Hindawi
Scientific Programming
Volume 2022, Article ID 6355959, 11 pages
https://doi.org/10.1155/2022/6355959

Research Article

Dance Art Scene Classification Based on Convolutional Neural Networks

Le Li

College of Music, University of Sanya, Sanya 572000, China

Correspondence should be addressed to Le Li; 16461071021@stu.wzu.edu.cn

Received 24 May 2022; Revised 17 June 2022; Accepted 22 June 2022; Published 8 July 2022

Academic Editor: Lianhui Li

Digital multimedia resources have become an important part of people's daily cultural life. Automatic scene classification of a large number of dance art videos is the basis for scene semantic based video content retrieval. In order to improve the accuracy of scene classification, the videos are identified using a deep convolutional neural network based on differential evolution for dance art videos. First, the Canny operator is used in YCbCr colour space to detect the human silhouette in the key frames of the video. Then, the AdaBoost algorithm based on cascade structure is used to implement human target tracking and labelling, and the construction and updating of weak classifiers are analysed. Next, a differential evolution algorithm is used to optimise the structural parameters of the convolutional neural network, and an adaptive strategy is adopted for the scaling factor of the differential evolution algorithm to improve the optimisation solution accuracy. Finally, the improved deep convolutional neural network is used to train the classification of the labelled videos in order to obtain stable scene classification results. The experimental results show that by reasonably setting the crossover rate of differential evolution and the convolutional kernel size of the convolutional neural network, high scene classification performance can be obtained. The high accuracy and low root-mean-square error validate the applicability of the proposed method in dance art scene classification.

1. Introduction

Due to the accelerated pace of life today, many people are busy with work, resulting in not having enough time to rest. People want to enrich their spare time activities more in their leisure time, such as dancing. However, as dance learning usually requires attending professional classes offline, resulting in many people not having much time to learn. Therefore, it has become a trend to learn dance by searching and watching online videos [1–7]. The main means of expression in the art of dance is the flexible footwork and graceful movement of the human body. Dance expresses feelings and reflects social life through this art form. As society continues to develop, people's demand for quality of life continues to increase. The traditional offline dance learning method is no longer able to meet people's needs, and is still very limited. As more and more people want to learn dance, which creates the problem of limited resources in teaching dance teachers, the original face-to-face teaching method can no longer meet the actual needs. Online teaching has become more and more accepted and has become a new mode of teaching.

The online teaching and learning process requires students to access the knowledge they learn through the computer network. As online teaching not only maximises the sharing of information resources, it can also generate multiple forms of teaching and learning, helping to improve teaching efficiency and achieve sharing of teaching resources. At present, there is an explosive growth in the use of digital multimedia video. Along with the popularity of the Internet, massive amounts of dance art video data have appeared on various online media [8–10]. According to relevant statistical reports, online media around the world generate about tens of T of video data every day, which contains video data of movies, music and dance.

With so much dance video data available, only a fraction of it is of interest to each individual. So, how can one find the data one needs from this dance video data? This requires

effective scene classification of this dance video data. Classifying scenes from a large number of dance videos can be a very difficult task. The traditional method is to annotate and classify these videos manually, thus, forming a database of dance videos that can be indexed by keywords. However, with the huge amount of dance video data, it would take a lot of human resources, money, and time to use the manual annotation method [11–13]. This manual approach requires staff to face a large amount of dance video data every day, which is prone to visual errors, thus, leading to errors in video annotation and classification. Therefore, this traditional method has major drawbacks [14]. An alternative approach is to use computers to analyse these massive amounts of video data and eventually achieve an automated dance video scene classification system. In using computer technology to annotate, classify, and retrieve dance videos, an efficient algorithm needs to be designed to process them [15]. In recent years, the issues of video annotation, video scene classification, and video retrieval have become a hot research topic in the multimedia field. Numerous scholars and research institutions have conducted in-depth research on this problem.

Traditional video scene classification methods generally use manually designed features for modelling. Wei et al. [16] proposed a motion human tracking algorithm based on region segmentation contours with more accurate and stable performance in complex occlusion situations. Suganya et al. [17] proposed an AdaBoost-STC and random forest based human eye tracking and localisation algorithm. Wang et al. [18] proposed a target tracker based on likelihood graph and real-time AdaBoost cascade. Both methods are effective in improving tracking speed without degrading tracking accuracy. Ibrahim et al. [19] conducted a video classification study using video saliency features. They divided the RGB colour channel of each frame into three images, and then combined the grey-scale images to arrange these three images in temporal order to obtain three spatio-temporal container models. These spatio-temporal containers were then subjected to pyramidal degradation and the regions of significance in the containers were divided using mean clustering. Finally, a support vector machine is used to classify the video scenes. This algorithm has a more complex process and is not effective in video scene classification. Calvin et al. [20] mapped motion vectors into the unit circle and divided it into 8 regions. Each motion vector is mapped to the coordinate axes of the corresponding region, and the corresponding matrix is derived as features for the data on the axes. Finally, the SVM is used to classify the video. However, this method can only detect the corresponding moments taken as features, and finally, the video is classified using SVM. However, this method can only detect some motion patterns in the video, such as jumping, running, swimming, and some other specific events, and cannot determine the scene classification of the video. Lu et al. [21] classify the video by taking the comparative values of luminance between regions in the video as featured, and by using Hidden Markov Model (HMM). This method is able to eliminate the influence of factors such as illumination on the video, but can only perform the classification of different categories of videos, such as news, movie, and animation videos. In addition, the calculation of the parameters of the HMM requires a large number of videos for training, and the whole process is more complicated.

Semantic-based information processing has developed rapidly in recent years with the development of artificial intelligence and data mining techniques. Many researchers are conducting research in mapping from the underlying features of the video to the semantic information of the video. By mining the semantic information of videos and forming semantic rules according to certain algorithms, scene classification of video data can be achieved. Therefore, the use of semantic information to classify video is also a future trend in video classification. Deep learning abandons the complex operation process of the underlying features in the traditional algorithm, so it can effectively achieve the task of video semantic information mining based on computer vision. Convolutional neural network (CNN) [22–24], which emerged in the field of deep learning, first achieved great success in image recognition and image segmentation. Then, breakthroughs in typical network structures continued, such as recurrent neural network (RNN) [25, 26], deep belief network (DBN) [27], generative adversarial networks (GAN) [28], and other types of network structures. These network structures are capable of enhancing the feature extraction capability of models in a supervised learning manner. Compared to traditional machine learning methods, deep neural networks perform feature extraction at different scales on images, combining gradients to explore better strategies, and saving the tedious manual feature extraction process. As a result, deep neural networks only require a well-designed network structure. With the excellent image feature representation capability, deep neural networks have good robustness in dealing with scene classification problems of sports, news, and other videos. However, dance art videos are more diverse and involve human target tracking and labelling problems, so, the various types of network structures available in deep learning do not perform well enough for the scene classification task of dance videos.

The aim of this study is to automatically classify scenes from dance videos using deep convolutional neural networks and to further improve the accuracy of the model through structural parameter optimisation. The proposed method helps to implement a video content retrieval task based on scene semantics.

Key innovations and contributions to the video include the following:

(1) Both the contour model and the AdaBoost algorithm show some advantages in terms of robustness and accuracy of video target tracking. Therefore, a combination of both is proposed to solve the person tracking problem in dance art videos.

(2) A deep CNN based on differential evolution (DE) [29] was proposed to address the problem of unsatisfactory classification efficiency and stability of the traditional CNN structure in processing the classification of dance video scenes based on

semantic information. The DE algorithm was introduced to optimally solve the network parameters, and an adaptive strategy was adopted for the scaling factor of the DE algorithm to improve the accuracy of the optimal solution.

The rest of the paper is organized as follows: in Section 2, the target detection based on human silhouette model in dance video isstudied in detail, while Section 3 provides the human tracking based on the cascade structure AdaBoost algorithm. Section 4 provides the DE-CNN based dance video scene classification. Section 5 provides the experimental results and analysis. Finally, the paper is concluded in Section 6.

2. Target Detection Based on Human Silhouette Model in Dance Videos

In dance video data, the human body often rotates, translates, and stretches. However, the detection of human targets becomes difficult, when the body's pose is constantly changing. Therefore, the video uses a statistical learning model based on a human contour model to implement human body detection.

2.1. YCbCr Colour Model. First, the RGB colour model is converted into a YCbCr colour model in the 3D colour space, which is mainly used for edge detection and image segmentation in the digital video field, and its colour cube diagram is shown in Figure 1.

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112 \\ 112 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad (1)$$

where Y is the intensity information, Cb and Cr are the colour difference components of the colour image.

2.2. Canny Operator Edge Detection. The human body image is preprocessed by edge detection, in order to extract the human contour features. The first order derivative of the two-dimensional function $f(x, y)$ of the human body image is expressed as follows:

$$\nabla f(x, y) = \begin{bmatrix} \dfrac{\partial f}{\partial x} \\ \dfrac{\partial f}{\partial y} \end{bmatrix}. \quad (2)$$

The second order derivative of a two-dimensional function $f(x, y)$ is expressed as follows:

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2}. \quad (3)$$

The luminance region can be divided by finding the pixel points that satisfy $\nabla^2 f(x, y) = 0$. The video uses the Canny operator [30] to implement human edge detection. The edge direction of each pixel point is calculated by equation (4).

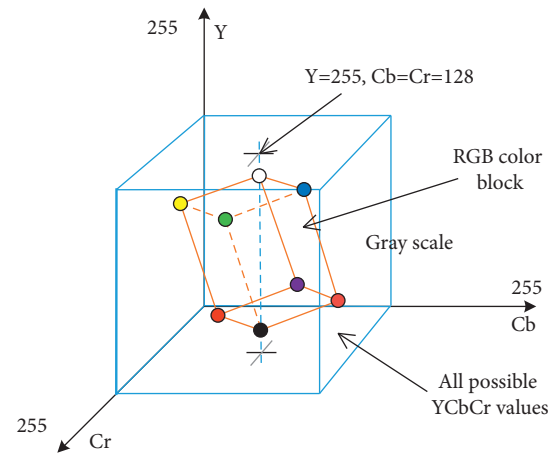

Figure 1: YCbCr colour model.

$$\alpha(x, y) = arc\ \tan\begin{bmatrix} \dfrac{\partial f}{\partial y} \\ \dfrac{\partial f}{\partial x} \end{bmatrix}. \quad (4)$$

The pixel point with the maximum pixel gradient is set as the edge pixel point. The pixel gradient is calculated as shown follows:

$$g(x, y) = \frac{1}{(\partial f/\partial x)^2 + (\partial f/\partial y)^2}. \quad (5)$$

3. Human Tracking Based on the Cascade Structure AdaBoost Algorithm

Recently, integrated methods like AdaBoost (adaptive boosting) have been successfully applied to many target tracking problems. AdaBoost classifier is a meta-algorithmic classifier and utilises the same base classifier (weak classifier). Based on the error rate of the classifier, the AdaBoost classifier can be assigned different weighting parameters. Finally, the integrated classifier outputs predictions occur by means of a summation operation of the weights.

3.1. Construction and Updating of Weak Classifiers. For each pixel in each image frame, the weak classifier is defined as follows:

$$h(x) = \text{sign}\left(h^T x\right), \quad (6)$$

where x is the sample and h is the adjusted segmentation hyperplane, calculated as follows:

$$h = \left(\mathbf{A}^T \mathbf{W} \mathbf{A}\right)^{-1} \mathbf{A}^T \mathbf{W} y, \quad (7)$$

where y is the sample label, $\mathbf{A}$ is a matrix, and $\mathbf{W}$ is a diagonal matrix of weights. The sample weights are updated as follows:

$$D_i = D_i e^{\alpha_t |h_t(x_i - y_i)}. \tag{8}$$

3.2. Description of the Algorithm Flow. The AdaBoost algorithm is an iterative algorithm. AdaBoost can aggregate multiple weak classifiers from the same training set to form a strong classifier. The main steps of the AdaBoost algorithm are shown as follows:

Step 1. Set the input be $\{(x_1, y_2), (x_2, y_2) \ldots (x_n, y_n)\}$ where, $x_i \in X$, $y_i \in \{-1, 1\}$ and the data set be X. Initialize the weights $D_1(i)$ is shown as follows:

$$D_1(i) = \frac{1}{n}, \quad i = 1, 2 \ldots n.t \tag{9}$$

Step 2. Find the weak classifier h_t: $X \longrightarrow \{-1, 1\}$, when $t = 1, 2 \ldots T$ and train a weak classifier h_j with each feature f_j, which gives a weighted error rate.

$$\epsilon_j = \sum_{t=1}^{n} D_t, h_t(x_i) \neq y_i. \tag{10}$$

Step 3. The classifier h_t with the smallest weighted error rate ϵ_j is selected and its smallest weighted error rate value is noted as ϵ_t. The weights of the weak classifier are then calculated as follows:

$$\alpha_t = \frac{1}{2}\ln\left(\frac{1 - \epsilon_t}{\epsilon_t}\right). \tag{11}$$

Step 4. The actual method used to update the sample weights is shown as follows:

$$D_{t+1}(i) = \frac{D_t(i)\exp[-\alpha_t y_i h_t(x_i)]}{Z_t}. \tag{12}$$

where Z_t denotes the normalisation parameter.

Step 5. Construct the final strong classifier using the following approach.

$$H(x) = \operatorname{sign}\left(\sum_{t=1}^{T} \alpha_t y_i h_t(x)\right). \tag{13}$$

4. DE-CNN Based Dance Video Scene Classification

4.1. Adaptive DE Algorithm. Let the population size be N, the attribute dimension be D, the differential scaling factor be F, the crossover rate be CR, and the value of each individual be $[U_{\min}, U_{\max}]$, then the j dimensional attribute [31] of the i-th individual can be shown as follows:

$$x_{ij} = U_{\min} + \text{rand} \times (U_{\max} - U_{\min}), \tag{14}$$

where $i = 1, 2, \cdots, N$, $j = 1, 2, \cdots, D$, rand are random numbers in (0, 1).

Individuals x_i^G, $(i = 1, 2, \ldots, N)$ of the G generation can obtain the $G + 1$ generation using the mutation operation.

$$v_i^{G+1} = x_{r_1}^G + F \times \left(x_{r_2}^G - x_{r_3}^G\right), \tag{15}$$

where r_1, r_2, and r_3 are three random individuals from the G generation. A common range of F values is $[0, 2]$.

The individual crossover method is shown as follows:

$$u_{ij}^{G+1} = \begin{cases} v_{ij}^{G+1}, & \text{rand}(0, 1) \leq CR, \\ x_{ij}^G, & \text{otherwise.} \end{cases} \tag{16}$$

Compare x_i^G with u_i^{G+1} and find the fitness value of each individual. Select the individual with the higher fitness value for the subsequent evolutionary process.

$$x_i^{G+1} = \begin{cases} u_i^{G+1}, f\left(u_i^{G+1}\right) > f\left(x_i^G\right), \\ x_i^G, f\left(u_i^{G+1}\right) \leq f\left(x_i^G\right), \end{cases} \tag{17}$$

where f represents the fitness function. The DE algorithm stops, when the maximum number of generations $G_{\max}$ is reached.

A common range of F values is $[0, 2]$. The optimisation process for DE is closely related to the F value. A wrong choice of F value will result in unsatisfactory optimisation performance of the differential evolution algorithm. Therefore, adaptive F values are introduced in the calculation. The value range of $F_{\min}$ and $F_{\max}$ is $[0, 2]$.

$$F = F_{\min} + \left(F_{\max} - F_{\min}\right) \times e^{1 - G_{\max}/G_{\max} - G + 1}. \tag{18}$$

The F value becomes progressively smaller as the evolutionary generation G changes. Early evolution pursues population diversification, while late evolution focuses on search ability, so that the DE algorithm is more likely to obtain optimal individuals.

4.2. CNN Model Design. Machine learning has played a huge role in computer vision processing techniques. Most of the traditional machine learning methods use shallow structures that deal with limited data operations. A large number of experiments have proven that the feature expressions learned from shallow structures, when dealing with complex classification problems, are difficult to meet the practical needs. In recent years, computer performance has continued to improve, providing a powerful support for deep learning. New deep learning models are constantly being proposed and successfully incorporated into application areas such as image recognition, speech recognition, and natural language processing.

Common deep learning models in image recognition include deep belief network (DBN), recurrent neural network (RNN), generative adversarial network (GAN), capsule network (CapsNet), restricted boltzmann machines (RBMs), and convolutional neural network (CNN). Based on the deep

convolution neural network, this paper selects the most representative dance video as the recognition object.

Originally designed, specifically to handle image recognition tasks, CNNs are multilayer neural networks and are currently the most classical and commonly used computational structure in the field of computer vision. The basic structure of a CNN consists of an input layer, an implicit layer and an output layer. The implicit layer is the core part of the convolutional neural network, which contains the convolutional layer, the pooling layer (also known as the downsampling layer), and the fully connected layer, as shown in Figure 2.

Pooling layers generally reduce the dimensionality of the input feature map between successive convolutional layers. The pooling layer effectively reduces the output feature vector of the convolution layer. This process uses a partially contiguous region of the image as the pooling region and translates the sliding window matrix of the pooling function within the region. The pooling size and step size control the sliding window size and translation rule respectively, as shown in Figure 3.

Let the set of dance video samples be $\mathbf{X} = (x_1, x_2, \ldots x_N)$. The m video attribute features are convolved through the l layer.

$$x_{lj} = f\left(\sum_{j\in m} x_{l-1} * k_{lj} + b_{lj}\right), \tag{19}$$

where k_{lj} and b_{lj} represent the weights and biases assigned to the features j by the l layer, respectively, and $*$ is the convolution.

$$f(z) = \frac{1}{1+e^{-z}}. \tag{20}$$

Convolution of m features is from N samples. Convolution kernel size $h \times w$:

$$g(x) = \max_{1\le k\le h\times w}(x_k). \tag{21}$$

Assuming $M = N/(h \times w)$, then the original sample $\mathbf{X} = (x_1, x_2, \ldots x_N)$ is reconstructed after convolution pooling as $\mathbf{X}' = (x_1, x_2, \ldots x_M)$. The conversion operation is then performed on $\mathbf{X}'$.

$$x_j^l = f\left(\sum_{i=1}^{M} a_{ij}\left(x_i^{l-1} * k_i^l\right) + b_j^l\right). \tag{22}$$

The restrictions are $\sum a_{ij} = 1, 0 \le a_{ij} \le 1$.

After obtaining all the connected layers of the CNN, the classifier is selected to predict the sample class. Let the training output and the actual value of the k-th node be y_k and d_k, respectively, and the error term be δ_k.

$$\delta_k = (d_k - y_k) y_k (1 - y_k). \tag{23}$$

Assuming that the l and $l+1$ layers contain L and P nodes, respectively, the error of node j in the l layer is δ_j.

$$\delta_j = h_j\left(1 - h_j\right)\sum_{k=1}^{P} \delta_k W_{jk}, \tag{24}$$

where h_j is the output and W_{jk} is the weight of the neuron j to the neuron k in the $l+1$ layer. The weights are updated as shown follows:

$$\Delta w_{jk}(n) = \frac{\eta}{1+N}\left(\Delta w_{jk}(n-1) + 1\right)\delta_k h_j, \tag{25}$$

where η is the learning rate.

The bias $\Delta b_k(n)$ is updated as follows:

$$\Delta b_k(n) = \frac{\alpha}{1+N}\left(\Delta b_k(n-1) + 1\right)\delta_k, \tag{26}$$

where α is the bias update step, typically $\alpha = 1$. The adjusted weights are shown as follows:

$$w_{jk}(n+1) = w_{jk}(n) + \Delta w_{jk}(n). \tag{27}$$

The adjusted offsets are shown as follows:

$$b_k(n+1) = b_k(n) + \Delta b_k(n). \tag{28}$$

The error for all nodes E is shown as follows:

$$E = \frac{1}{2}\sum_{k=1}^{M}(d_k - y_k)^2. \tag{29}$$

When E meets the set threshold, the iteration stops and a stable CNN model is obtained.

4.3. Classification Process Based on DE-CNN Model. Before the CNN can be applied to classify a video, the sample data to be classified first needs to be transformed, which is mainly to address the vectorisation process of the video attributes. The converted Skip-gram facilitates efficient input to the CNN. After the CNN video classification model is established, the random weights and biases are optimally solved by the DE algorithm. An adaptation function is established based on the video classification accuracy function. The optimal individuals of weights and biases are obtained by multigeneration evolution of DE. Finally, the video classification results are obtained using CNN for classification training, as shown in Figure 4.

5. Experimental Results and Analysis

5.1. Experimental Setup. In order to validate the performance of the DE-CNN model in dance video scene classification, simulation experiments were conducted on dance video sequences (resolution 640×480), with the length of 400 frames. Firstly, the performance of human target tracking was verified. Secondly, the performance was verified for different DE algorithm parameters. Then, the performance was verified for different convolutional kernel sizes. Finally, the performance of the DE-CNN model is compared with commonly used video scene classification algorithms.

The data sources for the video classification experiment were 11 large video websites. All videos were in MP4 format, and seven categories of dance videos were selected for the classification test: classical dance, ballet, folk dance, modern dance, tap dance, jazz dance, and Latin dance. The number of videos in each category is 500, so there are 3500 dance video sequences in the experimental video dataset. The

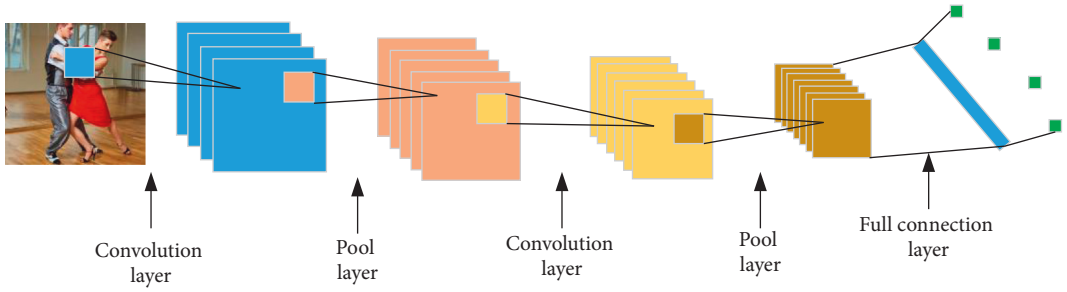

Figure 2: CNN network structure.

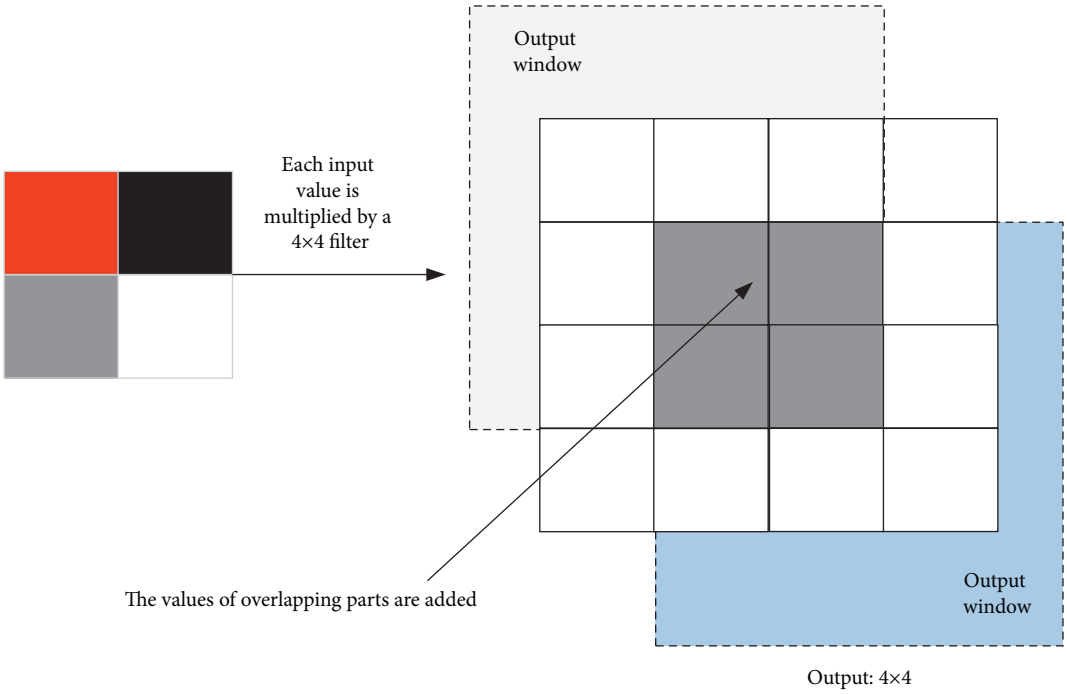

Figure 3: Pooling feature diagram.

length of each video sequence was 400 frames and the duration was 5 min. Some of the data of the dance video samples are shown in Figure 5.

The proposed method classifies the dance videos so that automatic scene recognition can be achieved. Information on the experimental video dataset is shown in Table 1.

The video from Table 1 was transformed using the Skip-gram structure, thus completing the video-to-attribute vector mapping. This allowed the video samples to be trained for CNN classification. During the experiments, the entire dance video sample set was trained and tested in a 7 : 3 ratio respectively. The experimental hardware environment is: CPU i7 3770 (3.4 Hz), 8 G RAM. The experimental software environment is: Windows 10 operating system, Matlab 7.0 simulation software. The initial values of DE algorithm settings are $F_{\min} = 0.2$, $F_{\max} = 0.9$, $CR = 0.1$, and $G_{\max} = 100$. CNN convolutional kernels are $2 * 2$ by default.

5.2. Human Target Tracking Performance. The effect of human target detection was first quantified in order to assess its robustness. The panning errors for human detection are shown in Figure 6. As can be seen from Figure 6, the human detection is good in the panning case with an average error of less than 10 pixels.

In addition, in order to quantitatively compare the tracking performance, the comparison experiments of the same video sequences are conducted by using hybrid algorithm, AdaBoost-STC algorithm, and adaptive EKF algorithm.

$$d = \sqrt{\sum_{i=1}^{N} (x_i - y_i)^2}, \tag{30}$$

where x_i is the centre of the trace result and y_i is the centre of the baseline result.

After repeating the experiment 100 times and taking statistical averages, the human tracking results for the three different algorithms on a 400-frame video sequence are shown in Figure 7.

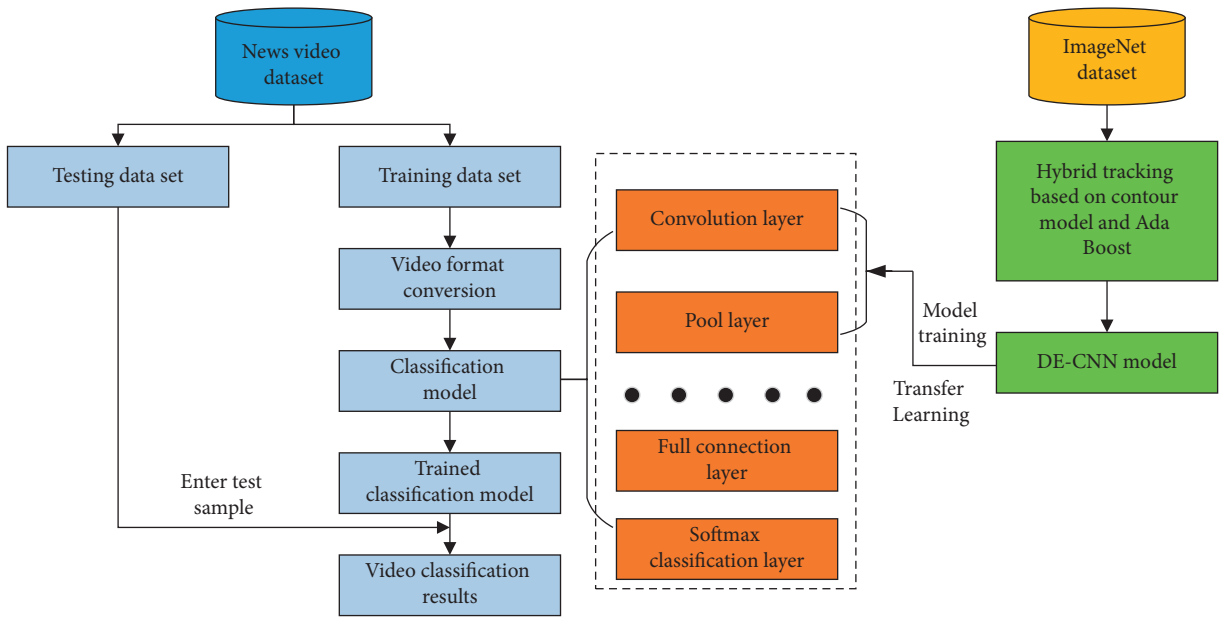

Figure 4: Flow of dance video scene classification based on DE-CNN model.

As can be seen in Figure 7, the difference in centroid pixel error between the three different algorithms is not very significant until 250 frames. However, as the tracking time increases, the hybrid tracking algorithm based on the contour model and AdaBoost shows a stronger advantage when it exceeds 250 frames. In other words, the hybrid tracking algorithms based on the contour model and AdaBoost are better in terms of stability and robustness under the same conditions.

5.3. Video Classification Performance with Different Convolution Sizes. CNN structures with different kernel sizes were used to test the experimental samples separately, and the results are shown in Table 2.

From Table 2, the best results were obtained when the convolutional kernel size of $3 * 3$ was chosen, and the classification accuracy of the dance video data samples came to 92.16%. When the size increases, the classification accuracy and standard deviation are decreasing. This is because the convolution size is too large, resulting in a larger convolutional granularity, which reduces the opportunity for the important attributes of the samples to participate in the convolution and transformation operations. The temporal performance of the DE-CNN algorithm on the dance video dataset, when the convolutional kernel size is $3 * 3$ is shown in Figure 8.

As can be seen from Figure 8, the classification time of the DE-CNN model was about 55 s at a convolutional kernel size of $3 * 3$. Ultimately, the classification accuracy of the DE-CNN model at convergence was all over 0.9.

5.4. Optimisation Performance of the DE Algorithm. In order to verify the optimisation performance of the DE algorithm for CNN, the performance of the test samples was simulated using the CNN algorithm and the DE-CNN algorithm, respectively.

As can be seen from Table 3, the DE-CNN algorithm showed better performance in the classification of dance video scenes. All three metrics of DE-CNN video classification exceeded 0.9. The maximum classification accuracy of DE-CNN was 93.18%, while the maximum classification accuracy of CNN was only 88.96%, so the accuracy of DE-CNN was significantly improved. This is mainly due to the fact that after weight optimisation by DE, the CNN obtains better weights and bias initial values, resulting in a more accurate video classification performance. The comparison of the convergence performance of the two algorithms will be continued below, as shown in Figure 9.

It can be seen that the convergence performance of DE-CNN is significantly superior compared to CNN. In the classification of dance video data samples, DE-CNN converges with an RMSE of about 0.18, while CNN converges with an RMSE value of about 2.5. Therefore, the DE-CNN algorithm has better classification stability compared to the CNN algorithm. In terms of convergence time, the CNN converges in about 5 s less than the DE-CNN. This may be due to the longer time taken by the DE algorithm to solve for the optimal weights and biases. However, in terms of the overall DE-CNN classification time, the DE algorithm consumes a small percentage of the time and has less impact on the video classification time.

5.5. Video Classification Performance of Different Algorithms. The commonly used plain Bayesian (NB) [32], BP neural network [33], LSTM neural network [34], and DE-CNN were used to compare and analyse the test dataset respectively, as shown in Figure 10.

In terms of classification accuracy of the videos, DE-CNN and LSTM algorithms have the highest classification accuracies. In terms of classification time, the LSTM

(a) (b) (c) (d) (e) (f)

Figure 5: Partial data presentation of the dance video sample. (a) Classical dance. (b) Ballet dance. (c) Folk dance. (d) Tap dance. (e) Jazz dance. (f) Latin dance.

Table 1: Information on the experimental video dataset.

Dance video category	Number	Video sources
Classical dance	500	Google videos, Baidu videos
Ballet	500	CCTV, movies, Microblog, Facebook
Folk dance	500	Youku App, Twitter, MetaCafe
Contemporary dance	500	Netflix, LiveLeak
Tap dance	500	Microblog, Google videos
Jazz dance	500	Facebook, LiveLeak
Latin dance	500	CCTV, Baidu videos, Twitter

algorithm consumes the longest time, followed by the DE-CNN algorithm, and the NB algorithm the least time.

The following continues to test the stability of the 4 algorithms in video scene classification. The RMSE performance of the 4 algorithms was verified and is shown in Figure 11.

It can be seen that the DE-CNN algorithm has the best RMSE values and the NB performs the worst. This also indicates that the classification RMSE values are more sensitive to the number of video categories. In summary, for scene classification of 3500 dance video sequences, the DE-CNN model still achieves good classification time and

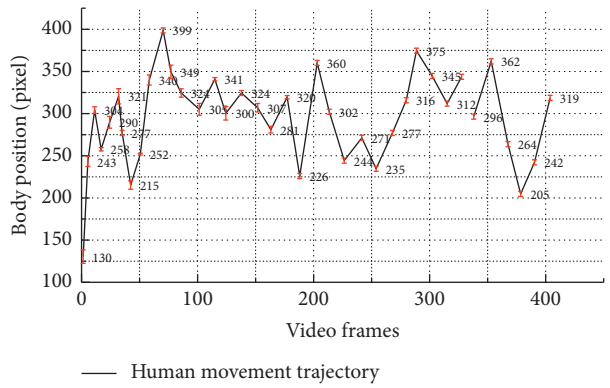

Figure 6: Translation error.

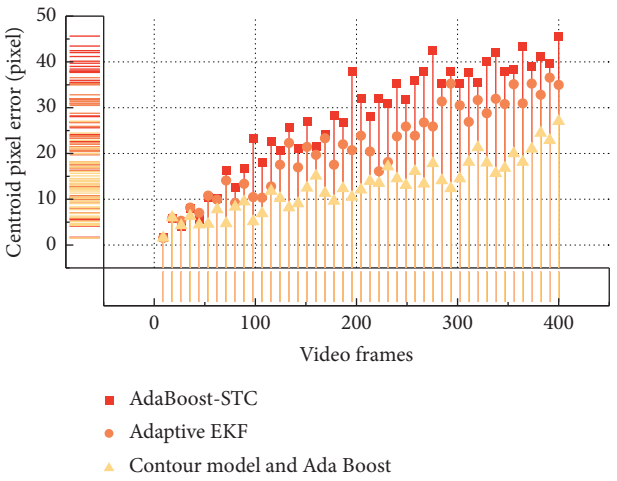

Figure 7: Tracking result of three algorithms.

Table 2: Classification accuracy.

Convolution kernel size	Number of categories	Accuracy	RMSE
2 * 2	7	0.9164	0.1862
3 * 3	7	0.9216	0.1847
4 * 4	7	85.1937	0.2334
5 * 5	7	68.6171	0.5219

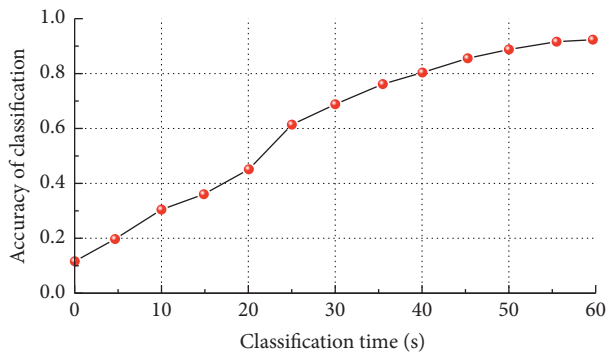

Figure 8: Classification accuracy (convolution kernel 3 * 3).

Table 3: Classification performance of CNN and DE-CNN algorithms.

Algorithms	Accuracy	Recall rate	F1 value
CNN	0.8646	0.8473	0.8064
DE-CNN	0.9275	0.9014	0.9012

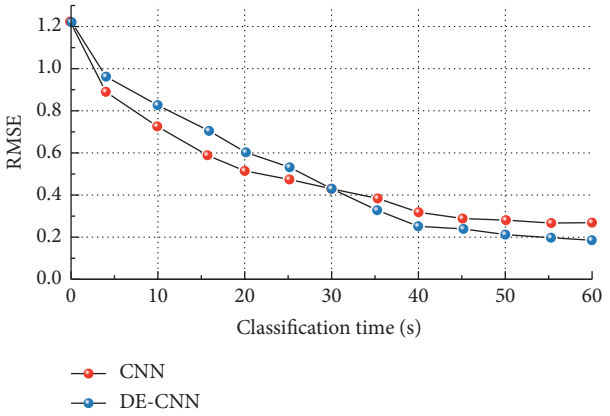

Figure 9: RMSE values of the two algorithms.

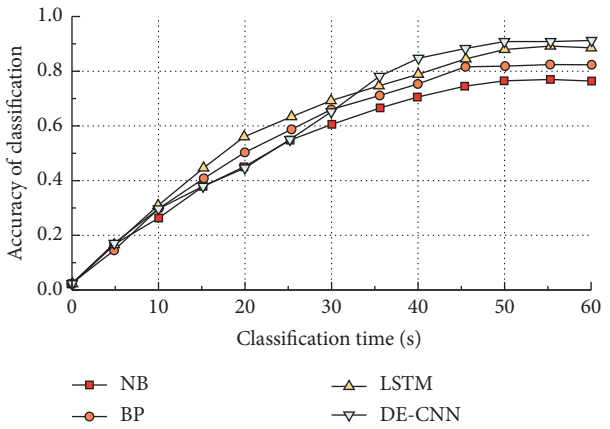

Figure 10: Classification accuracy of four algorithms.

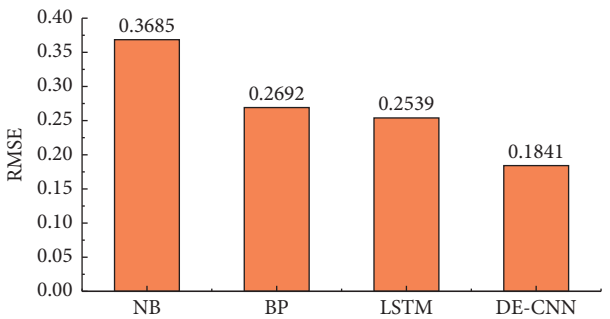

Figure 11: RMSE performance of different algorithms.

RMSE performance under the condition of obtaining high classification accuracy.

6. Conclusion

In this paper, a differential evolutionary convolutional neural network model is applied to scene classification of dance videos. A contour model-based detection approach is used to achieve human target detection, which effectively improves the robustness of human detection. The AdaBoost algorithm based on cascade structure is used to achieve human target tracking. The weight optimisation solution advantage of the differential evolution algorithm is used to improve the applicability of the convolutional neural network model in video scene classification. The following conclusions are drawn.

(1) The average error in human motion detection is less than 10 pixels, which indicates higher robustness.

(2) The proposed method has a smaller pixel error in the centroid of human movement than other methods and is suitable for a long tracking process.

(3) Compared with commonly used video classification algorithms, the proposed DE-CNN model has significant advantages in terms of classification accuracy and RMSE performance. Subsequent studies will further tune the differential evolution parameters to improve the video scene classification time performance.

Data Availability

The experimental data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

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

References

[1] G. Gao, W. Zhang, Y. Wen, Z. Wang, and W. Zhu, "Towards cost-efficient video transcoding in media cloud: insights learned from user viewing patterns," *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1286–1296, 2015.

[2] S. Norazean, M. A. Mazli, and G. Faizul, "Students' perceptions on using different listening assessment methods: audio-only and video media," *English Language Teaching*, vol. 10, no. 8, pp. 93–97, 2017.

[3] K. K. Loh, B. Tan, and S. Lim, "Media multitasking predicts video-recorded lecture learning performance through mind wandering tendencies," *Computers in Human Behavior*, vol. 63, pp. 943–947, 2016.

[4] J. Adams, G. Christian, and T. Tarshis, "Managing media: reflections on media and video game use from a therapeutic perspective," *Journal of the American Academy of Child & Adolescent Psychiatry*, vol. 54, no. 5, pp. 341-342, 2015.

[5] I. Dewi and W. A. Ni, "The positive impact of teams games tournament learning model assisted with video media on students' mathematics learning outcomes," *Journal of Education Technology*, vol. 4, no. 3, pp. 367–371, 2020.

[6] Y. Hao, T. Mu, R. Hong, M. Wang, N. An, and J. Y. Goulermas, "Stochastic multiview hashing for large-scale near-duplicate video retrieval," *IEEE Transactions on Multimedia*, vol. 19, no. 1, pp. 1–14, 2017.

[7] R. Fernandez-Beltran and F. Pla, "Latent topics-based relevance feedback for video retrieval," *Pattern Recognition*, vol. 51, pp. 72–84, 2016.

[8] Y. Zhu, X. Huang, Q. Huang, and Q. Tian, "Large-scale video copy retrieval with temporal-concentration SIFT," *Neurocomputing*, vol. 187, no. 4, pp. 83–91, 2016.

[9] R. Harakawa, T. Ogawa, and M. Haseyama, "[Paper] accurate and efficient extraction of hierarchical structure ofWeb communities forWeb video retrieval," *ITE Transactions on Media Technology and Applications*, vol. 4, no. 1, pp. 49–59, 2016.

[10] L. Gu, J. Liu, and A. Qu, "Performance evaluation and scheme selection of shot boundary detection and keyframe extraction in content-based video retrieval," *International Journal of Digital Crime and Forensics*, vol. 9, no. 4, pp. 15–29, 2017.

[11] W. Feng, R. Liu, and Z. Zhu, "Fall detection for elderly person care in a vision-based home surveillance environment using a monocular camera," *Signal, Image and Video Processing*, vol. 8, no. 6, pp. 1129–1138, 2014.

[12] R. M. Bommisetty, A. Khare, M. Khare, M. Khare, and P. Palanisamy, "Content-based video retrieval using integration of curvelet transform and simple linear iterative clustering," *International Journal of Image and Graphics*, vol. 132, no. 16, pp. 6–9, 2021.

[13] S. Ren, K. He, R. Girshick, J. Sun, and R.-C. N. N. Faster, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[14] T. Roska and L. O. Chua, "The CNN universal machine: an analogic array computer," *IEEE Transactions on Circuits & Systems II Analog & Digital Signal Processing*, vol. 40, no. 3, pp. 163–173, 2015.

[15] Z. Zeng, T. Huang, and W. X. Zheng, "Multistability of recurrent neural networks with time-varying delays and the piecewise linear activation function," *Neurocomputing*, vol. 21, no. 8, pp. 1371–1377, 2016.

[16] Z. Wei, Z. Lin, H. Kim, Y. Kim, and J. Kim, "An improved object tracking algorithm based on camshift combined with active contour and kalman filter," *Journal of Information and Computational Science*, vol. 11, no. 6, pp. 1753–1764, 2014.

[17] E. Suganya and C. Rajan, "An AdaBoost-modified classifier using stochastic diffusion search model for data optimization in Internet of Things," *Soft Computing*, vol. 24, no. 14, pp. 10455–10465, 2020.

[18] Y. Wang and L. Feng, "Improved Adaboost algorithm for classification based on noise confidence degree and weighted feature selection," *IEEE Access*, vol. 8, pp. 153011–153026, 2020.

[19] Z. Ibrahim, M. Saab, and I. Sbeity, "VideoToVecs: a new video representation based on deep learning techniques for video classification and clustering," *SN Applied Sciences*, vol. 1, no. 6, pp. 1–7, 2019.

[20] J. M. Calvin, M. Hefter, and A. Herzwurm, "Adaptive approximation of the minimum of Brownian motion," *Journal of Complexity*, vol. 39, no. 4, pp. 17–37, 2017.

[21] Y. Lu, K. Gu, and Y. Cai, "Automatic lipreading based on optimized OLSDA and HMM," *Soft Computing*, vol. 26, no. 9, pp. 4141–4150, 2022.

[22] X. Sun, P. Wu, and S. Hoi, "Face detection using deep learning: an improved faster RCNN approach," *Neurocomputing*, vol. 299, no. 7, pp. 42–50, 2018.
[23] M. Frid-Adar, I. Diamant, E. Klang, A. Amitai, J. Goldberger, and H. Greenspan, "GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification," *Neurocomputing*, vol. 321, no. 12, pp. 321–331, 2018.
[24] R. Andri, L. Cavigelli, D. Rossi, and L. Benini, "YodaNN: an architecture for ultralow power binary-weight CNN acceleration," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 1, pp. 48–60, 2018.
[25] J. Wu, C. Hu, Y. Wang, X. Hu, and J. Zhu, "A hierarchical recurrent neural network for symbolic melody generation," *IEEE Transactions on Cybernetics*, vol. 50, no. 6, pp. 2749–2757, 2020.
[26] Z. Li and S. Li, "Kinematic control of manipulator with remote center of motion constraints synthesised by a simplified recurrent neural network," *Neural Processing Letters*, vol. 54, no. 2, pp. 1035–1054, 2022.
[27] G. Fu, "Deep belief network based ensemble approach for cooling load forecasting of air-conditioning system," *Energy*, vol. 148, no. 4, pp. 269–282, 2018.
[28] J. M. Wolterink, T. Leiner, M. A. Viergever, and I. Isgum, "Generative adversarial networks for noise reduction in low-dose CT," *IEEE Transactions on Medical Imaging*, vol. 36, no. 12, pp. 2536–2545, 2017.
[29] M. Ramadas, M. Pant, A. Abraham, and S. Kumar, "ssFPA/DE: an efficient hybrid differential evolution-flower pollination algorithm based approach," *International Journal of System Assurance Engineering and Management*, vol. 9, no. 1, pp. 216–229, 2018.
[30] M. A. Ingle and G. R. Talmale, "Respiratory mask selection and leakage detection system based on Canny edge detection operator," *Procedia Computer Science*, vol. 78, pp. 323–329, 2016.
[31] A. Roy, C. P. Dubey, and M. Prasad, "Gravity inversion for heterogeneous sedimentary basin with b-spline polynomial approximation using differential evolution algorithm," *Geophysics*, vol. 86, no. 3, pp. 1–63, 2021.
[32] S. H. Alizadeh, A. Hediehloo, and N. S. Harzevili, "Multi independent latent component extension of naive Bayes classifier," *Knowledge-Based Systems*, vol. 213, no. 2, Article ID 106646, 2021.
[33] G. Stuart, N. Spruston, B. Sakmann, and M. Häusser, "Action potential initiation and backpropagation in neurons of the mammalian CNS," *Trends in Neurosciences*, vol. 134, no. 3, pp. 440–444, 2016.
[34] N. Zhang, S. L. Shen, A. Zhou, and Y. F. Jin, "Application of LSTM approach for modelling stress-strain behaviour of soil," *Applied Soft Computing*, vol. 100, Article ID 106959, 2021.