Hindawi
Scientific Programming
Volume 2021, Article ID 3481469, 9 pages
https://doi.org/10.1155/2021/3481469

Research Article

Sports Intelligent Assistance System Based on Deep Learning

Boyin Wu

School of Master of Science in Sports, Macao Polytechnic Institute, Macao 999078, China

Correspondence should be addressed to Boyin Wu; p2009629@ipm.edu.mo

Received 25 September 2021; Accepted 23 October 2021; Published 19 November 2021

Academic Editor: Le Sun

Copyright © 2021 Boyin Wu. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Traditional sports aid systems analyze sports data via sensors and other types of equipment and can support athletes with retrospective analysis, but they require several sensors and have limited data. This paper examines a sports aid system that uses deep learning to recognize, review, and analyze behaviors through video acquisition and intelligent video sequence processing. This paper's primary research is as follows: (1) With an eye on the motion assistance system's application scenarios, the network topology and implementation details of the two-stage Faster R-CNN and the single-stage YOLOv3 target detection algorithms are investigated. Additionally, training procedures are used to enhance the algorithm's detection performance and training speed. (2) To address the issue of target detection techniques' low detection performance in complicated backgrounds, an improved scheme from Faster R-CNN is proposed. To begin, a new approach replaces the VGG-16 network in the previous algorithm with a ResNet-101 network. Second, an expansion plan for the dataset is provided. (3) To address the short duration of action video and the high correlation of image sequence data, we present an action recognition method based on LSTM. To begin, we will present a motion decomposition scheme and evaluation index based on the key transaction frame in order to simplify the motion analysis procedure. Second, the spatial features of the frame images are extracted using a convolutional neural network. Besides, the spatial and temporal aspects of the image sequence are fused using a two-layer bidirectional LSTM network. The algorithm suggested in this research has been validated using a golf experiment, and the results are favorable.

1. Introduction

More and more people are devoting their time to sports such as golf and skiing as China's burgeoning sports and health industry becomes more active. When there is no systematic learning, beginners may not develop their technical level because of nonstandard movements, and in certain cases, they may even injure themselves playing sports. To achieve continuous improvements, you must assess and analyze your efforts constantly. Traditionally, sports coaches have done one-to-one coaching. High labor costs and poor flexibility are difficulties for the company. As it currently stands, there is a problem that has to be fixed when it comes to teaching and training. In professional athletes' daily training, they use sports assist devices. Professional training analysts, modeling the athletes based on motion sensor data, use the data of the limbs to track and correct the specifics of the athletes' motions. Because of the high price and the need for specialized motion sensors, the present sports aid system is difficult and costly for sports enthusiasts who are not professional athletes [1–8].

These recent advances in computer vision and natural language processing, made possible by a boost in the amount of training data and the addition of sophisticated feature expression capabilities, have been achieved using deep learning. A great number of practical achievements have been accomplished in the target detection, machine translation, and action identification departments. Many vision tasks are based on object detection, which is a research issue in computer vision. As a result, face recognition, autonomous driving, and target tracking have all seen widespread use in real life. This traditional target detection method extracts characteristics artificially, and the results are sometimes inferior due to poor feature extraction or inadequate recognition. Deep convolutional neural networks can automatically extract task-related feature information from enormous data because of the development of convolutional neural networks. Traditional machine learning

methods, which can easily be implemented by specifying rules for feature extraction, have clear advantages. The deep learning target detection technique, on the other hand, has improved the accuracy and speed of detection significantly. By further expanding the applications of computer vision, it increases the overall value. Another field of computer vision study is semantic analysis of video, specifically human action recognition, which has a variety of possible applications, such as human-computer interaction and gesture recognition. The primary goal of it is for videos. Pure static graphics no longer serve as a proper means of describing video aspects. Human motion recognition hence has to assess the temporal features in addition to extracting the spatial features of each frame of video. Recurrent neural networks are now frequently employed in machine translation, text generation, and personalized recommendation applications. Recurrent neural networks and convolutional neural networks are being combined to address the demanding problem of processing video sequences [6–15].

This paper studies the exercise assistance system, which does not require sensors. It can recognize movements only through video sequences, which is convenient for users to quickly review and analyze their movements in real time. It not only enriches the learning methods of sports but also provides new ideas for the development of sports assistance systems, which is extremely important for the promotion of the development of sports events.

The contribution of this paper can be summarized as follows: (1) the detection efficiency of target detection algorithm is improved in complex background; (2) a sports recognition algorithm based on LSTM is proposed, which can fuse spatial features as well as temporal features; and (3) the proposed method achieved advanced performance.

2. Related Work

Literature [16] proposed a deep convolutional neural network AlexNet. In the ImageNet competition that year, excellent results were achieved, which proved the huge potential of convolutional neural networks in image processing tasks. Literature [17] proposed R-CNN, which used a selective search algorithm to generate about two thousand candidate regions for each image. Then through the forward propagation of the convolutional neural network, feature extraction was performed on each candidate area, and finally, the feature information of each candidate area was classified using linear SVM. R-CNN can output and correct the bounding box. mAP can reach 58.5%, which was a relatively large improvement compared to the previous algorithm. However, the selective search method needed to extract feature values and classify each candidate frame. This process generated a lot of redundant calculations, which consumed a lot of memory and calculation time, and the average processing time for each picture was about 3 s. The fully connected layer as a classification also required a fixed-size input and forced conversion of the size of the input image would also cause image distortion. In literature [18], the Faster R-CNN was designed for the problem that the candidate region generation algorithm was too time-consuming and space-consuming. The principle of the Faster R-CNN was as follows: first, the image was extracted from the convolutional network to learn the feature and output to the RPN network, and then the RPN network determined in the classification layer that the anchor belonged to the foreground or the background and finally converted the region of interest into a feature vector. At the same time, it was output to the type recognition classifier and the border correction regressor. Faster R-CNN replaced the fully connected layer with a fully convolutional layer, which truly realized end-to-end calculation, and the detection speed and accuracy had been further improved. Literature [19, 20] proposed the YOLO series of algorithms. Divide the input image into several grids. If the target object's center was in the grid, each grid needed to predict the value of N boundary candidate boxes and output the position coordinates and confidence of each boundary box. End-to-end training could be achieved. Although the detection performance was not as good as the two-stage algorithm, its detection speed was faster. But it could only detect target objects that fall into the grid. When a grid contains multiple target objects that are close to each other, the detection performance is relatively poor.

Literature [21] proposed a dual-stream convolutional network, which divided the network into two independent streams, and the spatial convolutional stream learned the spatial characteristics of a single frame of pictures. The optical flow convolutional flow learned the optical flow sequence that represents the characteristics of video timing information. Finally, the two network streams were fused and output. The dual-stream network used a single-frame image and optical flow field bands to represent the spatial and temporal characteristics of the video sequence. However, it was difficult to represent the spatial characteristics of long-term video using a single-frame image. In view of the limitations of the dual-stream model for the sampling of long-term video sequences, literature [22] proposed a time segmentation network to improve the input of the dual-stream network. The long video was randomly divided into K segments and input into K dual-stream networks. Use random sampling to sample image frames from fragments and input them into the spatial convolution stream. Finally, the results of K optical flow networks were merged to obtain the final result, which obtained a high score of 94.2 on the UCF101 dataset. Literature [23] extended the two-dimensional convolutional neural network directly to the three-dimensional convolutional neural network. By adding information in the time dimension, the two-dimensional convolution kernel was extended to the three-dimensional convolution kernel, and the convolution kernel slides in the time dimension. The features of the time dimension could be effectively extracted. Literature [24] designed the I3D network, extended the pooling kernel to three dimensions, and added the optical flow information to the three-dimensional convolutional neural network. Literature [25] added the idea of jump connection of residual network and deepened the network depth of the three-dimensional convolutional network. The proposed network structure based on the three-dimensional convolutional network neural network was relatively simple and inherited the weight sharing and

integration of the convolutional network. The three-dimensional convolutional network was limited by the width of the convolution kernel, and it was difficult to learn long-time information. Literature [26] designed an action recognition algorithm based on a double stream network for hockey action analysis. The network first obtained human body posture information through a partial affinity field, secondly used optical flow field to extract time features, and finally combined posture information and optical flow to estimate the hockey player's movements. Literature [27] proposed that SoccerNet was used for football game video analysis. According to the image information of the football game video, it automatically recognized the key event time nodes in the football game, such as red and yellow cards, goals, and replacement players.

3. Target Detection Based on Deep Learning

With the continuous development of deep learning, target detection algorithms have gradually replaced traditional target detection algorithms. The proposal of R-CNN first applies deep learning technology to target detection tasks. Fast R-CNN is improved and optimized on the basis of R-CNN and has achieved good performance in detection efficiency and training time.

The current research on target detection algorithms is divided into two branches: one is the two-stage target detection with priority in detection accuracy, represented by Faster R-CNN. The feature information is classified and output. The regional suggestion network can output according to the feature information of different scales. Since the algorithm performs regression correction on the target object frame in the process of generating candidate regions and classifying, the algorithm has good detection accuracy; the other is single-stage target detection algorithm with priority on detection speed, and its representative is the YOLO series of algorithms. The principle of the algorithm is as follows: divide the picture into a fixed-size network and use a priori box of preset size to directly perform feature extraction and classification on the a priori box of each grid. Due to the entire process is required, the speed of YOLO is faster. Aiming at the application scenarios of the motion assistance system, this chapter compares and analyzes the current mainstream target detection algorithms based on deep learning. Finally, a comparative experiment is carried out on the motion video keyframe image dataset, and its application scenarios are analyzed according to the detection performance of different algorithms.

3.1. Faster R-CNN. It is the first target detection algorithm that can be trained in an end-to-end method. The subsequent two-stage target detection algorithm is basically improved according to the idea, and Faster R-CNN is considered to be a milestone of the two-stage target detection algorithm. As shown in Figure 1, the workflow of Faster R-ANN is as follows: first, the input image is extracted by the backbone, and then the feature map is input into RPN to obtain the proposal. Cut out the feature map of the candidate area and output it to the ROI Pooling to generate a candidate area, and finally output it to the classifier for type classification and the regression to correct the prediction frame.

Faster R-CNN introduces RPN to extract candidate regions. This is a CNN that can share features of the convolutional layer and extract candidate regions from this. The extracted part of the candidate region is embedded into the network, which in a true sense realizes the end-to-end target detection. After obtaining the candidate area, perform target classification and bounding box regression. RPN′ implementation is as follows: use a 3×3 sliding window to generate a feature vector of length 256 or 512 on the feature map extracted by the network, and then output this feature vector to two fully connected layers for prediction of the center coordinates; width and height of the candidate area are used to predict whether the candidate area belongs to the foreground or the background. This sliding window method can ensure that the regression layer and the classification layer cover the entire space of the feature map. For each sliding window, k region suggestions can be predicted at the same time, so there are 4k outputs for the regression layer, that is, the 4 coordinate parameters corresponding to the candidate region, and the classification layer is 2k, that is, whether the candidate region is the target or the background. The k candidate regions are the parameterization of k anchors. Each anchor has a strong translation invariance, which is beneficial to improve the quality of detection.

$$L(\{p_i\},\{t_i\}) = \frac{1}{N_{\text{cls}}}\sum_i L_{\text{cls}}(p_i, p_i^*) + \lambda\frac{1}{N_{\text{reg}}}\sum_i pi^* L_{\text{reg}}(t_i, t_i^*), \tag{1}$$

where i is the anchor index, p_i is the prediction probability of the anchor i target, if the anchor is positive, p_i^* is 1, otherwise, it is 0, and t_i represents the 4 coordinate parameters of the predicted candidate frame. t_i^* represents the coordinate parameters of the real target frame corresponding to the positive anchor. The specific classification loss and regression loss functions are as follows:

$$\begin{aligned} L_{\text{cls}}(p_i, p_i^*) &= -\log\left[p_i p_i^* + (1-p_i^*)(1-p_i)\right], \\ L_{\text{reg}}(t_i, t_i^*) &= R(t_i - t_i^*), \end{aligned} \tag{2}$$

where R is the robust loss function, as illustrated in the following equation:

$$\text{smooth}_{L_1}(x) = \begin{cases} 0.5x^2, & \text{if}|x| < 1, \\ |x| - 0.5, & \text{otherwise.} \end{cases} \tag{3}$$

Faster R-CNN uses VGG-16 as the backbone network. It has achieved relatively good results in ImageNet classification tasks, and its model compatibility is relatively high. It is widely used as a feature extraction network in various image analysis tasks. Since the target detection task not only needs to classify the image but also needs to locate the target, the ability to extract features may have an important impact on the accuracy of the model. An improved scheme of objection detection is proposed, using ResNet-101 instead of source code VGG-16 as the backbone network of the target

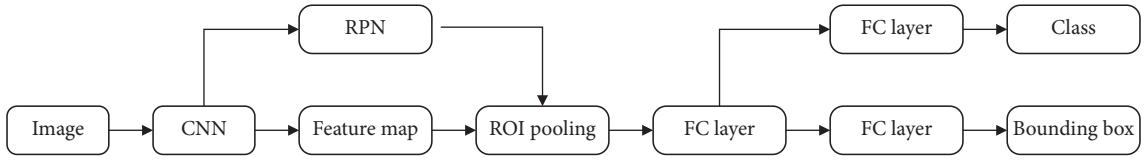

Figure 1: The structure of Faster R-CNN.

detection algorithm. Compared with VGG, the ResNet-101 network has a deeper network structure and can extract richer image feature information. Simultaneously, the network training efficiency is improved by using the residual module, and the network inference speed is accelerated.

3.2. YOLO. The YOLO algorithm is proposed by Redmon et al. It is the representative work of a single-stage target detection algorithm. Its core idea is to transform target detection into a regression task. Different from the two-stage algorithm, the input image can directly output the type information and position information of all detected targets in the image after one inference. In the case of ensuring the accuracy and efficient performance of the target detection task, the entire network only uses a single convolutional neural network, which greatly promotes the training speed and reduces the detection time. As shown in Figure 2, the principle of the YOLO algorithm: first divide the input image into grids, and stipulate that each network is only responsible for detecting objects whose target center falls in the current grid.

Suppose that each grid can predict B target objects, and each box needs to output 4 pieces of position information, that is, the center point coordinates of the target object, the relative height and width of the border, and the confidence of the object corresponding to the output border. The confidence of a five-dimensional tensor represents whether the box contains the target object and the accuracy of the position of the box relative to the real object, which is defined as follows:

$$\text{confidence} = \Pr(\text{Object}) \bullet \text{IOU}_{\text{pred}}^{\text{truth}}, \tag{4}$$

where $\Pr(\text{Object})$ represents whether the center of the detected target object falls in the box; if it falls in the box, its value is 1; otherwise, it is 0. The latter represents the interaction ratio between the predicted object box output by the network and the box of the actual position of the object.

The total loss of YOLO consists of three parts. The formula for classification loss is as follows:

$$L_{\text{cls}} = \sum_{i=0}^{S^2} 1_i^{\text{obj}} \sum_{c \in \text{classes}} \left(p_i(c) - \widehat{p}_i(c)\right)^2, \tag{5}$$

where i is the index of the grid, S is the size of the grid division, c is the category of the objection, and $p_i(c)$ represents the probability that i-th grid contains c target object. 1^{obj} indicates whether there is a target object in the grid i; a value of 1 means that the center of the target object falls in this box; otherwise, it is 0.

The loss function of the bounding box regression is

$$L_{\text{bbox}} = \lambda_{\text{coord}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} \left[\left(x_i - x_i'\right)^2 + \left(y_i - y_i'\right)^2 + \left(\sqrt{w_i} - \sqrt{\widehat{w}_i} \right)^2 + \left(\sqrt{h_i} - \sqrt{\widehat{h}_i} \right)^2 \right], \tag{6}$$

where B means that each grid can output B boxes and λ_{coord} means balance factor.

The loss function of confidence is

$$L_{\text{conf}} = \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{\text{obj}} \left(C_i - \widehat{C}_i\right)^2 + \lambda_{\text{noobj}} \sum_{i=0}^{S^2} \sum_{j=0}^{B} 1_{ij}^{\text{noobj}} \left(C_i - \widehat{C}_i\right)^2, \tag{7}$$

where C_i is the confidence of j-th output box; 1^{noobj} indicates whether the j-th output box of the grid i contains the detection target, if it contains a value of 0; otherwise, it is 1; λ_{noobj} means balance factor.

Therefore, the final loss function of YOLO is

$$L = L_{\text{cls}} + L_{\text{bbox}} + L_{\text{conf}}. \tag{8}$$

In the estimation process of YOLO, although the output between grids will not conflict, when predicting large size or adjacent objects, multiple grids may predict the same object. At this time, YOLO uses a nonmaximum suppression algorithm to filter out redundant output boxes. The confidence of the final output box is equal to the product of the maximum value P of the category prediction of the grid output and the maximum value of the confidence of the current grid output box. This can also filter out some mostly overlapping boxes. The confidence level of the detected object is output, and the box and category are considered at the same time so that the output of the confidence level is more credible.

3.3. Data Enhancement. Aiming at the problem of complex light sources that may appear in the actual detection scene of golf courses and the problem from different viewpoints and distances, we propose a data enhancement strategy to

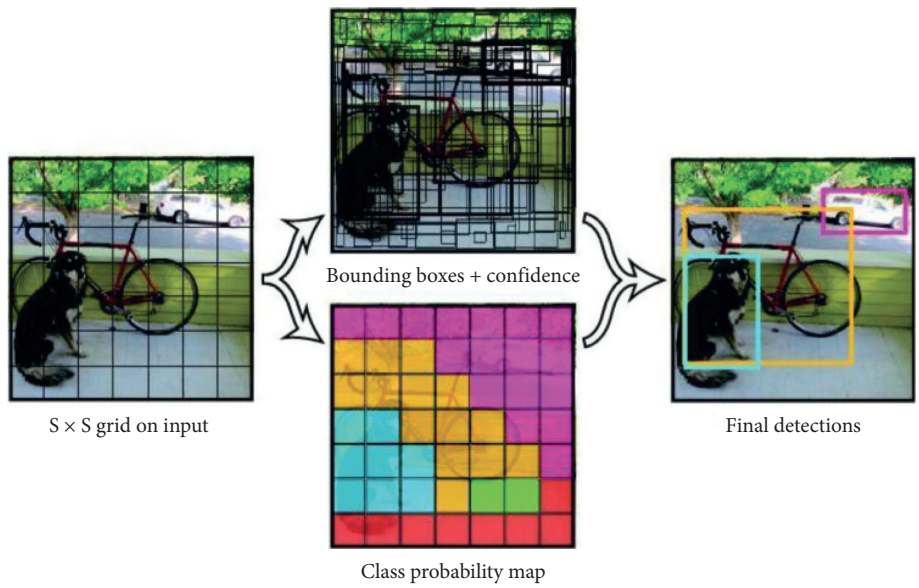

Figure 2: The mechanism of YOLO.

expand the dataset. For the open-air golf situation, where light is relatively strong, the overexposure environment of the camera is simulated by increasing the exposure value as well as the contrast. For underexposure environments such as cloudy days or in the shade of trees, reduce the exposure and contrast of the original image to increase the proportion of dark parts of the image. For the problem of the target object scale that does not need to be brought about by the camera angle and distance, three hybrid data enhancement methods are used: (1) random image translation, which translates the image in the horizontal or vertical direction; (2) random image zoom; (3) random zoom out or zoom in the image.

4. Action Recognition Based on LSTM

Action recognition based on LSTM is to add a recurrent neural network. This kind of hybrid network has the advantages of both CNN and RNN and can obtain information in the time dimension very well. Moreover, LSTM can tackle the problem of RNN due to the disappearance of gradients and cannot handle long-term video well. The problem has shown good results in capturing spatial motion patterns, time-series, and long-term dependencies.

4.1. LSTM Unit. Traditional neural networks, such as convolutional neural networks, have no memory function and cannot pay attention to the relationship between feature information at adjacent moments. Recurrent neural networks (RNN) are mainly used to analyze time-series sequences and can extract the time-series features. RNN combines the hidden layer with the input and outputs it to the hidden layer. In this way, RNN can combine the information at the previous moment to output, and the formula for forward propagation is as follows:

$$\begin{aligned} s_t &= \sigma\left(W_s x_t + U s_{t-1} + b_s\right), \\ o_t &= W_o s_t + b_y\Big), \end{aligned} \tag{9}$$

where W_s is the weight parameter of the input data at the current moment, U represents the weight parameter of the hidden state at the previous moment in the hidden state, W_o is the weight of the hidden state and output at the current moment, b represents the offset, and σ function represents the activation function.

RNN is very successful in tasks such as speech recognition and text generation. RNN can transmit characteristic information from front to back. Theoretically, the long-term dependence relationship can be solved; that is, the output at time t contains all the characteristic information at time 0-t, but this is not the case. Because in the training process, RNN also has the problem of gradient disappearance and gradient explosion, which makes the parameters of the neural network unable to be updated correctly. Therefore, the long-distance information often cannot be transmitted to the subsequent output sequence, which makes the input feature information of the network incomplete and ultimately affects the stability. RNN generally has a relatively good analytical power for data with a short distance between related information.

In order to solve the gradient explosion and disappearance, previous research proposed an LSTM network. LSTM unit records sequence information. The input and output of the LSTM unit are controlled by the switch of the control gate. The input gate can control the current input information to participate in the transmission of memory cells, and the forget gate can control the transmission of previous memory cells. The output gate can control how much information the memory unit can output at the current moment.

Each unit of the RNN needs to be connected in series to ensure that the hidden state of each layer will propagate backward through the network. Compared with the

traditional RNN, the LSTM network has several more door controls. Both the number of parameters and the number of calculations will rise sharply. When the network needs to predict a longer interval, a multilayer parallel network can be used for recognition tasks. In the action recognition task, the output of the task is to assign an action label to the input video sequence, while in the action detection task, it is necessary to output the correct label on all keyframes of the input video sequence. For example, for a 100-frame video sequence, for action recognition tasks, the network only needs to detect any 50 frames of data to output the correct label. In motion detection, if the predicted output tags are not continuous, multiple motion fragments will be generated. This is because LSTM can only transfer the characteristics of timing information in one direction, and the input information at the later stage of the sequence cannot participate in the output at the early stage of the sequence. Through this multilayer bidirectional LSTM network, each output of the sequence can use the input information. In the task of motion detection, discrete motion fragments can be automatically supplemented into continuous motions through multilayer bidirectional LSTM network propagation back and forth. The Softmax layer of the multilayer two-way LSTM network provides a score for each action category, and the behavior of the LSTM network is taken as the score for each action.

4.2. Golf Swing Algorithm. Donahue et al. [28] proposed an LCRN, which can achieve end-to-end training, take full advantage of CNN's strong ability to extract image information, and make the network have the ability to process timing information by adding LSTM. LCRN is composed of input layer, feature extraction layer, LSTM layer, and output layer. CNN is used to process variable-length video single-frame pictures and output the extracted feature maps of the single-frame pictures to the LSTM network. The final output layer produces a variable-length output.

This paper uses an action recognition algorithm based on LSTM. Similar to the method of LRCN, the single-frame picture of the input video obtains the feature map of the single-frame picture through the feature extraction network, and then the feature map is globally averaged and pooled and then input into the two-layer bidirectional LSTM network as shown in Figure 3. The LSTM network can add time-dimensional feature information to the spatial features. The double-layer LSTM network ensures that image features can be transmitted in both directions, and the time feature of each frame of image can fully combine the frame information in the front and back directions. The two-layer LSTM network further integrates the characteristic information of the single-layer LSTM network. After passing through two fully connected layers, LSTM's output is input to the Softmax layer for classification, and finally, the classification result of each frame of the input video sequence is obtained.

Time characteristics are the key to golf swing recognition. Generally, the process from Address (A) to Top (T) is the same as the process from Top (T) to Impact (I), except that the swing direction is opposite. The image features are basically the same. At the same time, athletes often carry out repeated pretargeting process before Address (A), and the video sequence may contain multiple points similar to the Address (A) image feature. It is difficult to distinguish if the time sequence feature is not added. The action recognition network based on the long and short-term memory network can accurately predict the keyframes of the model through the context information of the video sequence by adding the LSTM unit. The spatial features of the output of the feature extraction network pass through the two-layer two-way LSTM network to add timing features, then pass to two fully connected layers, and finally pass through the Softmax classifier to realize the mapping of spatiotemporal features to the posterior probability. The definition of the Softmax function is as follows:

$$S_i = \frac{e^{z_i}}{\sum_j e_j}. \tag{10}$$

Each frame of image will get a posterior probability distribution vector of all types, and the final output form is the mapping $e_t = (p_1, p_2, \ldots, p_c)$ of all event probabilities and the final output of the model $(e_1, e_2, \ldots, e_T)$, where T represents the sequence length and c represents the category of the output frame. This model c has 9 output frame types, including 8 golf swing motion keyframes and 1 invalid frame.

5. Experiments and Discussions

5.1. Evaluation of Golf Detection. The model after the training convergence is tested, and detection accuracy of golf ball, the detection accuracy of the golf club head, the average accuracy of the model, and the average detection speed of a single picture are, respectively, detected. The detection result is illustrated in Figure 4.

mAP of Faster R-CNN is 73.7%, and mAP of YOLOv3 is 62.7%. From the experimental data, it can be seen that Faster R-CNN has a better performance than YOLOv3. Because Faster R-CNN's RPN network can extract more prior frames, its recall rate is higher than that of YOLOv3. However, because the two-stage target detection algorithm must first extract the candidate frame and then classify according to the features in the classification frame, it needs to enter two classifiers, and the calculation amount is much larger than YOLOv3, so its detection speed is also much slower than YOLOv3. The detection speed of YOLOv3 is 0.03 s.

Use the data enhancement scheme to expand the data, and train Faster R-CNN with the expanded data. The detection result is illustrated in Table 1; using the enhanced dataset for training, the accuracy of target detection under complex light sources has been improved, and mAP of Faster R-CNN has increased by 4.3%. Especially under complex light sources, the detection performance of the model trained with enhanced data has been significantly improved. But for the golf ball in the distant state, due to the small size, the detection rate of the target object is still very low.

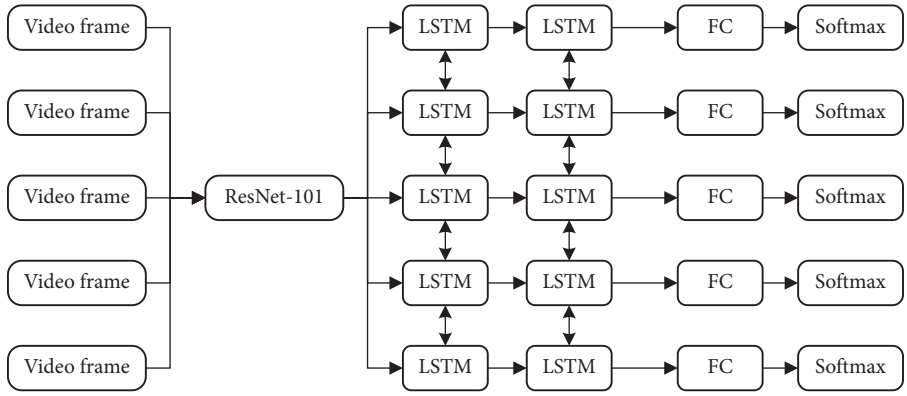

Figure 3: Swing action recognition network.

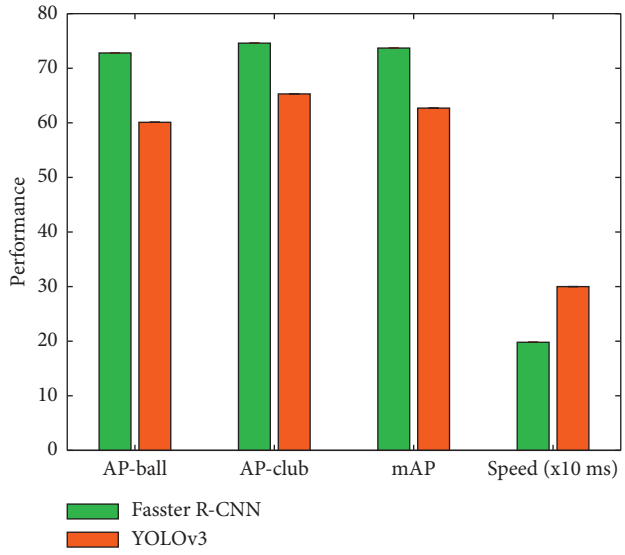

Figure 4: Detection performance comparison between FasterR-CNN and YOLOv3.

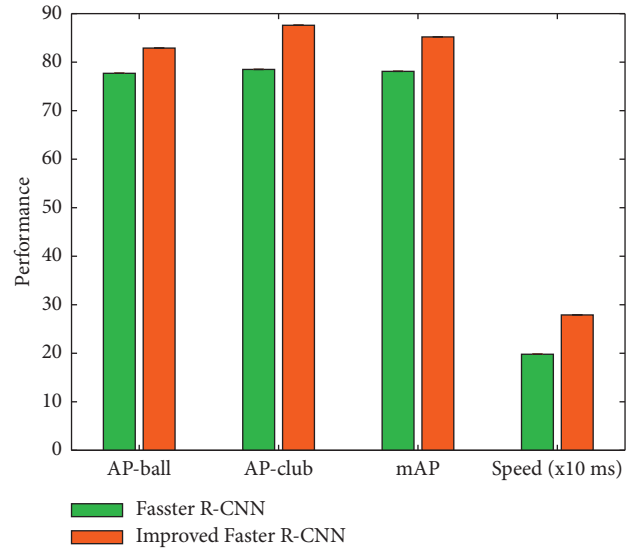

Figure 5: Target detection accuracy of the improved model.

Table 1: The impact of data augmentation on the accuracy of model detection.

Model	Data enhancement	mAP
Faster R-CNN	Yes	73.7
Faster R-CNN	No	78.0

This paper uses ResNet-101 instead of VGG-16 in the source code as the backbone network of the target detection algorithm and trains the improved Faster R-CNN model after data enhancement. The detection performance is verified on the test set as shown in Figure 5.

ResNet-101 can extract higher-dimensional image feature information, and the improved Faster R-CNN's mAP is as high as 85.2%. The CNN model has increased by 7.2%. Due to the deeper network structure of ResNet-101, the parameters in the network have also increased exponentially. However, due to the role of the residual module, the average of a single image can be seen from the detection results. The detection speed is only 0.081 seconds slower than the original structure. Because the motion assistance system does not require high real-time performance, the use of the improved Faster R-CNN can significantly improve the performance.

5.2. Evaluation on Action Recognition. The action recognition algorithm based on the LSTM is very important for the selection of the input sequence length T. This paper uses different sequence lengths T for experimental records, and the experimental result is illustrated in Figure 6. Under the condition of the other parameters unchanged, check the impact on the accuracy of the model. Affected by the hardware performance of the machine, when the sequence length T is too large, some comparative experiments need to reduce the batch size to ensure the GPU memory space. Batch processing can improve the training speed of the model and make full use of GPU parallel computing capabilities, relying on the advantages of weight sharing of convolutional neural networks; through one calculation, multiple input video samples are calculated in parallel, which greatly improves the training efficiency. However, batch processing is only helpful for the improvement of model training speed, and the size of

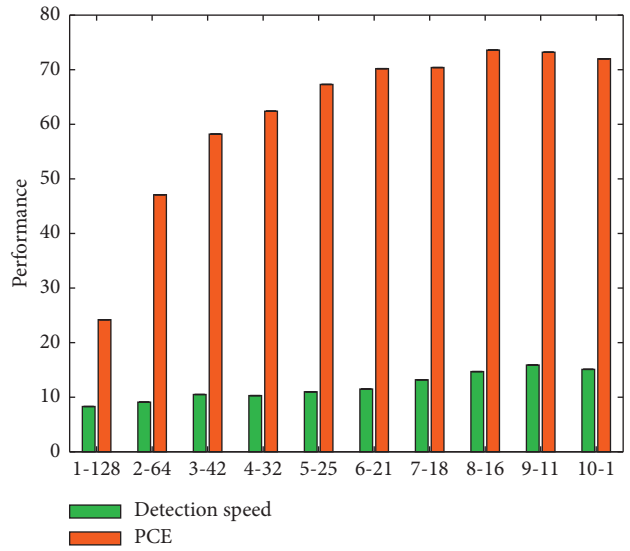

Figure 6: The influence of different T and B on accuracy and detection speed.

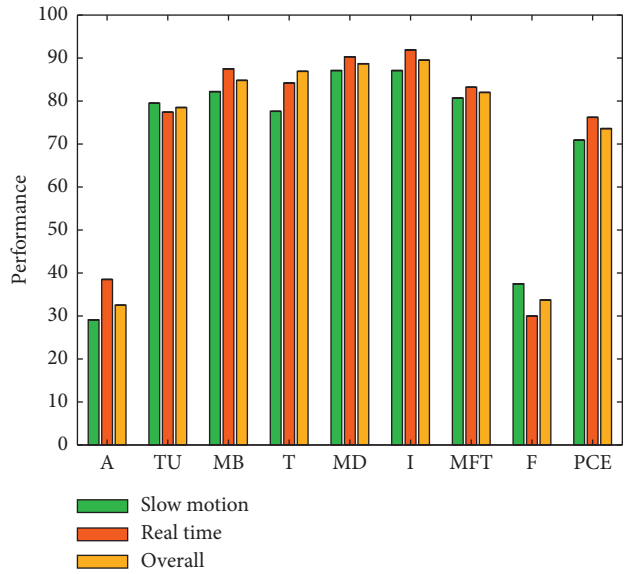

Figure 7: Detection accuracy of different keyframes.

batch processing has relatively little effect on the accuracy of model detection.

In the real-time state and slow-motion state, the detection efficiency of each keyframe of the action recognition model is tested. Figure 7 provides the detection accuracy of each keyframe of the model in the GolfDB test set. The PCE of the keyframe and the average value of the overall PCE are, respectively, detected for the slow-motion and real-time video data. In general, the overall performance of the model recognition is good, and the accuracy of the action recognition is 73.4%. Experiments have found that the detection rate of Address (A) and Finish (F) keyframes is worse than other frames. These frames have two common characteristics. First, the swing speed of the clubhead in and around the frame is relatively low. Since Address (A) is the beginning of the golf swing, its initial speed is 0, Finish (F) is the end of the swing, and its final speed is also 0. Secondly, the frame can only use the feature information in one direction. For the Address (A) frame, the image before the frame has no annotation information, and for the Finish (F) frame, there is no annotation information for the subsequent images. These two factors make it difficult for the model to accurately locate Address (A) and Finish (F) in time.

Experiments have found that the detection rate of Top (T) relative to adjacent frames is also relatively low, which may be due to the change in speed direction when the golf club is lifted to the highest point. The club speed of the surrounding frames is also relatively low, but because the golf club head stays at the top for a relatively short time and the distance between the Top (T) and the surrounding keyframes is relatively small, the Top (T) frame is compared with the Address (A) and Finish (F) which have a relatively little reduction in detection performance. At the same time, because the front swing is generally faster than the backswing, the frame detection performance of the front swing is generally better than the backswing. Compared with slow-motion samples, the detection efficiency of keyframes in each stage of real-time video is also more accurate, which further verifies the assumption that the model is sensitive to speed.

6. Conclusions

This paper studies the golf detection method and the golf swing motion recognition method for video sequences. Faster R-CNN and YOLOv3 models are trained using the golf dataset. Through comparative experiments, Faster R-CNN has better detection than the YOLOv3 algorithm. mAP of Faster R-CNN is 73.7%, and the average speed of a single image is 0.19s. The detection accuracy of YOLOv3 reaches 62.7%, and the average detection speed is 0.030s. Aiming at the complex lighting background that may appear in the golf course, the dataset is enhanced, and the open-air glare environment of golf is simulated by increasing the exposure and contrast of the picture by reducing the exposure and contrast while increasing the dark information method to simulate the insufficient light environment such as the shade of the stadium. By zooming and rotating the image, it simulates the scale problem of the target object in the image taken by the mobile phone or camera at different distances and different angles. Using ResNet-101 instead of VGG-16, ResNet-101 has a deeper network structure and can extract richer information. For action recognition algorithms, this paper proposes an action recognition network based on LSTM. The spatial information of each frame is input into the two-layer two-way LSTM, and the temporal features can be extracted. Finally, temporal and spatial features are input into the Softmax classifier to classify each frame of the image. According to different input sequence lengths, the final accuracy of the model is analyzed experimentally. For the difference in keyframe detection accuracy, the actual scene of the golf swing is analyzed. The network is trained on the GolfDB dataset, and the final detection accuracy of the model is 73.6%. With a tolerance of about 3

frames, the performance of the action recognition algorithm is as high as 88.5%.

Data Availability

The datasets used are available from the corresponding author upon reasonable request.

Conflicts of Interest

The author declares that he has no conflicts of interest.

References

[1] J. N. Roemmich, E. J. Richmond, and A. D. Rogol, "Consequences of sport training during puberty," *Journal of Endocrinological Investigation*, vol. 24, no. 9, pp. 708–715, 2001.

[2] N. Zulkifli, F. Harun, and N. S. Azahar, "XBee wireless sensor networks for heart rate monitoring in sport training," *Applied Physics A*, vol. 106, no. 2, pp. 295–307, 2012.

[3] V. A. Zaporozhanov, T. Borachinski, and Y. N. Nosko, "Assessment of children's potentials in dynamic of initial stage of sport training," *Journal of Physical Education & Sport*, vol. 15, no. 3, pp. 525–530, 2015.

[4] R. M. Matina and A. D. Rogol, "Sport training and the growth and pubertal maturation of young athletes," *Pediatric Endocrinology Reviews: PER*, vol. 9, no. 1, pp. 441–455, 2011.

[5] J. Pei, K. Zhong, J. Li, J. Xu, and X. Wang, "ECNN: evaluating a cluster-neural network nodel for city innovation capability," *Neural Computing & Applications*, pp. 1–13, 2021.

[6] A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, P. Martinez-Gonzalez, and J. Garcia-Rodriguez, "A survey on deep learning techniques for image and video semantic segmentation," *Applied Soft Computing*, vol. 70, Article ID S1568494618302813, 2018.

[7] C. Shang and Y. Zhang, "Combining Newton interpolation and deep learning for image classification," *Electronics Letters*, vol. 51, no. 1, pp. 40–42, 2015.

[8] F. Xiang, G. Ding, J. Su, W. Zhang, J. Wu, and X. Gu, "Dangerous target recognition of massive image and video based on deep learning," in *Proceedings of the e2019 Chinese Automation Congress (CAC)*, IEEE, Hangzhou, China, February 2020.

[9] M. Iliadis, *Sparse Representation and Deep Learning for Image and Video Reconstruction*, Northwestern University, Evanston, IL, USA, 2016.

[10] X. Zhang and J. Xiang, "Moving object detection in video satellite image based on deep learning," *Lidar Imaging Detection & Target Recognition*, vol. 2017, Article ID 10605, 2017.

[11] C. Smith, J. Cook, C. Bradley, R. Gosett, and R. Heynas, *Enhancing Deep Learning in Sports Science: The Application of Rich Media Visualization Techniques in Mobile and Reusable Learning Objects*, in *Proceedings of the EdMedia + Innovate Learning*, Vancouver, Canada, June 2007.

[12] H. Song, X. Y. Han, C. E. Montenegro-Marin, and S. Krishnamoorthy, "Secure prediction and assessment of SPorts injuries using deep learning based convolutional neural network," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 6, pp. 1–12, 2021.

[13] H. Cui and C. Chang, "Deep learning based advanced spatio-temporal extraction model in medical sports rehabilitation for motion analysis and data processing," *IEEE Access*, vol. 8, p. 1, 2020.

[14] H. Kaiyan, W. Qin, Research on 020 platform and promotion algorithm of sports venues based on deep learning technique," *International Journal of Information Technology and Web Engineering*, vol. 13, no. 3, pp. 73–84, 2018.

[15] M. Hagenbuchner, D. P. Cliff, S. G. Trost, N. V. Tuc, and C. E. Peoples, "Prediction of activity type in preschool children using machine learning techniques," *Journal of Science and Medicine in Sport*, vol. 18, no. 4, pp. 426–431, 2014.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[17] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 580–587, Columbus, OH, USA, June 2014.

[18] S. Ren, K. He, R. Girshick, and R.-C. Faster, "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.

[19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the Computer vision & pattern recognition*, pp. 779–788, IEEE, Las Vegas, NV, USA, June 2016.

[20] J. Redmon and A. Farhadi, "YOLO9000: Better, Faster, Stronger," in *Proceedings of the IEEE Conference on Computer Vision & Pattern Recognition*, pp. 6517–6525, IEEE, Honolulu, HW, USA, July 2017.

[21] K. Simonyan and A. Zisserman, "Two-stream convolutional networks for action recognition in videos," *Advances in Neural Information Processing Systems*, pp. 568–576, 2014, https://arxiv.org/abs/1406.2199.

[22] L. Wang, Y. Xiong, Z. Wang et al., "Temporal segment networks: towards good practices for deep action recognition," in *Proceedings of the European Conference on Computer Vision*, pp. 20–36, Springer, Amsterdam, The Netherlands, October 2016.

[23] D. Tran, L. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Learning spatiotemporal features with 3D convolutional networks," *IEEE International Conference on Computer Vision. IEEE*, pp. 4489–4497, 2015.

[24] J. Carreira, A. Zisserman, and Q. Vadis, "Action recognition? A new model and the kinetics dataset," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6299–6308, IEEE, Honolulu, HW, USA, July 2017.

[25] Z. Qiu, T. Yao, and T. Mei, "Learning Spatio-Temporal Representation with Pseudo-3D Residual Networks," in *Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 5533–5541, IEEE, Venice, Italy, October 2017.

[26] Z. Cai, H. Neher, K. Vats, D. Clasui, and J. Zelek, "Temporal hockey action recognition via pose and optical flows," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, IEEE, Long Beach CA, USA, June 2019.

[27] S. Giancola, M. Amine, and T. Dghaily, "Action Spotting in Soccer Videos," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1711–1721, IEEE, Salt Lake City, UT, USA, June 2018.

[28] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, and K. Sea, "Long-term recurrent convolutional networks for visual recognition and description," *Proceedings of the IEEE conference on computer vision and pattern recognition*, vol. 39, pp. 2625–2634, 2015.