Mathematical Biosciences and Engineering

MBE, 19(2): 1195–1212.
DOI: 10.3934/mbe.2022055
Received: 30 September 2021
Accepted: 23 November 2021
Published: 30 November 2021

http://www.aimspress.com/journal/MBE

Research article

A lightweight double-channel depthwise separable convolutional neural network for multimodal fusion gait recognition

Xiaoguang Liu[1,2], Meng Chen[1,2], Tie Liang[1,2], Cunguang Lou[1,2], Hongrui Wang[1,2] and Xiuling Liu[1, 2,*]

[1] College of Electronic and Information Engineering, Hebei University, Baoding, Hebei, China
[2] Key Laboratory of Digital Medical Engineering of Hebei Province, Hebei University, Baoding Hebei, China

* **Correspondence:** Email: liuxiuling121@hotmail.com.

Abstract: Gait recognition is an emerging biometric technology that can be used to protect the privacy of wearable device owners. To improve the performance of the existing gait recognition method based on wearable devices and to reduce the memory size of the model and increase its robustness, a new identification method based on multimodal fusion of gait cycle data is proposed. In addition, to preserve the time-dependence and correlation of the data, we convert the time-series data into two-dimensional images using the Gramian angular field (GAF) algorithm. To address the problem of high model complexity in existing methods, we propose a lightweight double-channel depthwise separable convolutional neural network (DC-DSCNN) model for gait recognition for wearable devices. Specifically, the time series data of gait cycles and GAF images are first transferred to the upper and lower layers of the DC-DSCNN model. The gait features are then extracted with a three-layer depthwise separable convolutional neural network (DSCNN) module. Next, the extracted features are transferred to a softmax classifier to implement gait recognition. To evaluate the performance of the proposed method, the gait dataset of 24 subjects were collected. Experimental results show that the recognition accuracy of the DC-DSCNN algorithm is 99.58%, and the memory usage of the model is only 972 KB, which verifies that the proposed method can enable gait recognition for wearable devices with lower power consumption and higher real-time performance.

Keywords: wearable devices; gait recognition; gait cycles; multimodal fusion; DC-DSCNN

1. Introduction

With the dynamic development of sensor technology, small and low-cost wearable devices are widely deployed in the medical and health fields. However, wearable devices cannot deploy sophisticated security authentication mechanisms due to limited computing and storage capacity [1], which puts the privacy of wearable device owners at risk [2]. In recent years, biometric-based methods have become the preferred method of identification. Currently, mobile devices have implemented identification systems based on biometric features, such as the face [3] and fingerprints [4]. However, some of these biometric techniques are intrusive to users and require complex hardware that are not conducive to their application in wearable devices. Therefore, biometrics based on non-intrusive and fewer additional hardware systems, such as gait recognition, have attracted a lot of attention from academia and industry [5].

For gait recognition based on wearable devices, it is more common to use inertial sensors. A sensor-based gait recognition method was first proposed by Ailisto et al. [6]. The acceleration signal corresponding to the waist during walking was collected and gait recognition was performed using template matching and cross-correlation calculation. Based on their research, Rong et al. [7] used the dynamic time warping (DTW) algorithm for gait-curve matching. In 2019, Sun et al. [8] proposed a speed-adaptive gait cycle segmentation method and an adaptive matching threshold generation method. The experimental results showed that their method achieved a 96.9% user recognition accuracy. However, the robustness of the template matching method for gait recognition in complex environments is poor. With the development of artificial intelligence technology, gait recognition methods based on machine learning, such as support vector machine (SVM) [9] and k-Nearest Neighbor (KNN) [10], are increasing. Although machine learning methods can be effective for gait recognition, it is labor-intensive and time-consuming.

In recent years, deep learning-based gait recognition methods have become popular [11–16]. In 2016, Gadaleta et al. [11] used convolutional neural networks (CNNs) as feature extractors for gait recognition for the first time. They designed an IDNet user recognition framework based on a CNN and one-class SVM. The results of this experiment showed that the CNN can automatically learn gait features and obtain better recognition performance. In 2018, Ruben et al. [12] proposed an end-to-end method based on deep learning, which uses information from multiple sensors as the input of a single-channel CNN separately and fuses the extraction results. They conducted experiments on the OU-ISIR dataset, and the accuracy of identity recognition using their method improved from 83.3% to 94.8%. In 2020, Zou et al. [13] used smartphones to collect accelerometer and gyroscope data while walking in a field environment for recognition and authentication. Then, they proposed a method that used a deep neural network that combined a CNN and long short-term memory (LSTM). Identification tests were performed on a dataset containing 118 subjects, and an accuracy of 93.7% was obtained. To extract the temporal features of the gait, Tran et al. [14] proposed a new multi-model LSTM network which has six channels. They put each signal data from a group of continuous signals into one of the channels of one LSTM, respectively. Then, they constructed a hybrid architecture which combined the LSTM network and the CNN network. The experimental results showed that the identification accuracy was 94.15% for the whuGait dataset. In 2021, Middya et al. [15] proposed a new deep CNN model for privacy protected user identification. They evaluated the model on a real-world benchmark dataset and achieved an accuracy of 98.8%. Although these methods can achieve better recognition performance, the models have a large number of model parameters and high memory usage, which is

not suitable for wearable devices [16].

In order to improve the performance of gait identification based on wearable devices and reduce the memory size and improve robustness of the model, we propose a multimodal fusion method based on a double-channel depthwise separable convolutional neural network (DC-DSCNN). We improve the model by reducing the model complexity to make it suitable for wearable devices. In addition, we improve the robustness of gait recognition by fusing the time series data and Gramian angular field (GAF) images. The experimental results show that the method proposed in this paper has high classification accuracy and small memory size.

The main contributions of this paper are as follows:

1) To evaluate the performance of models for gait recognition, we build one gait dataset which is collected from 24 subjects. Each subject provides 6 samples and each sample contains thousands of data points. Then, the gait cycles are divided according to the sample data. We selected 100 gait cycles from each subject dataset. Therefore, we have a total of 2400 gait cycle samples.

2) The new lightweight DSCNN network based on depthwise separable convolution is proposed in this paper which can automatically extract features and the number of parameters is small. In addition, this model can increase recognition accuracy while reducing model memory requirements.

3) A multimodal fusion DC-DSCNN network based on the time series data and GAF images of acceleration and angular velocity is proposed which can automatically learn gait features and realize gait identification.

The rest of the paper is organized as follows. Section 2 describes the process of gait cycle data acquisition, extraction, and processing in the HD Dataset based on wearable devices. We also briefly introduce the basic principles of GAF and depthwise separable convolution (DSC) in this section. The details of the proposed model are also presented in this section. The experimental results are given in Section 3. Finally, the conclusion and future work are discussed in Section 4.

2. Materials and methods

2.1. Data collection unit

In this paper, a miniature wireless inertial-magnetic motion tracker (MTw), which includes a 3D accelerometer and a 3D gyroscope, was used to collect gait data. The three-axis accelerometer obtains the axial acceleration by measuring the force of the wearable device in a certain axis (X, Y, or Z). Meanwhile, acceleration measured by the accelerometer can reflect movement of the wearable device user. The working principle of the three-axis gyroscope is to measure the angle between the vertical axis of the gyroscope rotor and the device in a three-dimensional coordinate system, and calculate the angular velocity. Therefore, the gyroscope can capture the angular velocity by measuring its own rotational state, which also can determine the user's movement state. To improve the recognition accuracy, a method of fusing acceleration and angular velocity data was adopted.

Considering identification without interfering with normal activities, wearable sensors should be designed to be lighter, smaller, and easier to wear. This paper selected the MTw (Figure 1) designed by Xsens as the data collection unit [17]. It has a size of 47 mm × 30 mm × 13 mm and its weights 16 g. The MTw was placed in the right of the waist and collected the acceleration and angular velocity data of the waist with a sampling frequency of 100 Hz. The MTw wirelessly transmits acceleration and angular velocity data to the PC in real time based on the Awinda Station connected to a recording PC.

The process of collecting gait data is illustrated in Figure 1.

Figure 1. Data acquired using the MTw.

2.2. Datasets

2.2.1. HD dataset

In this study, nine male and fifteen female volunteers were recruited via random sampling to collect their gait data during walking. All the subjects were informed of the purpose of the experiment, and data collection was performed with verbal consent. The age of volunteers was in the range of 22 to 60 years, and the height was between 1.58 and 1.87 meters, and the average weight was 45.5 kg. All the volunteers exhibited no abnormal gait during the experiment. In the experiment, the subjects' shoes and walking environment were free. For the homogeneity of the experimental data, the MTw was worn on the right side of the waist of all subjects, the Z-axis was aligned with the direction of gravity, the X-axis represented the forward direction, and the Y-axis indicated the horizontal direction. Subjects were asked to walk for one minute on a specified path at their own walking speed and repeat this six times. The inertial signal data were recorded at the beginning of the walk, and the data were saved after completing one path cycle, thus each volunteer has six gait samples. The data in the dataset were collected at a sampling frequency of 100 Hz and processed by a Kalman filter. For gait identification, all collected data were processed using the method described in Section 2.3. We constructed the HD Dataset by selecting 100 gait cycle segments from each subject dataset. The gait cycle segments were then labeled according to the volunteer's number. After matching all gait cycle segments and labels, we split the dataset into training and testing sets with the ratio of 7:3 by using a random shuffle method. There are 1680 training samples and 720 test samples. The detailed information of these datasets is shown in Table 1.

Table 1. Detailed information of the datasets.

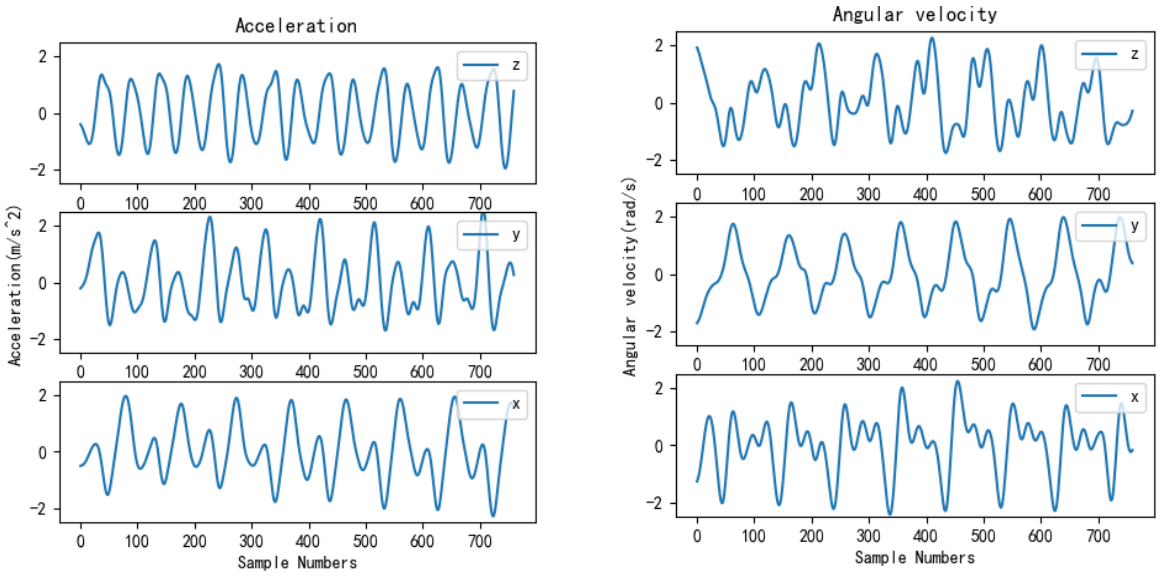
Dataset	Usage	Number of subjects	Samples for Training	Samples for Test
HD Dataset	Identification	24	1,680	720
whuGait Dataset #1	Identification	118	33,104	3,740

2.2.2. whuGait dataset

Zou et al. [13] constructed the whuGait dataset by collecting gait data using a smartphone's inertial sensor under unrestricted conditions in the field, and 118 subjects' gait data were collected. Each sample in the dataset contains 3-axis acceleration data and 3-axis angular velocity data with a sampling frequency of 50 Hz. Four datasets were constructed for gait recognition and two datasets can be used for gait authentication. In this study, Dataset #1 was selected to validate the gait recognition performance of the model. It contains 33,104 training samples and 3,740 test samples, and each subsample contains two gait cycles. Table 1 shows the details of Dataset #1.

2.3. Gait cycle extraction and processing

The movement of moving the body forward using a series of repetitive limb movements while ensuring stability is known as walking. During the forward movement of the body, one lower limb acts as a source of support and the other one swings forward and becomes the next new source of support when the heel hits the ground. Subsequently, the two lower limbs keep exchanging roles until they reach the destination. The completion of a single sequence from a heel landing to landing again with one lower limb is referred to as a gait cycle [18]. A sample of gait data is shown in Figure 2, which includes acceleration (a) and angular velocity (b) curves.



(a)Acceleration (b)Augular velocity

Figure 2. Data acquired using the MTw.

From Figure 2 it can be seen that the gait data have a certain periodicity. The period stability of the gait signal on the Z-axis is better than that on the X-axis and Y-axis. Therefore, the gait signal on the Z-axis is usually used to extract the gait cycle. Experimental analysis [19] has shown that the acceleration in the direction of gravity reaches the maximum value when the moment of heel landing and its minimum value corresponds to the initial force point when the foot stomp forward. By

analyzing the gait data of the subjects, it was found that the gait cycle which is divided by the maximum value of the acceleration in the Z-axis was more stable in terms of the shape of the data curve and the number of sample points than the minimum value. Therefore, the location of the maximum value point of the acceleration in the Z-axis is used as the basis for dividing the gait cycle segment in this experiment.

2.3.1. Gait cycle segmentation

Extracting the gait cycle is one of the key steps which affects the recognition accuracy of gait recognition systems. Techniques based on threshold or peak detection are the most widely utilized for gait cycle detection [20]. In this study, we use the maximum value of the gait acceleration signal in the direction of human gravity (Z-axis) as fundamental and used an algorithm combining peak detection and dynamic time warping (DTW) to segment the gait acceleration and angular velocity signal to achieve automatic segmentation of the gait cycle. Meanwhile, the time of each step in normal uniform walking is about 0.4 to 0.6 seconds. The sampling frequency of the MTw is 100 Hz, so that each gait cycle contains 80 to 120 sample points. The specific algorithm steps are as follows:

First traverse all the sample points of the Z-axis of the acceleration signal and detect all the maximum points. The principle of maximum value detection can be simply described as in the following formula:

$$x_{i-1} < x_i \,\&\, x_i > x_{i+1} \tag{1}$$

where x_i is the sample point at the current moment, x_{i-1} is the sampling points at the previous moment, and x_{i+1} is the sample points at the next moment.

Second, the pseudo-extreme points are removed according to a threshold value. The maximum points of the Z-axis of the acceleration data smaller than 9.8 m/s 2 are removed because the acceleration of a person's foot falling to the ground when walking should be greater than the acceleration of gravity. In addition, the time interval between two consecutive local maximum points should be between 0.4 and 0.6 seconds. The maximum value points that do not satisfy this requirement are removed. Figure 3 shows the results of maximum value point detection.

Finally, the gait data are divided into cycle segments according to the index value corresponding to the maximum point of acceleration.

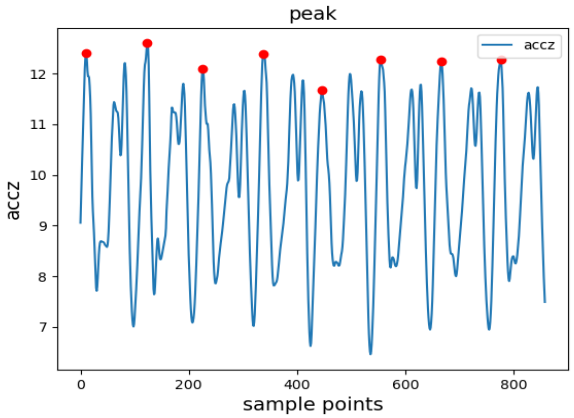
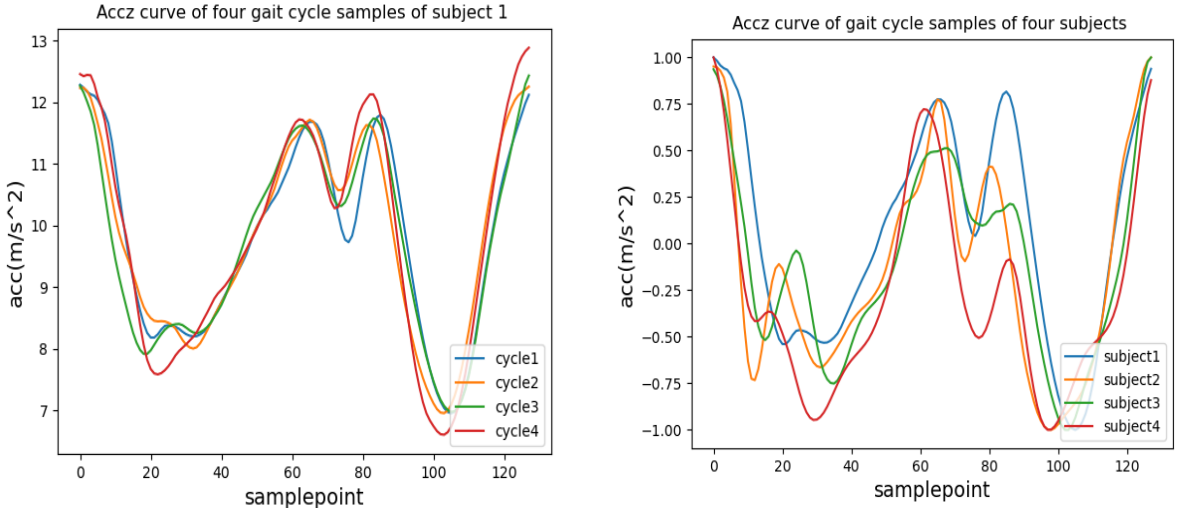


Figure 3. The result of the maximum value detection. The red dots represent the gait cycle segmentation points. The data between two red dots is one gait cycle.

To obtain the typical gait cycle segments which have a similar waveform trend, we utilize the DTW algorithm to match the similarity and filter out those gait cycles with insignificant periodicity and similarity. DTW [21] is a method to measure the similarity between two unequal long time series. It calculates the distance between two time series by lengthening and shortening them. If the distance of two time series is beyond the threshold, they are deemed as dissimilar. Since there are some disturbing gaits during walking, these should be eliminated. The gait cycle with significant periodicity is selected as a template. The DTW algorithm is used to calculate the similarity between the remaining segments and the template, and then gait cycle segments whose similarity exceeds a threshold are removed. Therefore, a series of gait cycle segments with similar waveform of the same subject are extracted and these have similar feature values which facilitates gait recognition.

2.3.2. Gait cycle processing

A user's walking speed will vary under the influence of different scenarios and external factors. Even in the same scene, the walking speed will be slightly affected by different factors. This study uses interpolation to process samples of gait cycles at different walking speeds. We use cubic linear interpolation to process all gait cycle segments and turn each gait cycle segment into a sample with 128 sample points. Therefore, the inertial signal sequence in each gait cycle will have 128 sample points after interpolation, regardless of the subject's stride. This is conducive to the later data processing and gait recognition. Figure 4(a) shows the Z-axis acceleration curve of the same subject for four gait cycles. Figure 4(b) shows gait cycles of the Z-axis acceleration curves of four different subjects. From Figure 4 we can see that the fluctuation trend of the sample curve of the same subject is roughly the same, and the fluctuation trend of the sample curve of different subjects has certain differences, so that it can be identified according to the gait cycle segmentation samples.



(a) Gait cycles from the same subject (b) Gait cycles from four subjects

Figure 4. Gait cycle comparison.

2.4. Gramian angular filed algorithm

In wearable device-based gait recognition, the accelerometer and gyroscope are the most commonly used sensors. Most of the raw gait data collected by wearable sensors are one-dimensional time series. Although deep learning methods (e.g., 1D CNN, LSTM, etc.) can process one-dimensional time series data while preserving nonlinear features, their correlation on the time series is not fully considered. Wang et al. [22] proposed to preserve the time dependence and correlation of the original data by converting one-dimensional time series into two-dimensional images using the GAF algorithm. The GAF algorithm represents the time series through a polar coordinate system instead of a Cartesian coordinate system, which evolved from the Gram matrix. The specific implementation of the GAF algorithm steps [22] are as follows:

Given a time series $X = \{x_1, x_2, \ldots x_N\}$ which contains N real-valued observations; first, normalization is conducted to ensure that X is in the range [-1,1] as follows:

$$\tilde{x}_{i-1}^{i} = \frac{(x_i - max(X)) + (x_i - min(X))}{max(X) - min(X)} \tag{2}$$

Next, the normalized one-dimensional time series is used to preserve the absolute time relationship of the series using polar coordinates, encoding the values as angular cosines and the timestamps as radius, which can be expressed by the following equation:

$$\begin{cases} \varphi_i = \arccos(\tilde{x}_i), & -1 \le \tilde{x}_i \le 1, \tilde{x}_i \in \tilde{X} \\ r_i = \frac{t_i}{N}, & t_i \in N \end{cases} \tag{3}$$

where $\tilde{x}_i$ is any observation in X, t_i is the timestamp corresponding to the time series X, and N is the total length of the timestamp. When converted to the polar coordinate system, the cosine values of the data obtained by the above normalization operation in the range [-1,1] fall into angular bounds [0, π].

Finally, after transforming the rescaled time series into the polar coordinate system by calculating the sum of the trigonometric function between each sample point, the temporal correlation among different time intervals is identified from the perspective of the angle. The Gramian Angular Summation Field (GASF) is defined as follows:

$$GASF = \begin{pmatrix} cos(\varphi_1 + \varphi_1) & cos(\varphi_1 + \varphi_2) & \cdots & cos(\varphi_1 + \varphi_n) \\ cos(\varphi_2 + \varphi_1) & cos(\varphi_2 + \varphi_2) & \cdots & cos(\varphi_2 + \varphi_n) \\ & \vdots & & \\ cos(\varphi_n + \varphi_1) & cos(\varphi_n + \varphi_2) & \cdots & cos(\varphi_n + \varphi_n) \end{pmatrix} \tag{4}$$

Briefly, there are three steps to convert a one-dimensional time series to a two-dimensional image using the GAF algorithm: scaling, coordinate system conversion, and trigonometric functions. Figure 5 shows the complete process of transforming the rescaled time series into an encoded map in polar coordinates. The time series is normalized using Eqs (2) and (3) converts it to the polar coordinate system. Finally, the image can be obtained by using the GASF (Eq (4)).

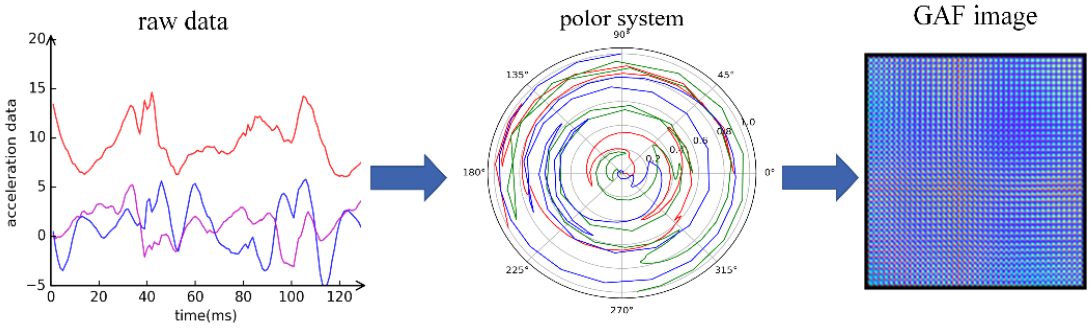


Figure 5. The process of converting a time series to an image.

2.5. Depthwise separable convolution

Inspired by the MobileNet model [23], our model mainly utilizes the depthwise separable convolution in the convolution layer. The depthwise separable convolution [24] consists of two parts: depth convolution and point convolution, which is primarily used to extract features. When feature extraction is performed on multi-channel inputs, one filter of the depth convolution corresponds to one input channel. Therefore, intermediate features of multiple channels can be obtained by the convolution operation. The pointwise convolution applies multiple 1 × 1 convolution kernels to the intermediate features to perform standard convolution operations and obtain multiple outputs with the same height and width as the input image. These outputs are combined on the channel axes to produce the final output. Using depthwise separable convolution has the effect of significantly reducing the number of parameters and computational cost, thus further improving the recognition efficiency. Figure 6 shows the convolution process for DSC.

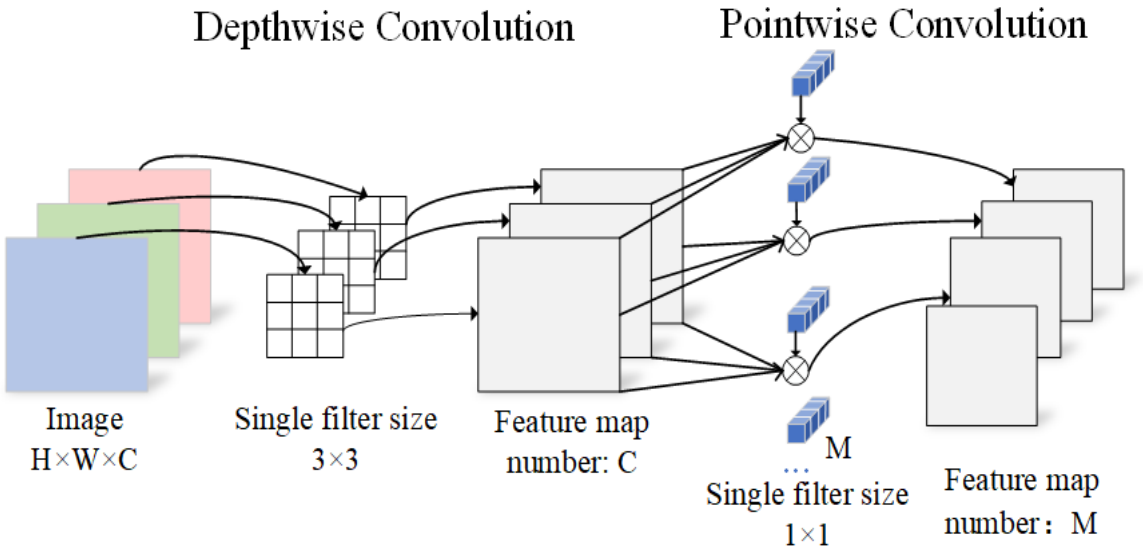


Figure 6. Example of the convolution process using the depthwise separable convolution.

The size of the input image is assumed to be H × W × C, where H, W, and C are the height, width, and number of channels of the input image, respectively. The deep convolution is performed for each channel using a convolution kernel of size 3 × 3, and the parameters of the deep convolution are calculated as follows:

$$N_{depthwise} = H \times W \times C \times 3 \times 3 \tag{5}$$

For pointwise convolution, the output channels of the feature maps generated by deep convolution are expanded by M convolution kernels of size 1×1. The cost of computing the parameters of pointwise convolution is:

$$N_{pointwise} = H \times W \times C \times 1 \times 1 \times M \tag{6}$$

Therefore, the computational volume of the depthwise separable convolution is the weighted value of the depthwise convolution and the pointwise convolution:

$$N_{separable} = N_{depthwise} + N_{pointwise} = H \times W \times C \times (3 \times 3 + M) \tag{7}$$

For standard convolution, the parameters are computed as:

$$N_{standard} = H \times W \times C \times 3 \times 3 \times M \tag{8}$$

By comparing Eqs (7) and (8), we can see that the computation of parameters for the depthwiseseparable convolution is reduced by (M × 9)/(M + 9) times compared to the standard convolution. Therefore, the computational parameters of the DSCNN network based on DSC can also be reduced and the gait recognition accuracy can be improved.

2.6. Double-channel separable convolutional neural network

In this paper a gait recognition method using multi-modal fusion is proposed, and a DC-DSCNN structure is designed. As shown in Figure 7, the two channels perform feature extraction on the time series data corresponding to the acceleration and angular velocity of gait and the GAF image, respectively. After this, the fusion of features is performed and the recognition results are output. The time series data and image will be separately trained with DSCNN, which can obtain the feature space of the gait data in different modalities and facilitate the optimal gradient descent during training. Thus, the feature information of the gait data can be better extracted.

2.6.1. Depthwise separable convolutional neural network

The network structure and parameters of the DSCNN model are shown in Table 2. The DSC in DSCNN is used to extract features. Three DSC layers containing different numbers of convolution kernels (32, 64, and 128) are used in the proposed model. A batch normalization layer is added between each DSC layer and the activation function layer. This is used to normalize the feature maps generated by the DSC layers. The purpose of this operation is to prevent gradient disappearance [25]. ReLU is chosen as the activation function that will compensate for the expressive deficiency of the linear model and mitigate the overfitting phenomenon. In addition, the max pooling layer with kernel of size 1 × 2 and stride of 2 is added after each the activation layer. The DSCNN module can be denoted as in Eq (9).

$$Y_{(H',W',C')} = MaxP(\delta_N(D_S(F_{(\mathrm{H,W,C})}))) \tag{9}$$

where F represents the feature map of size H × W × C. D_s represents the depthwise separable convolution operation. The kernel size of the DSC layer is set to 1 × 3. Batch normalization and the activation function are indicated by δ_N. MaxP is max pooling to reduce the size of the feature maps.

Global average pooling (GAP) is appended after the last module. It is a combination of two processes of the fully connected layer (FCL), where the feature map is expanded into multiple feature matrices and classified, thus eliminating the intermediate connection weight parameters and reducing a large number of parameters. The final layer is a softmax classifier that obtains the probability distribution associated with the input sample and outputs the classification results.

Table 2. Structure and parameters of the DSCNN model.

Layer Name	Kernel Size	Kernel Num.	Stride	Feature Map
DSConv1	1 × 3	32	1	1 × 128 × 32
BN	/	/	/	1 × 128 × 32
Activation	/	/	/	1 × 128 × 32
Pool1	1 × 2	/	2	1 × 64 × 32
DSConv2	1 × 3	64	1	1 × 64 × 64
BN	/	/	/	1 × 64×64
Activation	/	/	/	1 × 64 × 64
Pool2	1 × 2	/	2	1 × 32 × 64
DSConv3	1 × 3	128	1	1 × 32 × 128
BN	/	/	/	1 × 32 × 128
Activation	/	/	/	1 × 32 × 128
Pool3	1 × 2	/	2	1 × 16 × 128

2.6.2. Multimodal fusion

For gait recognition with the time series and GAF image, a multimodal fusion gait recognition method based on DC-DSCNN is proposed. The gait recognition process is shown in Figure 7. The multimodal fusion process combines information from the time series and GAF images of gait data to improve the accuracy of the prediction results and the robustness of the prediction model.

The identification process using the GAF image is as follows. The gait cycle segments are extracted from the time series acquired by the accelerometer or gyroscope, and then the one-dimensional time series are converted into a two-dimensional image matrix using the GAF algorithm. Taking the data acquired by the three-axis gyroscope as an example, the length of the gait cycle segment is given as S. Since each sample contains three channels, the dimension of each sample is S × 3. By using the GAF algorithm, the three-dimensional time series is converted into a two-dimensional image with three channels. In this study, the size of the GAF image is 128 × 128 × 3. This will obtain a GAF image in the same format as an ordinary RGB image. After this, the GAF images obtained from the angular velocity data are input to the DSCNN model. The result of the classification is the label corresponding to the highest probability after feature extraction. Experimental comparative results for gait recognition based on acceleration and angular velocity are presented in Section 3.3.

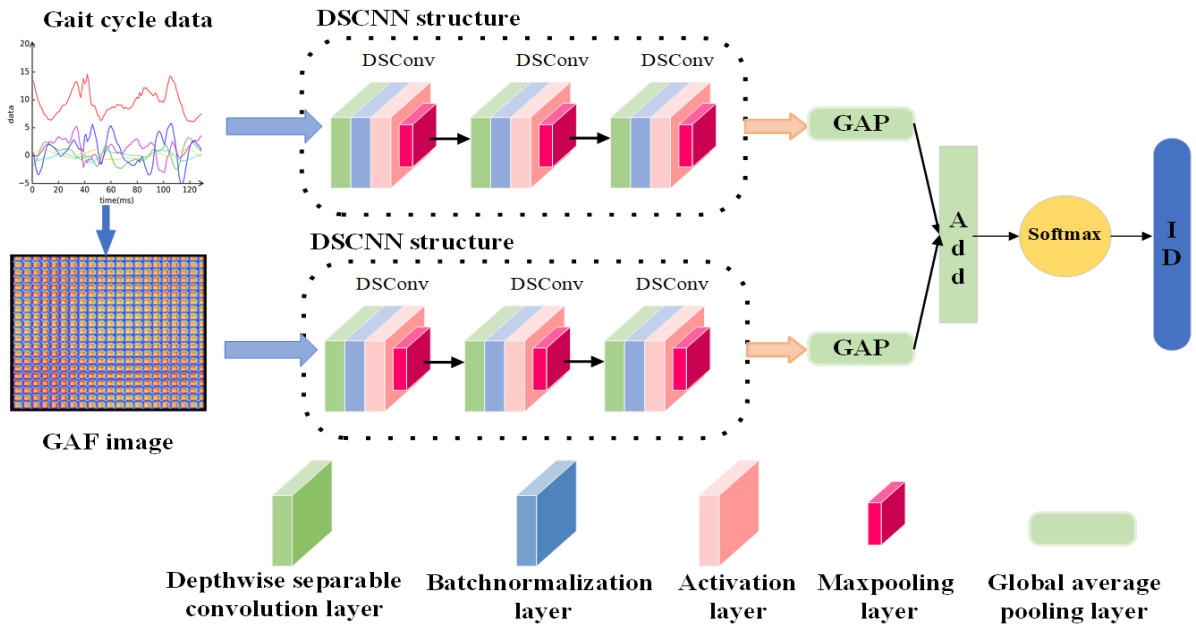


Figure 7. Gait recognition network architecture.

To improve the recognition rate and robustness, the time series and GAF images of gait cycles are used as inputs for two channels. For the upper layer network architecture, the gait cycle time series data are transferred to the DSCNN model, and after a series of feature extraction operations, the results are fed into the GAP layer. For the lower layer network, the time series data of the gait cycle are first converted into GAF images and then the same operation as the upper layer network is performed. Finally, the feature extraction results of both channels are added in a fusion layer to produce a joint feature vector. Then, the joint feature vector is fed into the softmax classifier. The classification label corresponding to the obtained maximum probability is the gait identification result.

2.6.3. Selection of activation function

Wang et al. [26] proposed a new CNN-based image recognition method in which an exponential linear unit (ELU) [27] was applied, and obtained higher recognition accuracy than state-of-the-art methods. Hence, we chose ReLU [28] and ELU as the activation function to train the DC-DSCNN model. The equation for ReLU is as follows:

$$f_{ReLU} = max(0, x) = \begin{cases} x, x \geq 0 \\ 0, x < 0 \end{cases} \tag{10}$$

ReLU can avoid the gradient saturation phenomenon and speed up the training process, but it has a problem that the gradient is 0 when x ≤ 0. ELU solves this problem by reducing the effect of bias offset, which brings the normal gradient closer to the unit natural gradient.

$$f_{ELU} = \begin{cases} \gamma(e^x - 1), x \leq 0 \\ x, \qquad\quad x > 0 \end{cases} \tag{11}$$

3. Results

3.1. Evaluation metrics

In order to verify the classification performance of the proposed DC-DSCNN network, we need to introduce the evaluation indicators. For gait recognition using deep learning, classifier performance can be measured by computing accuracy, precision, recall, and F1-score. The accuracy is the ratio of the number of samples correctly classified by the classifier to the total number of samples for a given test dataset. A higher accuracy indicates better predictive ability of the model. The F1-score is used to balance precision and recall, with higher values representing better classifier performance. However, gait identification is a multi-classification task and cannot directly use the F1-score. The simplest approach is to calculate the macro-F1 score. The macro-F1 score calculates the precision and recall for each category, and then calculates the average. The above evaluation metrics are shown in Eqs (12) to (16), where true positive, true negative, false positive, and false negative are represented by TP, TN, FP, and FN, respectively.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{12}$$

$$Precision = \frac{TP}{TP+FP} \tag{13}$$

$$Recall = \frac{TP}{TP+FN} \tag{14}$$

$$F1\ score = \frac{2\times Precision\times Recall}{Precision+Recall} \tag{15}$$

$$Macro\ F1\ score = \frac{1}{M}\sum_{i=1}^{M} F1\text{-}score_i \tag{16}$$

3.2. Performance comparisons of models for the time series data

In this experiment, the performance of the method proposed in this paper is evaluated with time series data. We conducted experiments on two datasets on a PC with an i5-8400 CPU and 8 GB RAM. In order to verify the effectiveness of our proposed method and that it is lightweight, we compared the performance of the gait recognition model using six methods. To ensure the consistency of the experiments, the optimizers were all changed to Adam, the learning rate was set to 0.001, and the number of epochs for training was 200. The experimental results for using different methods based on the HD Dataset are shown in Table 3.

Our results show that the existing gait recognition methods all have good recognition accuracy on the HD Dataset. This indicates that the different user's gait data collected using the accelerometer and gyroscope in the wearable device are distinguishable. In addition, to compare the difference between the last layer of the network using the FCL and the GAP layer, we constructed two gait recognition models based on the DSCNN module combined with FCL or GAP, respectively. The experimental results show that the model using GAP has a 2.08% higher recognition accuracy and a 99.6%

smaller number of parameters than FCL. Meanwhile, the accuracy and macro-F1 score obtained using the DSCNN model are higher than those obtained using the existing methods. Although the accuracy of the DSCNN model is 0.28% better than CNN+LSTM, the memory size is reduced by at least 36 M.

Table 3. Gait recognition performance evaluation using different methods.

Method	Accuracy	Macro-F1 score	Parameters Num	Memory Size
IdNet [11]	98.33%	0.9832	26,284	452 KB
CNN+LSTM [13]	98.61%	0.9865	4,716,406	56.7 MB
CNN+LSTM [14]	75.97%	0.7215	-	-
deep CNN [15]	98.33%	0.9836	3,013,704	34.8 MB
DSCNN(FCL)	96.81%	0.9695	4,208,246	45.8 MB
DSCNN(GAP)	98.89%	0.9894	15,566	532 KB

The performance of gait recognition based on the different models for Dataset #1 is also evaluated. We selected five methods [11,13–16] to compare with our method. The results are shown in Table 4. From the table, we can see that our DSCNN network achieved better recognition accuracy than the other methods used in studies [13–16]. In Dataset #1, our model achieved an accuracy of 94.44%. Compared to [13], our method improved the accuracy by 0.92%. In [14], a lightweight model called CNN+CEDS was proposed in order to reduce the complexity of the model. Although the recognition accuracy of our model is lower than that of CNN-CEDS, the memory size of our model is much lower, only 532 KB.

Table 4. Performance evaluation using different methods in Dataset #1.

Method	Accuracy	Macro-F1 score	Memory Size
CNN+LSTM [13]	93.52%	-	56.7 MB
CNN & LSTM [14]	94.15%	-	-
deep CNN [15]	90.64%	88.06%	36.4 MB
CNN-CEDS [16]	94.71%	93.98%	4.24 MB
DSCNN(GAP)	94.44%	93.42%	532 KB

3.3. Performance of the multimodal fusion method

In this part, we compare the performance of the DC-DSCNN model proposed in this paper with single modal and multimodal fusion. As described in Section 2.4, GAF images are generated in our method. A series of experiments are conducted to demonstrate the effectiveness of multimodal fusion based on DC-DSCNN in this section.

In order to evaluate the performance using GAF images, four cases were investigated: only using images extracted by acceleration, only using images extracted by angular velocity, using GAF images corresponding to acceleration and angular velocity as inputs for two channels, and GAF images with acceleration and angular velocity data fusion. The experimental results are shown in Table 5. The accuracy of using GAF images extracted by acceleration and angular velocity in the DSCNN model are 96.11 and 90.14%, respectively. The recognition performance is improved when GAF images of acceleration and angular velocity are input to two channels. However, the memory

size of the model is increased. The performance of the method with GAF images with fused acceleration and angular velocity data is better than other methods; the accuracy reaches 97.64%.

Table 5. Classification accuracy with GAF images of different input data.

Input Data	Accuracy
Acceleration	96.11%
Angular velocity	90.14%
Acceleration and Angular velocity	96.39%
Acceleration and Angular velocity fusion	97.64%

Therefore, we select the GAF images with fused acceleration and angular velocity data as the input to the DC-DSCNN model. We performed comparative experiments using only the time series of acceleration and angular velocity or only GAF images and using them both as inputs to DC-DSCNN. We can see from Table 6 that multimodal data fusion as input is better than the single modal. The accuracy and the macro-F1 score of multimodal fusion based on DC-DSCNN achieve at least 99.31 and 99.29%. Additionally, When the activation function of the ReLU is replaced by ELU, the accuracy is improved to 99.58%.

Table 6. Performance comparison of multimodal and singlemodal.

Model Input	Accuracy	Macro-F1 score
Time Series	98.89%	98.94%
GAF Image	97.64%	97.56%
Time Series + GAF (ReLU)	99.31%	99.29%
Time Series + GAF (ELU)	99.58%	99.52%

4. Conclusions

In this paper, we proposed a multimodal fusion gait recognition method based on DC-DSCNN. First, we proposed a gait cycle segmentation method combining peak detection and DTW and made the HD Dataset. Then, a DSCNN is proposed to extract features from the time series of acceleration and angular velocity. By comparing the DSCNN model with six other models, the experimental results show that our proposed method not only has 98.89% recognition accuracy, but also the model only occupies 532 KB of memory. In addition, in order to preserve the time dependence of the original time series, we adopted the GAF algorithm to convert the time series into two-dimensional format. We compared the model performance of GAF images using only acceleration and GAF images using only angular velocity, and using a combination of both. The results reveal that the GAF images combining acceleration and angular velocity have better recognition results, at least 1.53 and 7.5% higher than the other two. Finally, we proposed a multimodal fusion method that uses the time series and GAF images as inputs to the DC-DSCNN model which can be helpful in gait-based identification. The recognition accuracy on the HD Dataset can achieve 99.58% with a recognition time of 1.7 s. In this study, we classified the subjects' identities with gait data obtained from a wearable device. The proposed framework can

effectively recognize identity based on gait data. In the future, we intend to transfer the model to wearable devices for real-time gait recognition.

Acknowledgments

The authors would like to thank all the colleagues that have supported this work. This work is jointly supported by Natural Science Foundation of Hebei Province (No.F2021201002);Natural Science Foundation of Hebei Province (No.F2021201005);Science and Technology Project of Hebei Education Department (No.ZD2020146); Postdoctoral Scientific Research Project of Hebei Province (No.B2019005001); Key Research and Development Program of Baoding Science and Technology Bureau (No.1911Q001); The Program for Top 80 Innovative Talents in Colleges and Universities of Hebei Province (No.SLRC2017022); National Key Research and Development Program of China (No.2017YFB1401200).

Conflict of interest

The authors declare there is no conflict of interest.

References

1. J. Zhou, Z. F. Cao, X. L. Dong, X. D. Lin, Security and privacy in cloud-assisted wireless wearable communications: challenges, solutions and, future directions, *IEEE Wireless Commun.*, **22** (2015), 136–144. doi: 10.1109/MWC.2015.7096296.
2. K. Bayoumy, M. Gaber, A. Elshafeey, O. Mhaimeed, E. H. Dineen, F. A. Marvel, et al., Smart wearable devices in cardiovascular care: where we are and how to move forward, *Nat. Rev. Cardiol.,* **18** (2021), 581–599. doi: 10.1038/ s41569-021-00522-7.
3. R. Mungovan, Face recognition: fighting the fakes, *Biom. Technol. Today,* **2021** (2021), 5–7. doi: 10.1016/S0969-4765(21)00021-7.
4. G. Jeon, S. Lee, S. H. Lee, J. Shim, J. Ra, K. W. Park, et al., Highly sensitive active-matrix driven self-capacitive fingerprint sensor based on oxide thin film transistor, *Sci. Rep*., **9** (2019), 3216–3226. doi: 10.1038/s41598-019-40005-x.
5. M. Kumar, N. Singh, R. Kumar, S. Goel, K. Kumar, Gait recognition based on vision systems: A systematic survey, *J. Visual Commun. Image Representation*, **75** (2021), 103052–103064. doi: 10.1016/j.jvcir.2021.103052.
6. H. J. Ailisto, M. Lindholm, J. Mäntyjärvi, E. Vildjiounaite, S. Mäkelä, Identifying people from gait pattern with accelerometers, *Biom. Technol. Hum. Identif. II*., **5779** (2005), 7–14. doi: 10.1117/12.603331.
7. L. Rong, J. Zhou, M. Liu, X. Hou, A wearable acceleration sensor system for gait recognition, in *2007 2nd IEEE Conference on Industrial Electronics and Applications*., 2007. Available from: https://ieeexplore.ieee.org/document/4318894.
8. F. M. Sun, C. F. Mao, X. M. Fan, Y. Li, Accelerometer-based speed-adaptive gait authentication method for wearable IoT devices, *IEEE Int. Things. J.*, **6** (2018), 820–830. doi: 10.1109/JIOT.2018.2860592.

9. M. Ahmad, A. K. Bashir, A. M. Khan, M. Mazzara, S. Distefano, S. Sarfraz, Multi sensor-based implicit user identification, preprint, arXiv:1706.01739v3.
10. S. Choi, I. H. Youn, R. LeMay, S. Burns, J. H. Youn, Biometric gait recognition based on wireless acceleration sensor using k-nearest neighbor classification, *Int. Conf. Comput.*, 2014. Available from: https://ieeexplore.ieee.org/document/6785491.
11. M. Gadaleta, M. Rossi, IDNet: smartphone-based gait recognition with convolutional neural networks, *Pattern Recognit*., **74** (2018), 25–37. doi: 10.1016/j.patcog.2017.09.005.
12. R. Delgado-Escano, F. M. Castro, J. R. Cozar, M. J. Marin-Jimenez, N. Guil, An end-to-end multi-task and fusion CNN for inertial-based gait recognition, *IEEE Access*., **7** (2018), 1897–1908. doi: 10.1109/ACCESS.2018.2886899.
13. Q. Zou, Y. L. Wang, Q. Wang, Y. Zhao, Q. Q. Li, Deep learning-based gait recognition using smartphones in the wild, *IEEE Trans. Inf. Forensics Secur*., **15** (2020), 3197–3212. doi: 10.1109/TIFS.2020.2985628.
14. L. Tran, T. Hoang, T. Nguyen, H. Kim, D. Choi, Multi-model long short-term memory network for gait recognition using window-based data segment, *IEEE Access*., **9** (2021), 23826–23839. doi: 10.1109/ACCESS.2021.3056880.
15. A. I. Middya, S. Roy, S. Mandal, R. Talukdar, Privacy protected user identification using deep learning for smartphone-based participatory sensing applications, *Neural Comput. Appl*., **33** (2021), 17303–17313. doi: 10.1007/s00521-021-06319-6.
16. H. H. Huang, P. Zhou, Y. Li, F. M. Sun, A lightweight attention-based CNN model for efficient gait recognition with wearable IMU sensors, *Sensors*, **21** (2021), 2866–2879. doi: 10.3390/s21082866.
17. M. Paulich, M. Schepers, N. Rudigkeit, G. Bellusci, Xsens MTw Awinda: miniature wireless inertial-magnetic motion tracker for highly accurate 3D kinematic applications, *XSens: Enschede*, The Netherlands, (2018), 1–9. doi: 10.13140/RG.2.2.23576.49929.
18. L. F. Mo, L. J. Zeng, Running gait pattern recognition based on cross-correlation analysis of single acceleration sensor, *Math. Biosci. Eng*., **16** (2019). 6242–6256. doi: 10.3934/mbe.2019311.
19. B. Auvinet, G. Berrut, C. Touzard, L. Moutel, N. Collet, D. Chaleil, et al., Reference data for normal subjects obtained with an accelerometric device, *Gait Posture*., **16** (2002), 124–134. doi: 10.1016/S0966-6362(01)00203-X.
20. H. Prasanth, M. Caban, U. Keller, G. Courtine, A. Ijspeert, H. Vallery, et al., Wearable sensor-based real-time gait detection: a systematic review, *Sensors*, **21** (2021), 2727–2755. doi: 10.3390/s21082727.
21. M. Muller, Dynamic time warping, *Information Retrieval for Music and Motion*, (2007), 69–84. doi: 10.1007/978-3-540-74048-3_4.
22. Z. G. Wang, T. Oates, Imaging time-series to improve classification and imputation, in *Proceeding of 24th International Joint Conference on Artificial Intelligence*, preprint, arXiv:1506.00327v1.
23. A. G. Howard, M. L. Zhu, B. Chen, D. Kalenichenko, W. J. Wang, T. Weyand, et al., MobileNets: efficient convolutional neural networks for mobile vision applications, preprint, arXiv:1704.04861v1.
24. F. Chollet, Xception: deep learning with depthwise separable convolutions, in *2017 IEEE CVPR*, Honolulu, HI, USA, (2017), 1800–1807. doi: 10.1109/CVPR.2017.195.
25. M. N. Chong, Q. M. Li, J. Li., Parameter estimation via deep learning for camera localization, *IOP Conf. Ser.: Mater. Sci. Eng.*, **569** (2019). doi: 10.1088/1757-899X/569/5/052101.

26. S. H. Wang, Z. Q. Zhu, Y. D. Zhang, PSCNN: PatchShuffle convolutional neural network for COVID-19 explainable diagnosis, *Front. Public Health*, **9** (2021), 768278–768304. doi: 10.3389/fpubh.2021.768278.
27. D. A. Clevert, T. Unterthiner, S. Hochreiter, Fast and accurate deep network learning by exponential linear units (ELUs), preprint, arXiv:1511.07289v5.
28. V. Nair, G. E. Hinton, Rectified linear units improve restricted boltzmann machines, in *Proceedings of the 27th International Conference on Machine Learning*, Haifa, Israel, (2010). 807–814.

AIMS Press