Hindawi
Scientific Programming
Volume 2021, Article ID 7252896, 11 pages
https://doi.org/10.1155/2021/7252896

Research Article

A Sports Training Video Classification Model Based on Deep Learning

Yunjun Xu

School of Physical Education, Shaoyang University, Shaoyang 422000, China

Correspondence should be addressed to Yunjun Xu; yunjun.xu@adamson.edu.ph

Received 9 April 2021; Revised 9 May 2021; Accepted 22 May 2021; Published 30 May 2021

Academic Editor: Shah Nazir

A sports training video classification model based on deep learning is studied for targeting low classification accuracy caused by the randomness of objective movement in sports training video. The camera calibration technology is used to restore the position of the target in the real three-dimensional space. After the camera calibration in the video, the sports training video is preprocessed. The input video segment is divided into equal length segments to obtain the subvideo segment. The motion vector field, brightness feature, color feature, and texture feature of the subvideo segment are extracted, and the extracted features are input into the AlexNet convolutional neural network. ReLU is used as the activation function in this convolutional neural network. Local response normalization is used to suppress and enhance the output of neurons to highlight the performance of useful information, so that the output classification results are more accurate. Event matching method is used to match the convolutional neural network output to complete the sports training video classification. The experimental results of the proposed study show that the model can effectively solve the problems of target moving randomness. The classification accuracy of sports training video is more than 99%, and the classification speed is faster which is shown from the results of the experiments.

1. Introduction

With the rapid development of multimedia technology, sports get unprecedented attention and development. The mainstream research work of sports training video includes field and ground wire detection, player detection, recognition and tracking, camera calibration, event detection, and video abstract extraction. The classification of sports training video based on semantic information refers to the use of machine vision technology to automatically identify the types of sports training on the field and give the recognition results by using a certain way of expression [1]. Due to the extensive influence of sports, the introduction of machine vision technology and machine learning technology in sports training video classification has great potential commercial application value.

At present, there are few researches on sports training video classification. Zhu et al. used Gaussian mixture model to achieve player detection. The multitarget tracking method based on support vector regression particle filter was used to extract the trajectory of players and football, and the interactive space-time information between players and football trajectory was used to achieve tactical behavior expression and recognition in football game. Niu et al. achieved camera calibration by detecting and tracking the ground wire in the video image and finally achieved tactical behavior expression and recognition by using the space-time trajectory information of the interaction between players and football in real space. Matej Perse et al. proposed a two-stage framework to realize the tactical behavior recognition in basketball games. In the first stage, players' trajectory is segmented according to the Gaussian mixture model under the generalized context information in basketball games. In the second stage, players' trajectory is semantically expressed according to the key information, and the tactical behavior recognition is realized by using the template matching method. Chen et al. designed an automatic recognition system, which realized camera calibration by field line detection, and realized attack and defense pattern recognition in basketball game by using player trajectory description in

the field. Masui et al. used background subtraction to detect players and then represented the spatial distribution of players in different areas of the field by using symbol system, to realize football tactical behavior recognition. This idea was a nontracking tactical behavior recognition method. The existing tactical behavior recognition mostly used the target trajectory as the underlying visual feature, which faced many problems. Firstly, due to the mutual occlusion between targets, the randomness of target movement, and the complexity of the environment background, there are still many problems in the accuracy and persistence of target tracking; secondly, because the sports training video is mainly based on long-distance view, the identification of players and balls is poor under complex lighting conditions.

Deep learning forms more abstract high-level features by combining low-level features to discover distributed features of data. The multilayer network structure of deep model can make the network learn the organization form of features by itself [2], and get the final semantic features through multiple abstractions. In 2006, Hinton et al. proposed the first feasible depth model. Since then, deep learning has become a new research field of machine learning, known as a revolutionary new technology in the field of artificial intelligence. Deep learning constructs multilayer network model and combines low-level features to form high-level semantic features with abstract representation, so as to simulate the way of thinking of human brain for perception and recognition. At present, deep learning has been widely used in speech, image, and other data recognition, detection and other fields, and has achieved remarkable results. Following are the main contributions of the study:

(i) To study the sports training video classification model based on deep learning

(ii) Establish the sports training video classification model by using convolution neural network of deep learning method

(iii) To verify the effectiveness of the proposed approach through experiments

2. Materials and Methods

2.1. Camera Calibration. Camera calibration technology is used to restore the position of the target in the real three-dimensional space. On this basis, the radial distortion and tangential distortion in the nonlinear model are fully considered, the Rodrigues rotation equation is used to reduce the number of optimization parameters, and the steepest descent method and LM optimization method are used to solve the accurate parameters, respectively.

Because the actual lens in the video is not ideal perspective imaging, with varying degrees of distortion, this kind of distortion can be divided into radial distortion and tangential distortion [3]. In order to describe the imaging model accurately, two parameters are used to describe the lens radial distortion and tangential distortion. The relationship between ideal coordinates and distortion parameters is as follows:

$$\begin{cases} x_d = x_u + \delta_x = x_u + k_1 r^2 x_u + k_2 r^4 x_u + k_3 r^6 x_u + 2k_4 x_u y_u + k_5\left(2x_u^2 + r^2\right), \\ y_d = y_u + \delta_y = y_u + k_1 r^2 y_u + k_2 r^4 y_u + k_3 r^6 y_u + k_4\left(2y_u^2 + r^2\right) + 2k_5 x_u y_u. \end{cases} \tag{1}$$

In (1), (x_u, y_u) is the normalized image coordinate calculated by the pinhole camera model; (x_d, y_d) is the image coordinate actually containing distortion; δ_x and δ_y are the nonlinear distortion values; $r^2 = x_u^2 + y_u^2$; k_1, k_2, k_3, k_4, and k_5 are the nonlinear distortion parameters, where k_1, k_2, and k_3 are the radial distortion coefficients, which will cause the radial movement of real image points on the image plane; k_4 and k_5 are the tangential distortion coefficients.

Given the initial parameters, to solve the precise camera parameters is essentially to solve the unconstrained multidimensional extremum problem. Because there is a deviation between the theoretical value of pixel coordinates and the measured value after the target feature points are projected to the image plane [4–6], the optimal estimation of camera parameters needs to meet the minimum deviation. According to the nonlinear optimization theory, the objective function is expressed as follows:

$$\min F(x) = \min \sum_{i=1}^{n} \sum_{j=1}^{p} \left| m_{ij} - \hat{m}_{ij}\left(A, k_1, k_2, k_3, k_4, k_5, R_i, t_i, M_j\right) \right|^2. \tag{2}$$

In (2), n is the number of target images captured by the camera under different viewing angles; p is the number of target feature points; m_{ij} is the observed value of the coordinate of the j-th feature point of the i-th target image; $\hat{m}_{ij}$ is the theoretical value of the projection point coordinate of the target feature point under the nonlinear model; M_j is the spatial coordinate of the j-th feature point on the target.

In the process of capturing the target from different angles, the internal parameters of the camera are regarded as constant, and the external parameters are different from each shooting angle. The number of optimized parameters increases significantly with the increase of the target image [7–9]. Rodrigues rotation equation provides a method of using vector to represent rotation. If the 3×3 rotation matrix with 9 elements is represented by 3 elements of a vector $r = \begin{bmatrix} r_x & r_y & r_z \end{bmatrix}$, the external parameters of each image are reduced to 6, which greatly reduces the amount of calculation in the optimization process.

The relationship between rotation matrix and rotation vector is as follows:

$$R = \cos\theta \cdot I + \sin\theta \cdot [z]_x + (1 - \cos\theta) \cdot rr^T. \quad (3)$$

The steepest descent method searches along the negative gradient direction of the objective function until it reaches the lowest point of the objective function. For unimodal function, it can quickly get the extreme point. This method uses the principle that the function value along the negative gradient direction of the initial point decreases continuously to search. For the initial point X_0 of function F, there are sequences X_0, X_1, and X_2, which satisfies the relationship as follows:

$$X_{n+1} = X_n - \lambda_n \nabla F(X_n), \quad n \geq 0. \quad (4)$$

The corresponding function values have the following relations:

$$F(X_0) \geq F(X_1) \geq F(X_2) \geq \cdots F(X_n). \quad (5)$$

Because the objective function has the form of minimal sum of squares and the coordinates of feature points on the target image are nonlinear functions of parameters to be estimated, it belongs to nonlinear least squares optimization problem. LM method can avoid the case that $A_k^T A_k$ is ill-conditioned matrix in least squares. In LM algorithm, the descent direction is given by the following equation:

$$d_k = -\left(A_k^T A_k + \alpha_k I\right)^{-1} A_k^T f_k. \quad (6)$$

Through the above process to restore the position of the target in the real three-dimensional space, the accuracy of sports training video classification is improved.

2.2. Video Preprocessing. Before classifying the sports training videos, it needs to firstly preprocess the sports training videos. Shooting video on sports training site is usually divided into distance video, medium distance video, and close distance video [10]. The proportion of sports training remote shooting is relatively large; remote shooting can effectively obtain the whole field information. $V = \{v_1, v_2, \ldots, v_i, \ldots, v_N\}$ is used to represent video input, where V represents the video segment corresponding to a specific sports event, v_i represents the video image of frame i, and $i = 1, 2, 3, \ldots, N$, N indicates the number of frames converted into video frame image of the input video segment.

In order to classify sports training videos more accurately, the input video segments are segmented according to equal length [11], and several subvideo segments are obtained. The expression is as follows:

$$\begin{aligned} V &= \{V_1, V_2, V_3, \ldots, v_M\}, \\ v_j &= \{v_{j1}, v_{j2}, \ldots, v_{jq}, \ldots, v_{jm}\}. \end{aligned} \quad (7)$$

In the above equation, $jm = pm$, $j \neq p$, $j, p = 1, 2, \ldots, M$, $q = 1, 2, \ldots, m.v_j$ represents the j-th subvideo segment after video segmentation, v_{jq} represents the q-th frame image in the j-th subvideo segment, and M represents the number of subvideo segments. After the above processing, the input and segmentation of the sports training video are completed, and the time span of the segmented video field has a certain impact on the classification results.

2.3. Feature Extraction

2.3.1. Extraction of Motion Vector Field.

(1) Let the size of the sports training video be $M \times N \times T$, $M \times N$ denote the resolution, and T denote the length of the video sequence. The video is divided into $K \times L$ blocks; each block size is $h \times v$, where $h = M/K$ and C denotes the number of blocks in each block.

(2) A rectangular coordinate system is established and the motion vector is mapped to this coordinate system [12]. The mapping diagram of the motion vector field of the rectangular coordinate system is shown in Figure 1.

In Figure 1, $MV(i, j)$ is the block with position (i, j), $\theta \in [0, 2\pi)$ is the direction of the motion vector C. If C_x is the component of the motion vector of the C-th block in the horizontal (x) direction, C_y is the component of the motion vector of the C-th block in the vertical (y) direction, and ρ is the motion intensity of the block C; then,

$$\begin{cases} \rho = \sqrt{c_x^2 + c_y^2}, \\ \sin(\theta) = \dfrac{c_y}{\rho}, \\ \tan(\theta) = \dfrac{c_y}{c_x}. \end{cases} \quad (8)$$

(3) The coordinate system of continuous video frames is arranged in chronological order [13], and it is divided into Q equal angle sectors along the positive x direction, p is quantized to R intervals, and then the histograms of p and θ are made, respectively, so it can obtain

$$\begin{cases} \text{Hist}_q = \dfrac{1}{T} \displaystyle\sum_{t=1}^{T} \sum_{i=1}^{C} q_i^t, \quad q \in [1, Q], \\ \text{Hist}_r = \dfrac{1}{q} \displaystyle\sum_{t=1}^{T} \sum_{i=1}^{C} r_i^t, \quad r \in [1, R]. \end{cases} \quad (9)$$

In (9), q_i^t represents the number of motion vectors in quadrant q in frame t, and r_i^t represents the number of p quantized to r in frame t.

(4) The expectation and variance of the motion vector in the x and y directions are used to evaluate the motion in the block, namely,

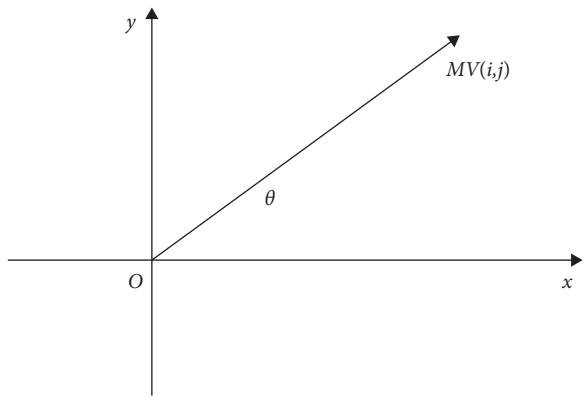


Figure 1: Diagram of motion vector field mapping.

$$\begin{cases} \mu_x = \dfrac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{C} C_{x,i}^t, \\ \mu_y = \dfrac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{C} C_{y,i}^t, \\ \sigma_x^2 = \dfrac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{C} \left(C_{x,i}^t - \mu_x\right)^2, \\ \sigma_y^2 = \dfrac{1}{T}\sum_{t=1}^{T}\sum_{i=1}^{C} \left(C_{y,i}^t - \mu_y\right)^2. \end{cases} \tag{10}$$

In (11), $C_{x,i}^t$ and $C_{y,i}^t$ represent the components of the motion vector of the i-th macroblock in the x and y directions in a frame, and μ_x, μ_y, σ_x^2, and σ_y^2 represent the expectation and variance of the motion vector of the macroblock in the x and y directions, respectively.

2.3.2. Extraction of Luminance Feature. Assuming that the frame resolution is $M \times N$, each frame is divided into $k \times k$ blocks, and the size of each block is $h \times v$, where $h = M/K$, $v = N/K$, x_i represents the brightness value of the i-th pixel in the block, and the average brightness value of each block is $\overline{X(l)}$, $l \in [1, k \times k]$, namely,

$$\overline{X(l)} = \sum_{i=1}^{h\times v} \frac{x_i}{(h \times v)}. \tag{11}$$

If y is used to represent the encoding value of the block luminance comparison, the encoding value of the luminance comparison result between the m-th block and the n-th block in the frame can be expressed by (12), where $l \le m \le k \times k$ and $2 \le n \le k \times k - 1$.

$$u = \begin{cases} 1, & \text{if } \overline{X(m)} > \overline{X(n)}, \\ 0, & \text{other.} \end{cases} \tag{12}$$

Through (12), the frames can be compared according to the average brightness of blocks and encoded with "1" and "0".

2.3.3. Color Feature Extraction. Assuming that the frame size is $M \times N$, the frame is converted into HSV model and divided into $k \times k$ blocks; each block size is $h \times v$, where $h = M/K$, $v = N/K$. $x_{i,m,n}$ represents the pixel value of the m component of the i-th pixel in the n-th block of the video, where $n \in [1, k \times k]$, $i \in [1, h \times v]$, and $m \in [H, S, V]$; then, the color characteristics of the sports training video are as follows:

$$\begin{aligned} \mu_{m,n} &= \frac{1}{h \times v}\sum_{i=1}^{h\times v} x_{i,m,n}, \\ \sigma_{m,n} &= \sqrt{\frac{1}{h \times v}\sum_{i=1}^{h\times v}\left(x_{i,m,n} - \mu_{m,n}\right)}, \\ S_{m,n} &= \sqrt[3]{\frac{1}{h \times v}\sum_{i=1}^{h\times v}\left(x_{i,m,n} - \mu_{m,n}\right)^3}. \end{aligned} \tag{13}$$

In the above equation, $\mu_{m,n}$, $x_{i,m,n}$, and $S_{m,n}$ respectively represent the mean value, variance, and third-order moment of m component in the n-th block.

2.3.4. Texture Feature Extraction. Let i have L gray levels in sports training video. G denotes a gray level cooccurrence matrix, and its element p_{ij} is the times of pixel pairs with gray level i and gray level j in i. p_{ij} is calculated as follows:

$$p_{ij} = N\{(x, y) | f(x, y) = i, f(x + \Delta x, y + \Delta y) = j\}, \tag{14}$$

where $f(x, y)$ is the gray level of the pixel (x, y), and Δx and Δy reflect the distance d and direction θ between the two points.

The most commonly used texture feature is used as the classification feature of sports video. The definition is as follows:

$$f_1 = \sum_{i=1}^{L}\sum_{j=1}^{L} \frac{p_{ij}}{1 + |i - j|}. \tag{15}$$

2.4. Sports Training Video Classification Model Based on Convolutional Neural Network

2.4.1. Neuron Layer Structure of Convolutional Neural Network. A convolutional neural network usually consists of multiple convolution layers, down sampling layers, and normalization layers. Finally, the two-dimensional feature map is connected into a vector and input to the final classifier through the fully connected layer to get the probability output.

(1) Convolution Layer. In a convolution layer, the features of the upper layer are convoluted by a learnable convolution kernel, and then the output features can be obtained through an activation function [14]. Each output may be combined to convolute the values of multiple inputs:

$$x_j^1 = f\left(\sum_{i \in M_j} x_i^{l-1} \cdot k_{ij}^{l-1} + b_j^1\right). \tag{16}$$

In the above equation, M_j represents the set of input features connected by a convolution kernel. M_j determines the connection between convolution kernel and input layer. The output feature map is obtained by convolution kernel of input feature map. Assuming that each convolution kernel extracts a pattern, each output feature map corresponds to a feature and each convolution kernel is equivalent to a feature map. This is because the convolution layer uses weight sharing technology; that is, each neuron uses the same convolution check input to do convolution and each neuron is only connected with some input neurons, which reduces the number of convolution layer parameters. Function f is the activation function of neurons, which is usually a nonlinear function.

The input of convolution layer is multiple two-dimensional planes, and each convolution core is connected with all input channels [15]. Convolution is performed in a three-dimensional space to obtain the position response output. Finally, the convolution checks the convolution of the whole input space to obtain a feature map. Usually, multiple convolution kernels are set in each convolution layer, and each convolution kernel extracts different features, so that each feature map represents the feature plane extracted by the corresponding convolution kernel.

(2) Down Sampling Layer. The purpose of the down sampling layer is to improve the robustness of the network to the small deformation of the input samples, so as to enhance the generalization performance of the network. y_{ijk} is used to represent the output of a neuron in the down sampling layer. The down sampling layer can be expressed as

$$y_{ijk} = \sum_{pq} w_{pq} x_{i,j+p,k+q}, \tag{17}$$

where w_{pq} is the normalized weighted window, which can make down sampling of every input feature map without crossing different feature maps. The number of output feature maps in the down sampling layer is the same as the number of input feature maps, which reduces the resolution of each feature map.

(3) Normalization Layer. The normalization layer is very important for improving the performance of neural network. In convolution neural network model, the normalization layer includes the normalization of the feature vector of the same feature map and the feature map located in different feature maps, which strengthens the feature map with higher response value, and drives different convolution kernels to learn different patterns [16, 17]. The subtraction and normalization operation at a given location are actually the value of the location minus the weighted value of each pixel in the neighborhood. The weight can be determined by a Gaussian weighted window. Division normalization is a common normalization algorithm, which can intensify the difference of response value and improve the effect of high characteristic of response value.

Local response normalization is a common normalization algorithm in convolutional networks. The response value can be expressed as

$$b_{x,y}^i = \frac{a_{x,y}^i}{\left(K + \alpha \sum_{j=\max(0,i-(n/2))}^{\min(N-1,i+(n/2))} \left(a_{(x,y)}^i\right)^2\right)^\beta}, \tag{18}$$

where $a_{x,y}^i$ represents the value of the i-th input feature map at the coordinate (x, y); N represents the number of input feature maps; n represents the normalization on the adjacent n maps.

The local response normalization layer contains three adjustable parameters, namely, the number of feature maps n and parameters α and β. All normalization layers adopt the same parameter setting, such that $n = 5$, $\alpha = 0.0005$, $\beta = 0.5$.

(4) Fully Connected Layer. The fully connected layer is usually at the top of the neural network, which forms a traditional multilayer perceptual network together with the decision-making layer to classify the features extracted from the convolution layer. The overfitting of convolutional neural network is mainly caused by more parameters in the fully connected layer. Dropout technology is usually added to the fully connected layer, and some neurons are randomly selected to participate in the training to prevent the network from overfitting.

A multilayer convolutional neural network is composed of the above five neuron layers, which perform different functions, respectively, and must be combined according to certain rules to achieve better results. Among the five neuron layers, only the convolution layer and the fully connected layer contain trainable parameters, and the convolution layer can retain the input spatial position information, which is required by the down sampling layer. The convolution layer is usually used alternately with the down sampling layer, so that different convolution layers can extract different scale features [18]. The fully connected layer will destroy the position information of feature planes and the difference between each feature plane. The fully connected layer is usually used as a part of the final multilayer perceptual classifier, which integrates the convolution layer and the down sampling layer to extract features and send them to the decision layer for classification.

2.4.2. Structure of Improved Convolutional Neural Network. The AlexNet convolutional neural network of deep learning is used to classify sports training videos. The AlexNet convolutional neural network consists of 23 layers, including five convolution layers and three fully connected layers.

(1) Use the New Activation Function ReLU. Generally, the activation function of artificial neuron is hyperbolic tangent function $f(x) = \tanh(x)$ or sigmoid function $f(x) = (1 + e^{-x})^{-1}$. In the experiment, it is found that when sigmoid or hyperbolic tangent function is used to calculate the error gradient by backpropagation, the derivation involves division, which leads to a large amount of calculation; once the number of layers of traditional neural network increases, the gradient fading problem occurs. The root cause is that when sigmoid or hyperbolic tangent function is used to calculate the error gradient by backpropagation, the change of function value slows down, and its derivative is close to zero, which makes other hidden layers far away from the output layer prone to gradient fading [19]; in addition, it is also a disadvantage of sigmoid function to add weight penalty factor to get sparsity and output nonzero mean value.

The advantages of ReLU function $f(x) = \max(0, x)$ are as follows: first, the calculation speed and convergence speed are faster; second, ReLU will make the output 0 when $x < 0$, resulting in network sparsity, reducing the interdependence of parameters, and alleviating the over fitting problem; third, its derivation is piecewise linear in both forward and backward propagation, avoiding the disappearance of gradient.

(2) Local Response Normalization (LRN). In neurobiology, there is a concept called "lateral inhibition", which refers to the ability of excited neurons to inhibit their adjacent neurons. That is to highlight the maximum peak in the local sensing area and increase the ability of biological perception.

It is in the neural network that the LRN layer realizes "lateral inhibition". Let a^i_{xy} be the activation value of neurons at position (x, y) of the i-th kernel function and b^i_{xy} be the activation value after normalization, and the total number of kernel functions is N; then, the mathematical model of LRN is expressed as follows:

$$b^i_{xy} = \frac{a^i_{xy}}{\left(k + \alpha \sum_{j=\max(0, i-(1/2))}^{\min(N-1, i+(n/2))} \left(a^i_{xy}\right)^2\right)^{\beta}}, \tag{19}$$

where the sum operation is normalized at the adjacent position of n around (x, y), and the super parameters k, n, α, and β need to be determined by the verification set. It is very effective to add LRN layer after using ReLU function as the activation function. The ReLU function has unlimited activation ability when $x > 0$, which needs LRN normalization. It is expected that the LRN layer can detect the features with high frequency and amplify them by suppressing the peripheral neurons; the LRN layer will suppress the uniform response in any given local neighborhood; that is, if all the values are large, then the normalization will suppress all the values uniformly. The purpose of LRN layer is to make useful information more prominent by inhibiting and enhancing neuron output.

2.5. Event Matching. Based on the output of convolutional neural network, the events of sports training test video sequence and reference video sequence are matched by event matching method. Given L_1 observation symbols of video class, a multistate traversed convolutional neural network model is trained by using features extracted from sports training video frames, to obtain the event sequence (event probability and corresponding state transition) in the corresponding reference video. The reference event sequence is used to create a dictionary for a given sports training event [20]. For the event with a specific state transition (k, l) in the reference event, the probability distribution of the event is approximated by a Gaussian density function $N(\mu_{kl}, \sigma_{kl})$, where μ_{kl} and σ_{kl} represent the mean value and variance of the density function, respectively. It is given by the following equation:

$$\begin{aligned} \mu_{kl} &= \frac{1}{L_1} \sum_{t=1}^{L_1} e^p_t(k, l), \\ \sigma_{kl} &= \sqrt{\frac{1}{L_1} \sum_{t=1}^{L_1} \left(e^p_t(k \cdot l) - \mu_{kl}\right)^2}. \end{aligned} \tag{20}$$

Each state transition is assigned a mean value and variance to represent the probability $e^p_t(k \cdot l)$ of the event occurring in the category. For the sports training video clips that do not appear in the training stage, a reference convolution neural network model is used to obtain the events. Let $e^p_t(k \cdot l)$ denote the event probability of state transition (k, l) at time t when the test sequence in the observation symbol provides a reference model. Let L_2 denote the number of observation symbols in the test sequence. The similarity between the test video clip and the reference model is expressed by the following equation:

$$s = \frac{1}{L_2} \sum_{t=1}^{L_2} \frac{1}{\sqrt{2\pi}\sigma_{kl}} \exp\left[-\frac{\left(\widehat{e}^t_p(k, l) - \mu_{kl}\right)^2}{2\sigma^2_{kl}}\right]. \tag{21}$$

The similarity value s between video clips and all kinds of sports training is compared, and they are classified into the category with the highest similarity value.

3. Results and Discussion

In order to verify the feasibility and effectiveness of the sports training video classification model, eight data sets which are often used in the classification research in the network are selected as the test objects. The data sets include eight types of sports training videos, such as basketball, volleyball, and football. The detailed contents of the videos in each data set are shown in Table 1.

Table 1 shows that the experimental data set contains many types of sports training videos. Different sizes and types of sports training videos are used to test the classification performance of different models of sports training

Table 1: Details of the experimental data sets.

Data set name	Video content	Video frames/N	Duration/s	Size/MB
DataSetA	Badminton, basketball, table tennis	1582	66	4.52
DataSetB	Tennis, volleyball	1351	56	3.64
DataSetC	football, running	1254	52	2.85
DataSetD	Snooker, tennis	3269	136	7.48
DataSetE	Basketball, tennis	5642	235	9.52
DataSetF	Table tennis, volleyball	1478	62	3.48
DataSetG	Basketball, football	3151	131	8.64
DataSetH	Running, basketball	1856	77	4.85

videos. Support vector machine model and HMM model are selected as comparison models.

Three models are used to classify the sports training videos of 8 data sets, and the classification results are shown in Table 2.

The experimental results in Table 2 show that the classification of sports training videos can be realized by using the proposed model. The classification results of sports training videos by using the proposed model are similar to those of actual sports training videos, which indicates that this model has high classification performance of sports training videos.

In the result of sports training video classification of the proposed model, two images are randomly intercepted in basketball training video, as shown in Figure 2.

As can be seen from the experimental results in Figure 2, using the proposed model to classify basketball training videos can accurately classify videos according to the extracted features of sports training videos, and randomly intercepted pictures are all accurate for basketball training, which verifies that the proposed model has high classification effectiveness of sports training videos.

The classification accuracy, recall rate, and precision rate are selected as the important indexes to evaluate the classification performance of the proposed model. n_c is used to represent the number of correct recognition results, n_m is used to represent the number of wrong recognition results, and n_f is used to represent the number of failed recognition results. In order to effectively reduce the error caused by a single experiment, the average value of five experiments is selected, and 2 : 1 ratio is set to randomly divide training samples and test samples. The evaluation index equation is as follows:

$$\begin{aligned} \text{recall ratio} &= \frac{n_c}{(n_c + n_m)}, \\ \text{precision ratio} &= \frac{n_c}{(n_c + n_f)}. \end{aligned} \tag{22}$$

Statistics of the accuracy comparison results of different data sets and different types of sports training video classification are shown in Figure 3.

As can be seen from the experimental results in Figure 3, under different data sets and different types of sports training, the classification accuracy of sports training video classified by the proposed model is higher than 99%, and the classification accuracy of sports training video classified by this model is significantly higher than that of the other two models, which effectively verifies that this model has higher classification accuracy of sports training video.

Statistics of the recall rate comparison results of different data sets and different types of sports training video classification are shown in Figure 4.

As can be seen from the experimental results in Figure 4, under different data sets and different types of sports training, the recall rate of sports training videos classified by the proposed model is higher than 98.5%, and the recall rate of sports training videos classified by this model is significantly higher than that of the other two models, which verifies that this model has higher classification accuracy of sports training videos.

Statistics of the precision rate comparison results of different data sets and different types of sports training video classification are shown in Figure 5.

As can be seen from the experimental results in Figure 5, under different data sets and different types of sports training, the precision rate of sports training video classification using the proposed model is higher than 98%, and the precision rate of sports training video classification using the proposed model is significantly higher than that of the other two models, which verifies the high accuracy of sports training video classification using the proposed model.

The analysis of the above experimental results shows that the classification accuracy, recall rate, and precision rate of different data sets and different types of sports training videos are the best. Basketball and football sports training videos have strong continuity and change more frequently, so more quantitative features are needed to better obtain the change features in videos. Basketball and volleyball videos usually have close range images, while baseball and tennis videos are mostly shot from a long-distance perspective, so feature extraction is difficult. Football is also shot from a long-distance perspective; it has continuous movement in the field and can be well collected by increasing the number of states. In football and basketball videos, it most uses a single camera to track players or regions of interest; unlike other sports training, switching between multiple cameras frequently is conducive to event detection. The model can effectively improve the randomness of target movement and improve the classification accuracy by extracting video features.

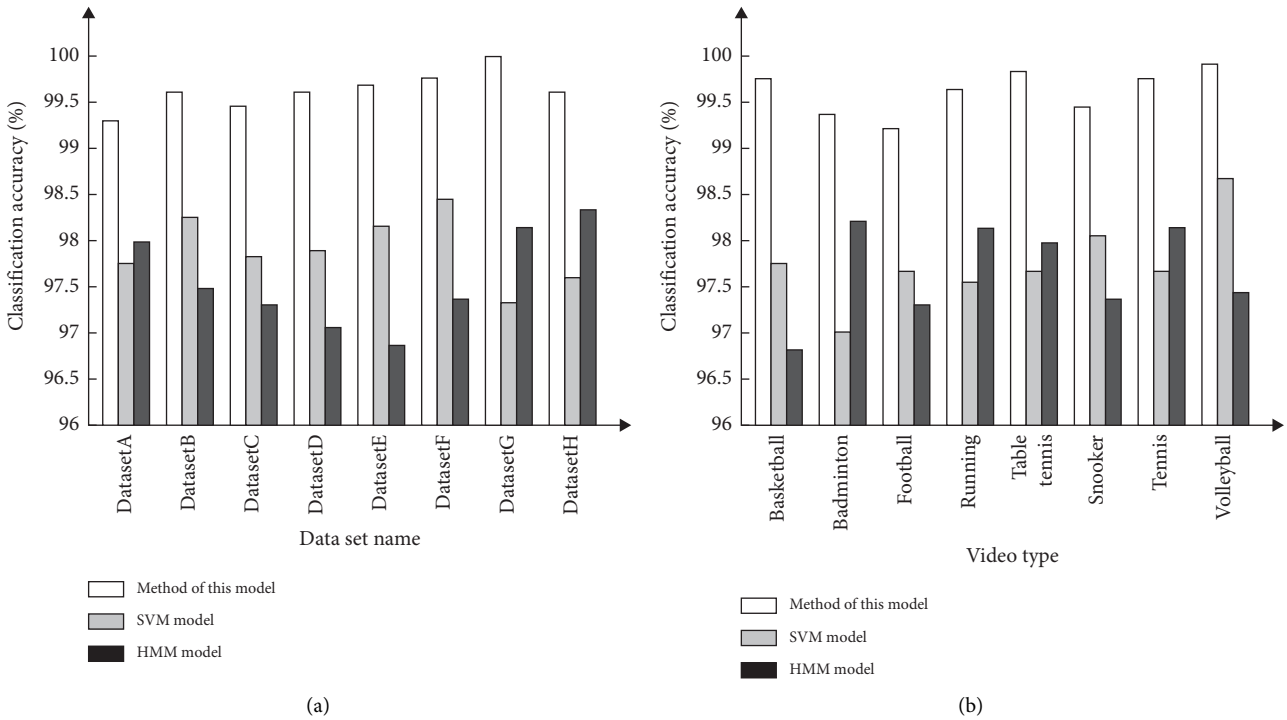
The training time and test time of sports training videos classified by three models with different data sets are counted. The comparison results are shown in Table 3.

Table 2: Classification results of sports training videos.

Types	Actual number of frames/N	Method of this model/N	SVM model/N	HMM model/N
Basketball	1481	1473	1352	1376
Badminton	2384	2415	2384	2584
Football	3436	3418	3364	3468
Running	2564	2542	2498	2348
Table tennis	1765	1759	1743	1842
Snooker	3652	3627	3584	3452
Tennis	2755	2711	2684	2679
Volleyball	1546	1638	1724	1711
Total	19583	19583	19333	19460

(a) (b)

Figure 2: Basketball training classification results.



(a) (b)

Figure 3: Comparison of classification accuracy. (a) Different data sets. (b) Different types of sports training videos.

The experimental results in Table 3 show that the classification speed of sports training video using the proposed model is the fastest, and the accurate classification results of sports training video can be obtained by using shorter training time and test time of the proposed model, which verifies that this model has higher classification efficiency of sports training video.

The above experimental results show that the proposed model can accurately classify all kinds of sports training videos, which shows that this model has good

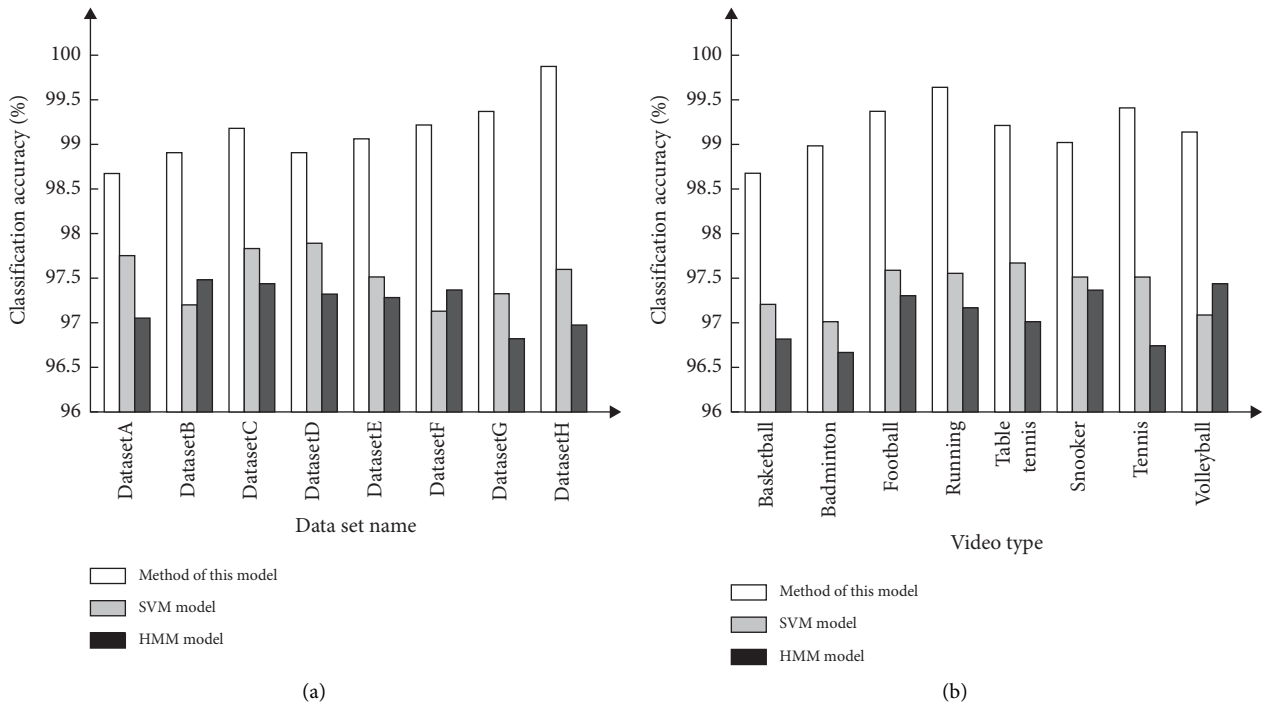


Figure 4: Comparison of classification recall rates. (a) Different data sets. (b) Different types of sports training videos.

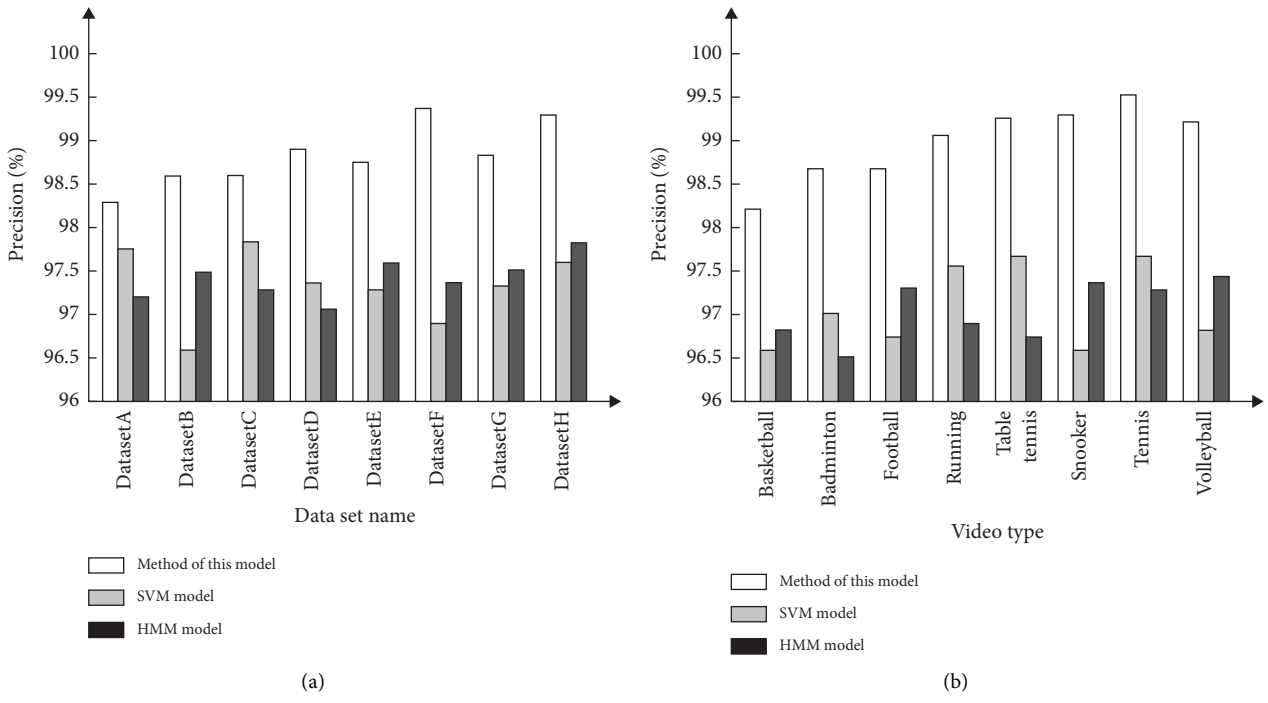


Figure 5: Comparison of classification accuracy. (a) Different data sets. (b) Different types of sports training videos.

classification performance. The main reason is that this model uses deep learning model to establish classification model, which can effectively improve the classification accuracy of sports training videos. For close range videos with similar categories, it still has high classification accuracy. This model has high accuracy and comprehensive performance in classifying all kinds of sports training videos.

Table 3: Comparison of classification speed.

Data set name	Method of this model/ms		SVM model/ms		HMM model/ms	
	Training time	Testing time	Training time	Testing time	Training time	Testing time
DataSetA	152	215	864	856	1165	658
DataSetB	352	234	915	485	1248	594
DataSetC	425	152	1052	715	1352	605
DataSetD	356	236	1135	359	1426	736
DataSetE	412	245	1658	645	1489	852
DataSetF	385	236	2354	597	2654	945
DataSetG	642	215	1842	612	2854	439
DataSetH	531	284	2358	784	1696	896

4. Conclusion

At present and with the passage of time, the amount of sports training video data in the Internet is growing rapidly. In order to effectively manage and retrieve sports training video, accurate classification of sports training video is very important for consideration. Aiming at the shortcomings of existing approaches of sports training video classification, this paper establishes sports training video classification model based on deep learning method. Convolution neural network with deep learning is used for the classification purpose in the proposed research. After classification, event matching operation is performed, and video classification is realized according to similarity. The experimental results show that the proposed model can effectively determine all kinds of sports training videos and accurately detect the occurrence of events through convolution neural network, so as to achieve high-precision classification of sports training videos. Compared with other models, the proposed model has the advantages of simple implementation, fast processing speed, high classification accuracy, high generalization ability, and adaptability.

Data Availability

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

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] C. Y. Fang, K. B. Jia, and P. Y. Liu, "Identification of taxi violation behavior based on surveillance video," *Computer Simulation*, vol. 37, no. 05, pp. 331–336, 2020.

[2] Y. Kumar, M. Sheoran, G. Jajoo, and S. K. Yadav, "Automatic modulation classification based on constellation density using deep learning," *IEEE Communications Letters*, vol. 99, pp. 1–11, 2020.

[3] Q. Li, S. Zheng, Y. Huang, and D. Liu, "Automatic classification of nvst short-exposure data based on deep learning," *Publications of the Astronomical Society of the Pacific*, vol. 133, no. 1020, Article ID 024505, 2021.

[4] A. S. Garea, D. B. Heras, and F. Argüello, "Caffe cnn-based classification of hyperspectral images on gpu," *The Journal of Supercomputing*, vol. 75, no. 3, pp. 1065–1077, 2019.

[5] M. Oleynik, A. Kugic, Z. Kasáč, and M. Kreuzthaler, "Evaluating shallow and deep learning strategies for the 2018 n2c2 shared task on clinical text classification," *Journal of the American Medical Informatics Association*, vol. 26, no. 11, pp. 1247–1254, 2019.

[6] S. Park, W. K. Chung, and K. Kim, "Training-free bayesian self-adaptive classification for semg pattern recognition including motion transition," *IEEE Transactions on Biomedical Engineering*, vol. 67, no. 6, pp. 1775–1786, 2020.

[7] M. Chen, J. Cao, B. Zhang, H. Zhu, and X. Cai, "Lbp-05-classification and mutation prediction based on liver cancer hisopathological images using deep learning," *Journal of Hepatology*, vol. 70, no. 1, pp. e142–e143, 2019.

[8] T. Zebin and S. Rezvy, "Covid-19 detection and disease progression visualization: deep learning on chest x-rays for classification and coarse localization," *Applied Intelligence*, vol. 51, no. 2, pp. 1010–1021, 2021.

[9] C. Ajmi, J. Zapata, J. J. Martínez-Álvarez, G. Doménech, and R. Ruiz, "Using deep learning for defect classification on a small weld x-ray image dataset," *Journal of Nondestructive Evaluation*, vol. 39, no. 3, pp. 68–72, 2020.

[10] G. Murtaza, L. Shuib, A. W. Abdul Wahab et al., "Deep learning-based breast cancer classification through medical imaging modalities: state of the art and research challenges," *Artificial Intelligence Review*, vol. 53, no. 3, pp. 1655–1720, 2020.

[11] X. Zhang, J. Li, Z. Cai, L. Zhang, Z. Chen, and C. Liu, "Overfitting suppression training strategies for deep learning-based atrial fibrillation detection," *Medical & Biological Engineering & Computing*, vol. 59, no. 1, pp. 165–173, 2021.

[12] Y. Wang, J. Wang, W. Zhang, J. Yang, and G. Gui, "Deep learning-based cooperative automatic modulation classification method for mimo systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 4, pp. 4575–4579, 2020.

[13] F. Wang, H. Huang, and J. Liu, "Variational based mixed noise removal with CNN deep learning regularization," *IEEE Transactions on Image Processing: A Publication of the IEEE Signal Processing Society*, vol. 1, no. 99, pp. 1–9, 2019.

[14] D. A. Duev, A. Mahabal, F. J. Masci et al., "Real-bogus classification for the zwicky transient facility using deep learning," *Monthly Notices of the Royal Astronomical Society*, vol. 489, no. 3, pp. 3582–3590, 2019.

[15] R. K. Samala, H.-P. Chan, L. Hadjiiski, M. A. Helvie, C. D. Richter, and K. H. Cha, "Breast cancer diagnosis in digital breast tomosynthesis: effects of training sample size on multi-stage transfer learning using deep neural nets," *IEEE*

Transactions on Medical Imaging, vol. 38, no. 3, pp. 686–696, 2019.

[16] M. Shah, A. Roomans Ledo, and J. Rittscher, "Automated classification of normal and stargardt disease optical coherence tomography images using deep learning," *Acta Ophthalmologica*, vol. 98, no. 6, pp. e715–e721, 2020.

[17] Y. Liang, Z. Hu, and K. Li, "Power consumption model based on feature selection and deep learning in cloud computing scenarios," *IET Communications*, vol. 14, no. 10, pp. 1610–1618, 2020.

[18] N. Maffulli and F. Oliva, "Coper classification early after acl rupture changes with progressive neuromuscular and strength training and is associated with 2-year success: letter to the editor," *The American Journal of Sports Medicine*, vol. 47, no. 11, pp. NP64–NP65, 2019.

[19] L. R. Dugas, C. R. Labella, N. Alawad, J. Pasulka, and N. Jayanthi, "Benefits and challenges of serial sports training risk assessment and counselling in kids: the t.r.a.c.k. randomised intervention study," *British Journal of Sports Medicine*, vol. 53, no. 4, pp. 243–251, 2019.

[20] J. W. Orchard, W. Meeuwisse, W. Derman, M. Hgglund, and R. Bahr, "Sport medicine diagnostic coding system (smdsc) and the orchard sports injury and illness classification system (osiics): revised 2020 consensus versions," *British Journal of Sports Medicine*, vol. 54, no. 7, Article ID 101921, 2020.