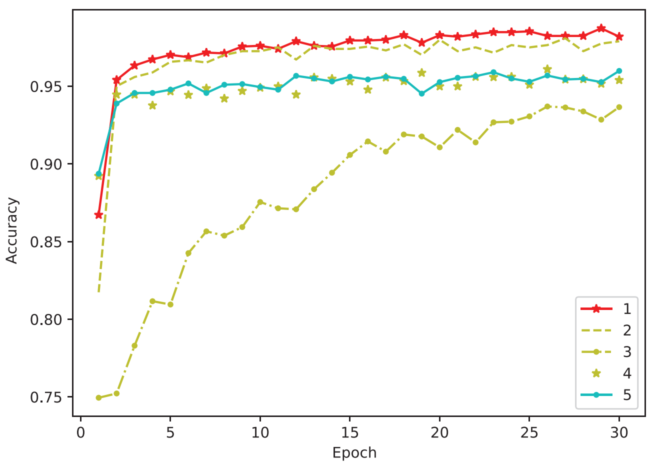

FIGURE 13 Accuracy curves of five methods on the Le2i dataset. 1:The proposed method. 2:only using RGB figures to detect falls. 3:3d-keypoints + SVM. 4:2d-keypoints + random forest. 5:Human pose estimation+KNN

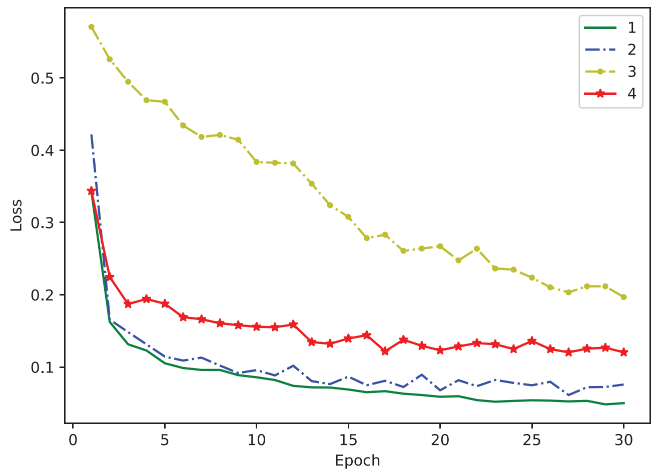

FIGURE 14 Loss curves of five methods on the Le2i dataset. 1:The proposed method. 2:only using RGB figures to detect falls. 3:3d-keypoints + SVM. 4:2d-keypoints + random forest. 5:Human pose estimation+KNN

among the five methods. Compared with method 2, which only uses RGB images for fall detection, it proves that the key point extraction and labeling can effectively improve the detection accuracy. At the same time, the proposed method is better than the methods 3, 4, and 5. Because when there is a certain deviation in key point extraction, only using key points instead of human posture information will have no way to correct the wrong key point information, resulting in an error in the final detection result. It can be seen from Figure 14 that the proposed algorithm has the fastest convergence speed and the best convergence.

Besides, we compared with other methods using the same dataset, and the comparison results are shown in Table 2. It can be seen that the accuracy of the method proposed in this paper

TABLE 2 Model performance comparison between the proposed algorithm and other existing algorithms on the Le2i dataset

Method	Precision	Recall	Accuracy
RGB-MobileNetV2	96.1%	96.32%	96.7%
Two stream CNN (3D-CNN+VGG-16) [27] (2017)	/	/	96%
Human pose estimation+Kinematic theory [28] (2021)	/	/	98%
Two stream CNN (RGB-MHI) [29] (2021)	98.67%	97.63%	98.12%
The proposed method	98.65%	97.7%	98.6%

FIGURE 15 There is a bias in extracting key points of the human body before the brightening process (the upper two sets of pictures), and the extraction of key points after the brightening process (the two sets of pictures below) is more accurate

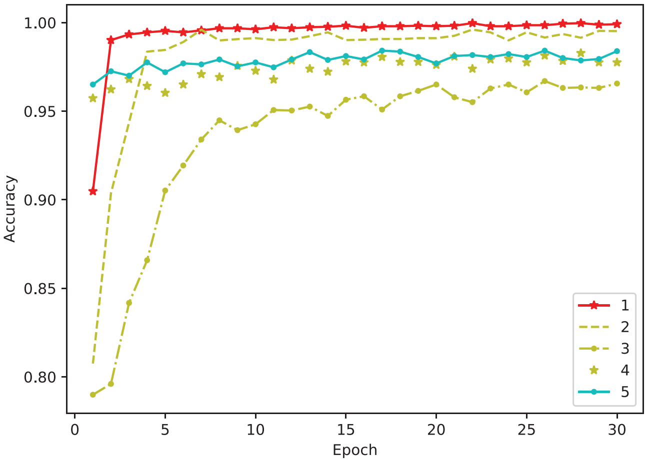

FIGURE 16 Accuracy curves of five methods on the UR dataset. 1:The proposed method. 2:only using RGB figures to detect falls. 3:3d-keypoints + SVM. 4:2d-keypoints + random forest. 5:Human pose estimation+KNN

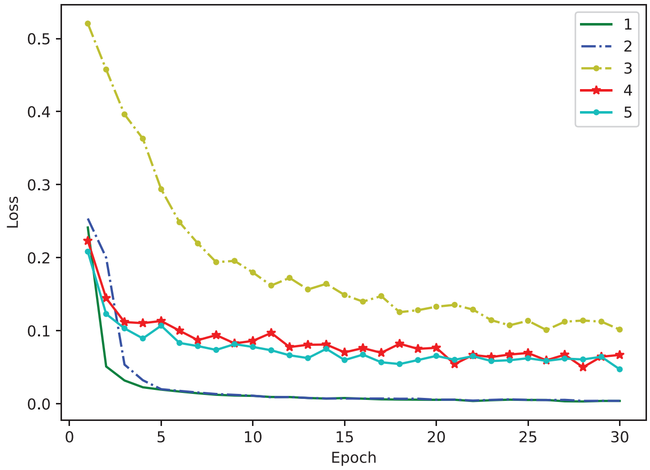

FIGURE 17 Loss curves of five methods on the UR dataset. 1:The proposed method. 2:only using RGB figures to detect falls. 3:3d-keypoints + SVM. 4:2d-keypoints + random forest. 5:Human pose estimation+KNN

is 98.6%, which is better than the other methods listed. Accordingly, it is proved that the proposed algorithm is effective on the Le2i dataset.

3.6 | Experiment on the UR dataset

We chose the RGB images in the UR fall dataset captured by camera 0 with a resolution of 640×480 for fall detection. Since there are several groups of image sequences with a particularly dark light in the UR data set, there is a certain deviation in the key point extraction of these images. To make keypoint extraction more accurate, we use Equation (2) to lighten the too dark picture. After calculating the average brightness value of normal pictures in other scenes, the minimum pixel value of the pictures in the normal brightness scene is 130.7. So we take 130 as the threshold for image brightness and brighten the photos whose brightness is below the threshold in the dark scene, and the pictures that are greater than the threshold are not changed. By calculating the pixel value distribution, we found that the average value of most image pixels is between 39 and 56. In order to contain more pixel information, we set the pixel existence

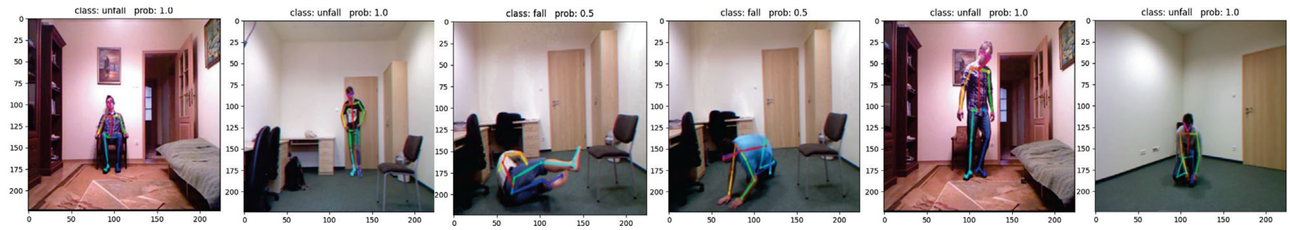

FIGURE 18 Detection results of the algorithm on the UR dataset

interval to [1,99]. Pixel values outside the interval are then culled. Finally, the image pixels with outliers removed are pulled back to the [0,255] interval. And to avoid pixel overflow, we set the interval to [255*0.1,255*0.9]. Where the x_o represents the original pixel value and the x_p represents the highlighted pixel value.

$$\frac{x\_o}{99-1} = \frac{x\_p}{\left[255 \times (0.9-0.1)\right]}. \tag{2}$$

As shown in Figure 15, the deviation of key point markings in low-light pictures is large (the upper two sets of images in Figure 11). The key points are mislabeled in the areas without human activity. However, the above situation disappears in the brightened picture, and the key points are marked more accurately (the two pictures below in Figure 11). This proves that the highlighting operation has played a certain role in promoting the labeling of the human body key points. Ultimately after image preprocessing, the training set and the testing set are divided according to the ratio of 7:3.

The output of the OpenPose is used as the input of the MobileNetV2 to detect falls. To compare the effect of feature enhancement on the UR dataset, the RGB images that have not been processed by OpenPose are directly input into MobileNetV2 to detect the falls. Besides, a comparison with the other three existing methods is also carried out to demonstrate the effectiveness of the proposed method. The accuracy and convergence curves of the comparison experiment are shown in Figures 16 and 17. It can be seen from Figure 16 that the proposed method has excellent accuracy among the five methods. Compared with method 2, which only uses RGB images for fall detection, it proves that the key point extraction and labeling can effectively improve the detection accuracy. At the same time, the proposed method is better than methods 3, 4, and 5. From Figure 17, the proposed algorithm has similar convergence values to the process that only uses RGB images for fall detection, but the proposed method has the fastest convergence speed. The above results proved that using the OpenPose algorithm to extract human key points for feature enhancement improves the detection accuracy of the algorithm model.

Figure 18 shows the detection result output of the algorithm in the testing dataset of the UR. The proposed algorithm can accurately detect whether there is falling behavior in each frame of the picture in the testing dataset.

Apart from that, the accuracy, precision and recall of the model are compared with other algorithms using the same dataset. The results are shown in Table 3. We can see that the accuracy of the algorithm proposed in this paper is 99.75%, which is higher than the listed other algorithms.

TABLE 3 Model results comparison

Method	Accuracy
Based on motion information [30] (2018)	99.6%
Keypoints+LSTM/GRU [31] (2020)	98.2%
Based on fused saliency maps [32] (2021)	99.67%
Dense block with a multi-channel convolutional fusion (MCCF) strategy [33] (2021)	96.6%
Two stream CNN (3D-CNN+VGG-16) [27] (2021)	99%
keypoints+PCA+classifier [34] (2021)	98.5%
The proposed method	99.75%

4 | CONCLUSION

Based on the current fall detection researches, an algorithm integrating features enhancement and fall detection is proposed to detect falls. Firstly, the OpenPose is introduced to extract human keypoints and label them in the images. After that, the modified MobileNetV2 network is used to detect falls by combining both human keypoint information and human pose information in the original images. The above operation can perform feature enhancement without increasing the complexity of the image and use the original image information to correct the deviation in the keypoint labeling process. Ultimately, the experiments on public datasets show that the proposed method realizes higher accuracy of fall detection.

AUTHOR CONTRIBUTIONS

Mengqi Gao: Conceptualization, data curation, formal analysis, investigation, methodology, project administration, software, validation, visualization, writing - original draft, writing - review and editing. Jiangjiao Li: Data curation, supervision, writing - review and editing. Dazheng Zhou: Writing - review and editing. Yumin Zhi: Writing - review and editing. Mingliang Zhang: Writing - review and editing. Bin Li: Funding acquisition, supervision, writing - review and editing.

ACKNOWLEDGEMENTS

This work was supported in part by the National Natural Science Foundation of China under Grant 61973185, in part by the

Development Plan of Young Innovation Team in Colleges and Universities of Shandong Province under Grant 2019KJN011, in part the Natural Science Foundation of Shandong Province under Grant ZR2020MF097 and the Colleges and Universities Twenty Terms Foundation of Jinan City (2021GXRC100).

CONFLICT OF INTEREST

The authors declare no conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are openly available in Le2i dataset at https://doi.org/10.1109/SITIS.2012.155, reference number [25]. And the data that support the findings of this study are openly available in UR fall dataset at https://doi.org/10.1016/j.cmpb.2014.09.005, reference number [26].

ORCID

Mengqi Gao https://orcid.org/0000-0001-6218-1132

REFERENCES

1. Moustafa, H., Kenn, H., Sayrafian, K., et al.: Mobile wearable communications. IEEE Wireless Commun. 22(1), 10–11 (2015). https://doi.org/10.1109/MWC.2015.7054713
2. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, vol. 25, pp. 1–9. MIT Press, Cambridge (2012). https://doi.org/10.1145/3065386
3. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. Comput. Sci. (2014). https://doi.org/10.48550/arXiv.1409.1556
4. Szegedy, C., Liu, W., Jia, Y., et al.: Going deeper with convolutions. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9. IEEE, Piscataway (2015)
5. He, K., Zhang, X., Ren, S., et al.: Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778. IEEE, Piscataway (2016)
6. Chan, T.H., Jia, K., Gao, S., et al.: PCANet: A simple deep learning baseline for image classification. IEEE Trans. Image Process. 24(12), 5017–5032 (2015). https://doi.org/10.1109/TIP.2015.2475625
7. Abdo, H., Amin, K.M., Hamad, A.M.: Fall detection based on RetinaNet and MobileNet convolutional neural networks. In: 2020 15th International Conference on Computer Engineering and Systems (ICCES), pp. 1–7. IEEE, Piscataway (2020)
8. Liu, Y.H., Hung, P.C.K., Iqbal, F., et al.: Automatic fall risk detection based on imbalanced data. IEEE Access 9, 163594–163611 (2021). https://doi.org/10.1109/ACCESS.2021.3133297
9. Vishnu, C., Datla, R., Roy, D., et al.: Human fall detection in surveillance videos using fall motion vector modeling. IEEE Sens. J. 21(15), 17162–17170 (2021). https://doi.org/10.1109/JSEN.2021.3082180
10. Chen, W., Jiang, Z., Guo, H., et al.: Fall detection based on key points of human-skeleton using openpose. Symmetry 12(5), 744–761 (2020). https://doi.org/10.3390/sym12050744
11. Lu, N., Wu, Y., Feng, L., et al.: Deep learning for fall detection: Three-dimensional CNN combined with LSTM on video kinematic data. IEEE J. Biomed. Health. Inf. 23(1), 314–323 (2018). https://doi.org/10.1109/JBHI.2018.2808281
12. Nizam, Y., Mohd, M.N.H., Jamil, M.M.A.: Human fall detection from depth images using position and velocity of subject. Procedia Comput. Sci. 105, 131–137 (2017). https://doi.org/10.1016/j.procs.2017.01.191
13. Lv, X., Gao, Z., Yuan, C., et al.: Hybrid real-time fall detection system based on deep learning and multi-sensor fusion. In: 2020 6th International Conference on Big Data and Information Analytics (BigDIA), pp. 386–391. IEEE, Piscataway (2020)
14. Sekii, T.: Pose proposal networks. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 342–357. Springer, Berlin (2018)
15. Cao, Z., Simon, T., Wei, S.E., et al.: Realtime multi-person 2d pose estimation using part affinity fields. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 7291–7299. IEEE, Piscataway (2017)
16. Fu, N., Liu, D., Cheng, X., et al.: Fall detection algorithm based on lightweight OpenPose model. Sens. Microsyst 40(11), 131–134 (2021)
17. Ramirez, H., Velastin, S.A., Meza, I., et al.: Fall detection and activity recognition using human skeleton features. IEEE Access 9, 33532–33542 (2021). https://doi.org/10.1109/ACCESS.2021.3061626
18. Sandler, M., Howard, A., Zhu, M., et al.: Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 4510–4520. IEEE, Piscataway (2018)
19. Kapse, A.S., Shoba, S., Tamuli, R., et al.: Android based fall detection and tracking app for aged people. In: 2022 Second International Conference on Artificial Intelligence and Smart Energy (ICAIS), pp. 1113–1116. IEEE, Piscataway (2022)
20. Paneerselvam, A., Yaakob, R., Perumal, T., et al.: Fall detection framework for smart home. 2018 IEEE 7th Global Conference on Consumer Electronics (GCCE), pp. 351–352. IEEE, Piscataway (2018)
21. Astriani, M.S., Bahana, R., Kurniawan, A., et al.: Promoting data availability framework by using gamification on smartphone fall detection based human activities. Procedia Comput. Sci. 179, 913–919 (2021). https://doi.org/10.1016/j.procs.2021.01.100
22. Lin, B.-S., et al.: Fall detection system with artificial intelligence-based edge computing. IEEE Access 10, 4328–4339 (2022). https://doi.org/10.1109/ACCESS.2021.3140164
23. Chang, W.-J., Hsu, C.-H., Chen, L.-B.: A pose estimation-based fall detection methodology using artificial intelligence edge computing. IEEE Access 9, 129965–129976 (2021). https://doi.org/10.1109/ACCESS.2021.3113824
24. Woo, S., Park, J., Lee, J.Y., et al.: Cbam: Convolutional block attention module. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 3–19. Springer, Berlin (2018)
25. Charfi, I., Miteran, J., Dubois, J., et al.: Definition and performance evaluation of a robust SVM based fall detection solution. In: 2012 Eighth International Conference on Signal Image Technology and Internet Based Systems, pp. 218–224. IEEE, Piscataway (2012)
26. Kwolek, B., Kepski, M.: Human fall detection on embedded platform using depth maps and wireless accelerometer. Comput. Methods Programs Biomed. 117(3), 489–501 (2014). https://doi.org/10.1016/j.cmpb.2014.09.005
27. Yuan, Z., Hu, H.: A fall recognition method based on two-stream convolutional neural network. J. Henan Normal Univ. (Natural Sci. Ed.) 45(03), 96–101 (2017). https://doi.org/10.16366/j.cnki.1000-2367.2017.03.014
28. Dentamaro, V., Impedovo, D., Pirlo, G.: Fall detection by human pose estimation and kinematic theory. In: 2020 25th International Conference on Pattern Recognition (ICPR), pp. 2328–2335. IEEE, Piscataway (2021)
29. Jin, Y., Chen, G.: Real-time fall behavior detection based on two-stream convolution. Comp. Eng. Design 42(09), 2621–2626 (2021). https://doi.org/10.16208/j.issn1000-7024.2021.09.029
30. Lotfi, A., Albawendi, S., Powell, H., et al.: Supporting independent living for older adults; employing a visual based fall detection through analysing the motion and shape of the human body. IEEE Access 6, 70272–70282 (2018). https://doi.org/10.1109/ACCESS.2018.2881237
31. Lin, C.B., Dong, Z., Kuan, W.K., et al.: A framework for fall detection based on openpose skeleton and lstm/gru models. Appl. Sci. 11(1), 329 (2020). https://doi.org/10.3390/app11010329
32. Li, H., Li, C., Ding, Y.: Fall detection based on fused saliency maps. Multim. Tools Appl. 80(2), 1883–1900 (2021). https://doi.org/10.1007/s11042-020-09708-6

33. Cai, X., Liu, X., An, M., et al.: Vision-based fall detection using dense block with multi-channel convolutional fusion strategy. IEEE Access 9, 18318–18325 (2021). https://doi.org/10.1109/ACCESS.2021.3054469
34. Alaoui, A.Y.El Fkihi, S., Thami, R.O.H.: Fall detection for elderly people using the variation of key points of human skeleton. IEEE Access 7, 154786–154795 (2019). https://doi.org/10.1109/ACCESS.2019.2946522

How to cite this article: Gao, M., Li, J., Zhou, D., Zhi, Y., Zhang, M., Li, B.: Fall detection based on OpenPose and MobileNetV2 network. IET Image Process. 17, 722–732 (2023). https://doi.org/10.1049/ipr2.12667