Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 9259328, 11 pages
https://doi.org/10.1155/2022/9259328

Research Article

Visualization of Football Tactics with Deep Learning Models

Xin Zuo

College of Physical Education, Henan University of Science and Technology, Luoyang, 471000 Henan Province, China

Correspondence should be addressed to Xin Zuo; 9901975@haust.edu.cn

Received 29 March 2022; Accepted 22 April 2022; Published 2 June 2022

Academic Editor: Kuruva Lakshmanna

In the last several years, computer vision tasks involving visual identification and tracking have seen a rise in the usage of deep learning technologies in recent years. An extremely difficult but rewarding endeavor is identifying and following football players' targets. This may be used to study football tactical visualization. Due to the similar appearance and frequent occlusion of targets in football video, traditional methods often can only segment targets such as players and balls in the image but cannot track them or can only track them for a short time. Based on the related research of computer vision and deep learning, using several cameras, this study develops a system that can properly monitor many targets in a football stadium for a lengthy period of time. The main research contents of this paper are as follows: (1) a CNN for target displacement prediction is proposed, which no longer relies on the previous linear motion model or quadratic motion model, so that the multitarget tracking algorithm can be applied to more scenes. (2) For the first time in a multitarget tracking algorithm, a continuous conditional random field is used to model the asymmetric nature of the target relationship. At the same time, the CNN for target displacement prediction can be cascaded with the continuous conditional random field for end-to-end training, which greatly reduces the training difficulty. The parameters of the experiment in this paper are simple, and comprehensive and systematic experiments verify the validity and correctness of this work from different aspects.

1. Introduction

Humans rely heavily on their sense of sight to take in information from the outside world, with vision accounting for the vast majority of what we learn. As computers become more widely available and widely used, an increasing number of individuals are using them in their day-to-day lives and work. Computers enhance the human mind's ability to think and perceive. Computer vision is the result of computers simulating human visual perception skills [1]. By using cameras and computers to capture and interpret visual information, computer vision is a study that was aimed at making it possible for computers to do certain human visual tasks. Math, image processing, biology, and computer science are just few of the fields that come together in computer vision. When it comes to computer vision, video object tracking is a prominent research issue, both domestically and internationally [2]. Using video target tracking, a computer constantly infers the location of a target in the video. Identifying and tracking the target in each frame of the video is its primary goal, as is providing the target's position at every point in the video, the whole of the region. Since its invention, video target tracking has found widespread usage in civilian and military settings alike, with applications as diverse as video surveillance, human-computer interaction, robot visual navigation, virtual reality, intelligent transportation, and medical diagnostics. In our daily life, sports video is a very important entertainment video, which occupies a considerable proportion in people's entertainment life. Football video is a very wide-ranging sports video [3–5]. In order to keep the audience's attention, football films frequently include a variety of special effects. Additionally, mobility data is needed for useful analysis by coaches. Real-time motion data recording is also desired by players. The path of motion is deduced. To meet these requirements in professional football, high-cost people and material resources are required [6–8]. Using a low-cost target identification and tracking system in a larger-scale amateur football game has significant practical value and relevance. As a result, computer vision tasks involving visual identification and tracking have seen a rise in the usage of deep learning technologies in recent years. On this basis, the implementation of deep learning technology has become an urgent problem to

be solved in the next step. This topic is based on football video, all players are tracked in this scene, and the movement trajectory of each player in normal games is extracted, and then, the research and analysis of football tactics are carried out. In the past, for this task, it was often used to record the player's movement trajectory by attaching a chip to the player, but this method was costly and could not output the player's movement data in real time. The hardware of this subject only uses the camera, which can accurately extract the player's entire motion data and output it in real time, which greatly saves the cost, improves the user experience, and can be well applied in the real scene.

The contributions of this work are as follows: (1) a CNN for target displacement prediction is proposed, which no longer relies on the previous linear motion model or quadratic motion model, so that the multitarget tracking algorithm can be applied to more scenes. (2) For the first time in a multitarget tracking algorithm, a continuous conditional random field is used to model the asymmetric nature of the target relationship. At the same time, the CNN for target displacement prediction can be cascaded with the continuous conditional random field for end-to-end training, which greatly reduces the training difficulty.

2. Related Work

According to the different camera placements, the current player detection techniques in football videos could be divided into two categories: fixed camera-based and moving camera-based [9]. The picture taken by the former was in a fixed scene, so the background was unchanged or only slightly changed; the picture taken by the latter was in a dynamic scene, and the lens was always following the target area of interest for shooting, so its background was changing. Reference [10, 11] installed 4 fixed cameras on one side of the field, and reference [12] installed 6 fixed cameras evenly distributed on both sides of the field, using the background subtraction method to detect players. The video shot by the moving camera usually referred to the live football video on the TV. The lens of this kind of video would have changes such as panning, zooming, and rotation so that the lens always followed the area of interest, and the complete live football video usually had switch lens; these lenses included telephoto, medium, and close-up. There were two main types of existing site extraction methods: nonparametric methods and parametric methods [13–20]. For the nonparametric method, in fact, the main color of the site was modeled to represent the area of the site pixel; the difference was the color space selected and the adaptive degree of the algorithm. The bottom-up technique and the top-down method were the two basic concepts behind contemporary moving target tracking technology. To track a target, the former did not depend on past knowledge but rather derived motion information straight from the picture sequence. This kind of method could achieve better results only when the camera was fixed; the latter depended on the constructed model or prior. Knowledge was achieved by performing matching operations in the image sequence or solving to obtain the posterior probability. The matching operation was to obtain the best match by solving the similarity between the target model and the candidate target, and to solve the posterior probability, the maximum posterior probability was selected. The corresponding state vector was used as the current state of the target [21]. There were now four types of tracking algorithms: model, region, contour, and feature tracking [22]. Nonrigid objects could not be tracked using model tracking since the form of the target item had to be modelled and tracked in successive picture frames. To monitor a moving target, one may use area tracking or contour tracking. Region tracking used color information from a specified area, but this approach failed when the target was obscured. Contour tracking used a closed curve contour to represent the moving target; however, initializing the curve was challenging. Instead of using the whole target as the tracking object as was the case with model-based or region-based tracking, feature-based tracking isolated a few key characteristics to utilize as the tracking object. Reference [23] proposed a multitarget tracking algorithm based on motion detection for the tracking problem of people in video sequences. The core of the tracking algorithm was the establishment of an association matrix. When the number of targets between consecutive frames was constant, the correlation matrix based on the centroid distance was used. When the number of targets changes, the correlation matrix based on the intersection area was used corresponding to different situations such as target merger, separation, disappearance, and new addition. The intersecting area was used to judge the occurrence of occlusion, and different processing strategies were adopted according to whether the occlusion was serious or not. However, this method was carried out on the premise that the above 4 situations did not occur at the same time, and the relationship between each target of the current frame and each track of the previous frame needs to be calculated, and the amount of calculation was too large. Reference [24] also studied the multitarget tracking algorithm for pedestrians in video sequences. In this paper, the mean-shift algorithm was used to achieve normal tracking, and the area was used as the occlusion factor to judge the occurrence of occlusion. However, when the target and the background color were indistinguishable, the tracking effect of the mean-shift method was not good. The references [25–28] all adopted the multitarget tracking algorithm based on particle filter, but the calculation amount of particle filter was too large and the real-time performance is not good; especially when there were many targets to be tracked, the calculation amount increased exponentially. Player tracking in football videos also belonged to multitarget tracking, and the methods used in previous work were also different.

When there was no occlusion, region-based detection was used to achieve tracking; when occlusion occurred, template matching, histogram back-projection, or merge-split method were selected according to different situations. In reference [29], each detected player was initialized with a particle filter, and the color information of the player was used to achieve tracking, which could obtain a better tracking effect, but the particle filter calculation was relatively large, when the player needs to be tracked. When the number increased, the real-time performance would be very

poor, and the method in this paper could not solve the problem of occlusion between players on the same team. In references [30, 31], the author defined a state vector for each player in the tracking queue, which contained the player's center of mass, velocity, circumscribed rectangle, label of the occluded player, and number of categories of the team, and the players were not occluded, left, entered, disappeared, etc., and then, the maximum a posteriori probability method was used to obtain the optimal state and observation sequence. There was no explanation for how to judge various states, and the method in this paper could only deal with the occlusion of players of different teams. In reference [32], player tracking was implemented by Markov Monte Carlo data association, and some data-driven models were used to improve the efficiency of Markov chains, but the problem of occlusion between players was not discussed in depth. Existing deep learning-based target tracking algorithms mainly focused on the single-target domain, because deep neural networks could learn more effective visual representation features than hand-designed features to discriminate background and target. Reference [33] introduced the ECO algorithm, which converts high-dimensional features into compressed dimensions through factorization operations, reducing training parameters and complexity. Reference [34] proposes an end-to-end deep architecture that incorporates geometric transformations into a correlation filter-based network. The architecture introduces a novel spatial alignment module that provides continuous feedback for transforming objects from boundaries to centers with normalized aspect ratios. This allows correlation filters to work on aligned samples for better tracking. Literature [35] proposes the RPCF algorithm, which is the first time to introduce ROI-based pooling in the relevant filter formula. It proposes a correlation filter algorithm with equality constraints, which can equivalently implement ROI-based merging operations without actual ROI sample extraction. The learned filter weights are less sensitive to overfitting problems and more robust.

3. Method

3.1. Single-Target Tracking Algorithm Based on Deep Learning. Target tracking algorithms can be divided into single-target tracking algorithms and multitarget tracking algorithms according to the number of tracking targets. The single-target tracking algorithm can be considered as a simplified version of the multitarget tracking algorithm, that is, a multitarget tracking algorithm that does not consider the interaction between targets. The general single-target tracking algorithm generally marks a target manually by a human in the first frame, and then, the tracking algorithm needs to follow the object firmly in the next video sequence. In recent years, with the development of deep learning, traditional single-target tracking algorithms based on correlation filters have been gradually abandoned, or some main modules have been replaced with deep networks. At the same time, the single-target tracking algorithm based on deep learning has occupied the top of the major single-target tracking data lists for a long time. Some algorithms can not only achieve good results but also achieve real-time tracking speed. Next, three deep learning-based single-target tracking algorithms are introduced, including MDNet with good effect, GOTUIRN with fast speed, and SiameseFC network with balanced effect and speed.

3.1.1. MDNet. MDNet was the winner of the 2015 visual object tracking challenge, and its introduction marked the beginning of CNNs to guide single-object tracking algorithms. Most of the previous deep learning-based tracking algorithms only used models trained on some general data as feature extractors, and the results confirmed that doing so can indeed improve the tracking effect. But soon everyone found that if there is not enough labeled tracking data, the effect of the model trained based on general data is limited. Different from multitarget tracking, the targets of single-target tracking are often of different types, and for the tracking problem, there should be some commonality between different targets. At the same time, there are some conflicting definitions, such as the first video needs to track a person, and the second video needs to track a car, and then, the person belongs to an irrelevant background in the second video. This conflicting definition often makes the convergence of the CNN unstable, and the effect is not improved. At the same time, different classes of objects have different colors, movement patterns, and shapes. Based on the above shortcomings, South Korea's Pohang University of Science and Technology proposed a new single-target tracking architecture. This network is called a multidomain adaptive network, namely, MDNet. MDNet learns generic scene patches from multiple annotated video sequences. Then for each video, class-dependent blocks are learned. Through this optimization method, the network can not only extract common visual features between objects but also optimize for different types of objects. However, the tracking time of MDNet is proportional to the number of tracking targets, making it unfavorable for real-time tracking.

The specific training method is as follows. The target position of the previous frame is the center, and 256 candidate regions are sampled through Gaussian distribution, and then, the size of each region is unified to 107×107 pixels and then sent to the network. The general scene block of the network is composed of three convolutional layers stacked, which mainly refer to the design points of the VGG-A network. The category correlation block consists of three fully connected layers, the first and second fully connected layers are used as category feature extraction operators, and the third is a binary classifier. Assuming that k videos are used for training, there are k category-related blocks. When testing, for a new video, a new category-related block will be created, the general scene block will be fixed, and the newly created category-related block will be updated online. Online update is divided into long-term and short-term two ways. When the confidence of the tracked sample is higher than the set threshold, the sample will be added to the training positive sample. The long item corresponds to an update in historical time, that is, a fixed time interval. The short term corresponds to the number of samples, that is, the upper limit of the total number of

fixed samples. When the score is below 0.5, the sample is discarded. The negative samples for training are obtained through short terms, and the hard negative mining technique is also used in the process of negative sample generation. Selecting difficult samples as negative samples will make the network more discriminative. MDNet has carefully designed the network structure for the single-target tracking task, trained with a large amount of data and fine-tuned online. When selecting candidate regions, changes in different scales are also considered. The only problem is that the real-time performance is poor. However, it has successfully guided the single-target tracking task to the field of CNN, which can be said to be an epoch-making work.

3.1.2. SiameseFC. From the perspective of template matching, SiameseFC takes advantage of the powerful feature expression capabilities of CNNs and takes a big step forward in the field of single-target tracking. The core of the work is to obtain a model for similarity comparison through a large amount of data training. Compare the similarity between the target and the template, and the most similar position is the position of the tracking target. The specific calculation is as follows:

$$f(p, n) = \mu(p) + \mu(n) + d, \tag{1}$$

where p, n represent the template image and the current frame, respectively, μ is the feature extraction operation of CNN, d is the offset, and $f(p, n)$ is the output similarity response graph. Find the point with the greatest similarity in the graph, and use it as the position of the target in a new frame of image.

The training steps of SiameseFC are also relatively simple, and the reverse iterative optimization is performed through the loss function. But it is worth noting that the SiameseFC network needs to be pretrained on a large-scale dataset, that is, to find a general feature description operator and then do some fine-tuning on the target tracking video. The authors use images from the large-scale vision challenge (ImageNet) dataset to search for labeled regions across the entire image. The author believes that in the training process, the classification of the target is not considered, which is conducive to learning a network with stronger generalization ability for tracking any target. Experiments show that this training method can achieve better results. SiameseFC greatly improves the speed while ensuring the accuracy and almost meets the real-time requirements. At the same time, the network adopts a fully convolutional structure, which means that there is no need for a fixed size input in the actual tracking process. SiameseFC pioneered the application of the siamese structure to the field of single-target tracking, and many subsequent works were based on this and improved in different directions. It is precisely because SiameseFC starts from template matching that it is easy to fail for suddenly changing targets. At the same time, when the discrimination is not strong enough to cause multiple similar objects to appear at the same time, the tracking will fail.

3.1.3. GOTURN. Most of the above-mentioned single-target tracking algorithms adopt the method of pretraining and online training. In practical application scenarios, online training methods generally do not meet the requirements of the industry for the speed of target algorithms. Therefore, a truly practical single-target tracking algorithm should have the following characteristics. First, a pretrained model can obtain better results on new datasets. The second is to meet the real-time performance. GOTURN was proposed under this need. First of all, it adopts the offline learning method; that is, it does not fine-tune the network parameters on the test set. Secondly, it can get a speed of 100 frames per second on the GPU, and the real-time performance is better. Once GOTURN was launched, it gained a lot of attention. The idea of GOTURN is very simple: input the current frame and the previous frame, and directly predict the position of the target frame. Like SiameseFC above, it is assumed that the motion of the object in the video on the image plane obeys a Gaussian distribution; that is, the probability of the current target near the target position in the previous frame is much greater than the possibility of being far away. First, target alignment and cropping are performed on the previous frame of image; that is, the target is placed in the middle of the cropping-block, and the size of the cropping-block and the target size are in a fixed ratio. After obtaining the coordinates of the cropped block of the previous frame of image, use the same method to crop out an image block of the same size at the same position in the current image. Then, the two image blocks enter the convolutional layer, respectively, and then along the direction of the channel, the features are cascaded and finally sent to the fully connected layer to obtain the final regression result. GOTURN and SiameseFC are very similar in architecture; for example, the cross-correlation layer used by SiameseFC can be simulated by a fully connected layer. During offline training, GOTURN mainly uses pictures and videos, and the network directly returns the coordinates of the target. GOTURN also models the motion of the object simply, as follows:

$$c_x' = c_x + w \times \Delta x, \tag{2}$$

$$c_y' = c_y + h \times \Delta y, \tag{3}$$

$$w' = w \times \gamma_w, \tag{4}$$

$$h' = h \times \gamma_h, \tag{5}$$

where Δx and Δy refer to the relative displacement of the object and γ_w and γ_h refer to the relative deformation of the object. The authors also found that the motion increments in the x and y directions follow a Laplace distribution with a mean of 0, while the relative deformation γ_w and γ_h follows a Laplace distribution with a mean of 1. However, due to this assumption, GOTURN is not well suited to the target of severe mutation. According to the Laplacian distribution characteristics, the probability of the occurrence of a target with a larger movement displacement is small, so when generating samples, the displacement data is also generated according to the Laplacian distribution characteristics.

3.2. Multitarget Tracking Algorithm Based on Deep Learning. Historically, one of the most active areas of research in the field of computer vision has been the development of multitarget tracking algorithms based on deep learning. As an extension of the single-target tracking algorithm, the multitarget tracking algorithm incorporates the constraints imposed by the relationship between the targets, resulting in a more complex tracking method. The tracking of many targets is more difficult than the tracking of a single target. Single-target tracking can be thought of as the process of filtering and searching for a target on a piece of continuous data that has been segmented. Multiple candidate regions are used to achieve target matching, and the quality of data association is what ultimately decides the advantages and disadvantages of using the multitarget tracking algorithm. The multitarget tracking algorithm based on deep learning will be discussed in detail in the next chapter, and two different types of multitarget tracking algorithms based on deep learning will be examined and compared in the following chapter. Using the standard CNN as the foundation, the first algorithm, multitarget tracking, is created, which merely makes use of the DNN's feature extraction capabilities. The second application is the use of RNNs to perform time series feature information fusion using time series data.

3.2.1. Spatiotemporal Attention Algorithm. The spatial-temporal attention mechanism (STAM) is an extension of the single-target tracking algorithm to the multitarget tracking field, as shown in Figure 1. It adds spatial and temporal attention mechanisms to the CNN based on solving a single target and achieves good results. STAM has two particularly important contributions. The first is that it improves the framework of single-target tracking algorithms and proposes a framework that can be efficiently used in multitarget tracking algorithms. Second, it proposes a spatial and temporal attention mechanism to control the update of training samples in order to solve the problem of interaction between targets. STAM adopts a common single-target network architecture, uses a pretrained VGG network as a feature extractor, samples the area near the target of the previous frame image according to the motion information, and then collects regional pooling features and sends them to a three-layer binary classification device. The specific training method is to first obtain the position of the person in the current detection frame, such as the upper left corner, through the segmentation subnetwork. After obtaining the position of the human body, the visible area of the region of the human body is calculated. When the visible area is higher than a threshold, it is considered a positive sample and is used to update the network's classifier. At the same time, when the visible area is lower than a threshold, it is considered as a negative sample and used to train network parameters. When the visible area is lower than the high threshold and higher than the low threshold, it is considered that the person is blocked or interacting with other people. At the same time, it also smooths the templates saved in history according to the size of the visible area. Each target has a dedicated branch of convolutional layers, which is updated by a loss function of training positive and negative samples of the current frame and past frames. At the same time, the motion model of each target is updated according to the historical trajectory state of the target, and finally, the attention weighted smoothing is performed. Although the STAM tracking effect is better, it needs to train a subnetwork for each target. When there are too many tracking targets, there are too many convolutional subnetworks, a large amount of calculation, and a long tracking time, which is not conducive to application in practical scenarios.

3.2.2. Attention Mechanism. The feature extraction in the previous section only extracts wide-area features and does not make certain features. For example, for partially occluded objects, only the features of unoccluded areas are useful. STAM proposes a visual confidence response map $V(r_t^i)$; the specific definition is as follows:

$$V\left(r_t^i\right) = f\left(\lambda\left(C_t^i\right)\right), \tag{6}$$

where λ represents feature extraction through region pooling operation and f represents CNN, which is used to predict a visual confidence response map of the same size as the feature, where the maximum value of the response map is 1 and the minimum value is 0. Then, the spatial attention mechanism is defined as follows:

$$f_V\left(C_t^i\right) = V\left(r_t^i\right) \times \lambda\left(C_t^i\right), \tag{7}$$

where $f_V(C_t^i)$ is the weighted target feature, which can be sent to the binary classifier for learning.

In order to further improve the tracking effect of occluded objects, STAM also designs a temporal attention mechanism $f_T(C_t^i)$, which is specifically defined as follows:

$$f_T\left(C_t^i\right) = \frac{1}{T}\sum_{t=1}^{T} V\left(r_t^i\right) \times \lambda\left(C_t^i\right). \tag{8}$$

That is, the feature map is smoothed to a certain extent in time.

3.3. Multicue RNN Algorithm. The occlusion problem, in particular long-term occlusion, is the most challenging obstacle for all target tracking algorithms to overcome. Due to the occlusion, the appearance model is no longer valid. The only way to forecast the approximate position of the target is to depend solely on its motion model or on its interaction with another target. The target tracking algorithm of multicue RNN provides an online target tracking algorithm that is capable of fusing many bits of information together at the same time. This method models the interaction information between the surface properties of the item, the motion model, and its target using a recurrent neural network. A combination of different types of information is used to solve the problem of tracking chain association

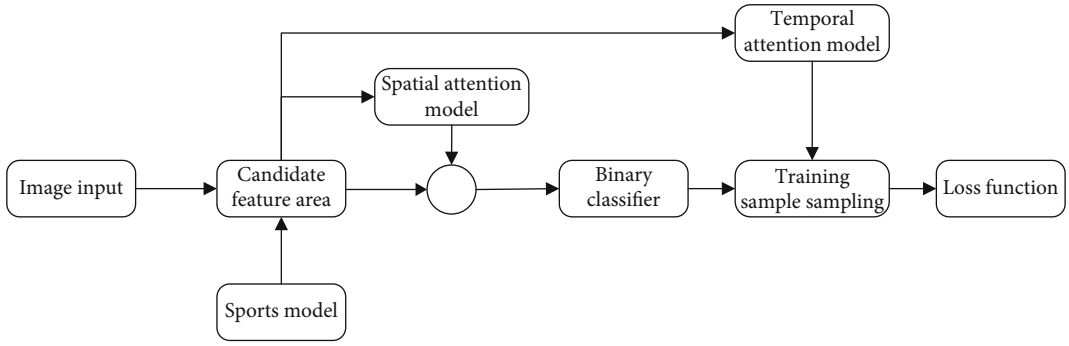

Figure 1: STAM frame diagram.

mistakes or to solve the problem of occlusion in the case of a combination of different types of information. The observation was reestablished, and a more favorable outcome was reached. AMIR may be taught from the beginning to the end without the need for the various hyperparameters that are required by typical multiobject tracking methods. In addition, because it takes into account the entire trajectory of the target history, the motion model based on the RNN proposed by it can be more accurate than traditional linear motion modeling or quadratic motion modeling because it takes into account the entire trajectory of the target history and because the nature of the RNN is such that the weight of the target trajectory closer to the current time is smaller than the weight of the target trajectory farther from the current time, resulting in even greater accuracy. This strategy can be thought of as a device for focusing attention. As an additional proposal, it presents an RNN based on the interaction between objects, which may be used to anticipate and track objects across long distances in busy environments.

3.3.1. Appearance History Model. Appearance model is used to calculate the matching degree between target appearances. AMIR defines a long short-term (LSTM) memory network for historical appearance information matching. The implementation details are as follows. The first step is to train a CNN using the pedestrian reidentification dataset and then remove the discriminative part, that is, the fully connected layer, leaving the convolutional layer as the feature extraction operator. The second step is to extract the appearance features of the tracking target on each frame, that is, to cut the target from the background and send it to the CNN in the previous step to obtain the appearance features, which is a 500-dimensional feature vector. Finally, LSTM is used to exchange time series information, and the historical appearance feature vector is obtained, which is cascaded with the appearance feature vector of the current frame detection frame and sent to the discriminator.

3.3.2. Motion Model. The motion model is used to reduce the target matching search space, that is, to predict the target position of the current frame through the historical speed. Unlike the linear motion model or spline interpolation model used by traditional methods, AMIR also uses an LSTM network for historical velocity modeling. Define the speed of tracking target i at time t as

$$v_i^t = \left(v_{x_i}^t, v_{y_i}^t\right) = \left(x_i^t - x_i^{t-1}, y_i^t - y_i^{t-1}\right), \tag{9}$$

where (x_i^t, y_i^t) is the coordinates of each object in the image plane. The implementation details are as follows. The first step is to extract the speed feature, use LSTM for historical speed modeling, and obtain a 500-dimensional historical speed feature vector, which is cascaded with the speed of the current frame detection frame relative to the target and sent to the discriminator. Get the final speed matching result.

3.3.3. Interaction History Model. In multitarget tracking, the motion model of the target self is often inaccurate, especially in some congested scenes. Often due to some reasons, the trajectory of the target does not conform to some conventional motion models, and often, the interaction model can solve these problems. The interaction model considers the structural information of all targets in the entire scene. As long as most of the target motion information does not change much, the entire motion structure will not change greatly, even if some targets move unpredictably. In view of the above reasons, AMIR proposes an interaction model based on RNN, which assists the judgment of target displacement by defining the environment grid map around the target. The specific definition is as follows:

$$O_i^t(m,n) = \bigvee_{j \in N_i} 1_{mn}\left[x_j^t - x_i^t, y_j^t - y_i^t\right], \tag{10}$$

where $\bigvee_{j \in N_i}$ represents the field of tracking target i and $1_{mn}[x, y]$ is the indicator function. When there is a target at the position (x, y), $1_{mn}[x, y]$ is 1; otherwise, it is 0. Then, $\{O_i^1, \cdots, O_i^t\}$ is sent to the LSTM network for time series analysis, and finally, a 500-dimensional environmental interaction variable is obtained, which is then cascaded with the environmental interaction variable of the detection frame of the current frame and then sent to the binary classifier for discrimination.

4. Experiment and Analysis

4.1. Single-Target Tracking Algorithm Experiment. This chapter adopts OTB100 as the evaluation dataset. OTB100 contains a total of 100 tracking video sequences. The evaluation annotation uses the OPE curve as the evaluation index. As the name implies, the OPE curve is the result of only one tracking; that is, if the tracking fails, the current tracking chain is ended without reinitialization. The OPE curve has two evaluation forms; one is the curve based on the area intersection ratio. When the area of a frame predicted by the algorithm and the area of the labeled frame overlap is greater than a threshold, the frame is considered to be tracked successfully. The percentage of total successful frames over all frames is called the success rate. The other is based on pixel offset evaluations. When the offset between the center point of the frame predicted by the algorithm and the center point of the labeled frame in a frame is less than a threshold, the frame is considered to be tracked successfully. The percentage of total successful frames over all frames is called precision. The two curves complement each other and are indispensable. For example, the success rate can evaluate the size change of the target, but the accuracy rate cannot. The integral of the curve area is the main evaluation index. The specific experimental results are shown in Figures 2–4.

A portion of the standard single-target tracking method based on correlation filter is examined in this section, and it is contrasted with the single-target tracking algorithm based on deep learning in this section. MDNet is the most successful and accurate of these algorithms, and it outperforms all others in terms of success and accuracy. GOTURN is the fastest, but it is also the worst of the algorithms tested. In order to balance MDNet and GOTURN, the SiameseFC network was developed. The effect of the network is significantly improved when compared to GOTURN, but the speed is twice as slow. While not as quick as GOTURN, Siamese is still significantly faster than alternative algorithms in many cases. When compared to other traditional single-target tracking algorithms such as DSST, MEEM, and CSR-DCF, the effect of the single-target tracking algorithm based on deep learning is highly competitive in one aspect, whereas the effect of the single-target tracking algorithm based on deep learning is not.

4.2. Multitarget Tracking Algorithm Experiment

4.2.1. Datasets. A good multitarget tracking dataset is the basis for designing a good algorithm, and it is also one of the prerequisites for horizontal comparison with other algorithms. First of all, a complete multitarget tracking dataset should have the following conditions: (1) tracking from different perspectives, for example, linear motion models tend to predict more accurately in the monitoring perspective, but there will be a high degree of accuracy in videos shot by handheld mobile devices. (2) For target scenes with different densities, if there are few targets, it cannot fully reflect the multitarget tracking algorithm's ability to deal with target interaction occlusion. If there are too many targets, the advantages and disadvantages of some real-time algorithms cannot be properly reflected. (3) Different weather or lighting, in general, shadows or strong lighting can easily cause huge changes in the pixel value of the target appearance, so this is an important part of the robustness of the multitarget tracking algorithm. (4) Different resolutions, often under different resolutions, show different details of objects. For example, at low resolutions, similar objects are easily mismatched. (5) Different video frame rates are often very sensitive to the parameters of the motion model, so it is also an important part of the robustness of the response algorithm.

In order to make a fair comparison with other algorithms, this paper selects MOT2015 and MOT2016 provided by the multitarget tracking challenge as training and test sets to simulate player movement in football games. MOT2015 and MOT2016 contain various complex scenarios, such as common surveillance video and video captured by mobile devices. Due to severe video jitter and pedestrian occlusion, videos collected by mobile devices are often difficult to track objects. MOT2015 is a multitarget tracking dataset containing more than 20 video sequences. The acquisition devices include still cameras and moving cameras. The collected perspectives include upward, head-up, and top-down perspectives, as well as different weather conditions and different lighting conditions. There are more than 11,000 images in the video sequence, of which the training set contains 5,500 images, a total of 500 tracking targets, and nearly 40,000 tracking target boxes. The test set contains 5700 images, more than 700 tracking targets, and about 60,000 tracking target boxes. The detection frame provided by the dataset is obtained by fusing channel features.

4.2.2. Experimental Results and Analysis. In order to demonstrate the effectiveness of the tracking algorithm based on deep learning, STAM and AMIR were tested on the MOT2015 and MOT2016 datasets, respectively. The experimental results demonstrate that the deep learning-based multitarget tracking algorithms STAM and AMIR are compared with other traditional multitarget tracking algorithms. On the main indicator MOTA, the best results or competitive results were achieved. On the MOT2015 dataset, AMIR is 7.7% higher than the metric learning-based MDP algorithm, 4.3% higher than the optical flow-based NOMT algorithm, and 9% higher than the structured association-based SCEA algorithm. STAM is 4.3% higher than the MDP algorithm based on metric learning, similar to the NOMT offline multitarget tracking algorithm based on optical flow, 5.6% higher than the structured association SCEA algorithm, and higher than the CDA_DDALpb based on the traditional appearance model. 1.5%. On the MOT2016 dataset, AMIR is about 9.8% higher than the LTTSC-CRF algorithm based on continuous conditional random fields, 0.9% higher than the optical flow-based NOMT algorithm, and 13.3% higher than the energy optimization-based CEM algorithm. STAM is 8.4% higher than LTTSC-CRF algorithm based on continuous conditional random field and 11.9% higher than CEM algorithm based on energy optimization. Experiments have shown that compared with

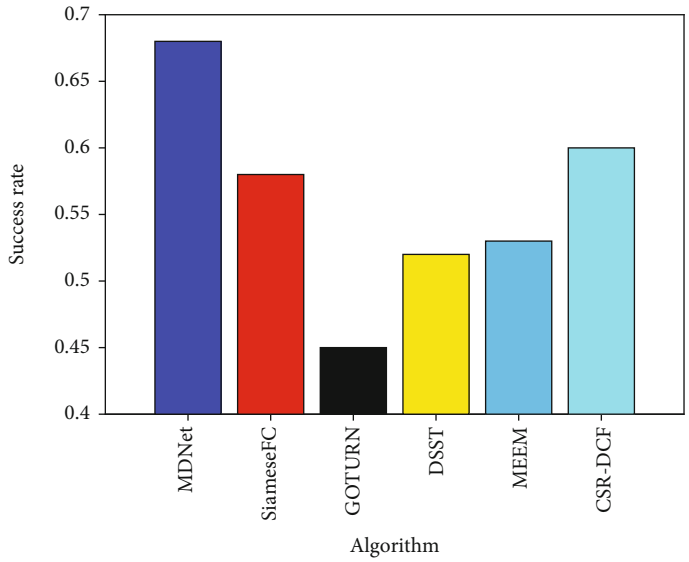

Figure 2: Success rate of single-target tracking algorithm.

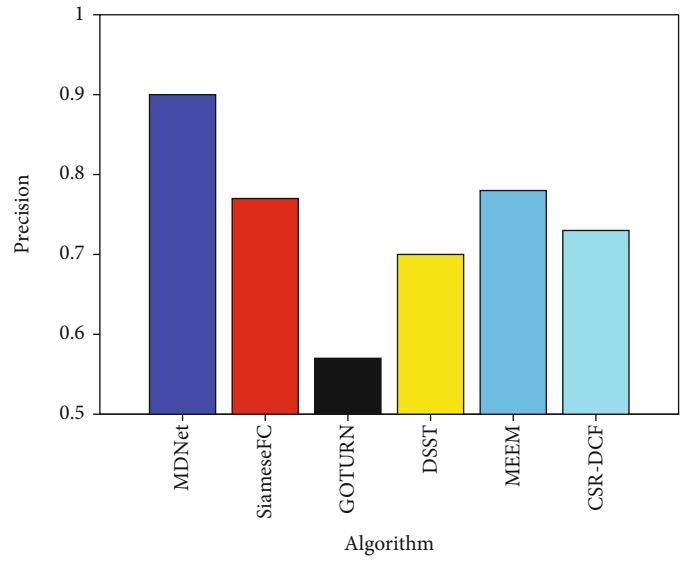

Figure 3: Precision of single-target tracking algorithm.

other traditional multitarget tracking algorithms, deep learning algorithms show great competitiveness. The specific results are shown in Tables 1 and 2.

4.3. Visualization Experiment of Football Tactics. The software design is mainly divided into four layers, the first is the input layer, the four cameras independently collect the stadium images, the second is the single-camera multitarget tracking module, which performs independent multitarget tracking in the images collected by each camera, and then, the multicamera data fusion layer fuses the data obtained by multiple cameras and multitarget tracking, corresponding to each player, and finally is the output layer, which saves the obtained player movement data into the database and can further analyze the data, including the player's movement data. Among them, the single-camera multitarget tracking module and the multicamera fusion module together constitute the multicamera multitarget tracking algorithm. The system structure diagram is shown in Figure 5.

This section selects the clips from the 1/8 finals of the UEFA Champions League in a certain season, named sequence 1 and sequence 2, respectively. There is also a clip from the UEFA Champions League semifinal match in a certain season, named sequence 3. Three snippets are used to test the player detection and tracking algorithm in this paper. The resolution of the video is 856×480, and the format is AVI. The obtained player tracking results are shown in Table 3.

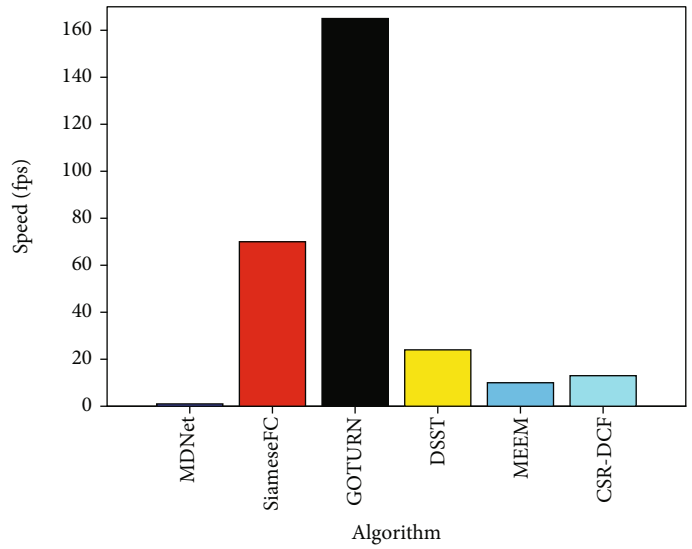

Figure 4: Speed of single-target tracking algorithm.

Table 1: Results of STAM and AMIR on the MOT2015 test set.

Method	AMIR	STAM	NOMT	CDA_DDALpb	MDP	SCEA
MOTA	38.5%	35.1%	34.6%	33.6%	30.8%	29.5%

Table 2: Results of STAM and AMIR on the MOT2016 test set.

Method	AMIR	STAM	NOMT	CDA_DDALpb	LTTSC-CRE	CEM
MOTA	49.2%	47.8%	48.3%	45.7%	39.4%	35.9%

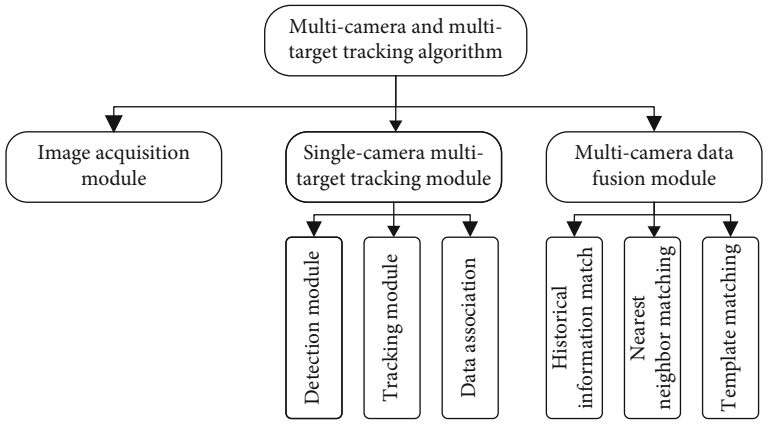

Figure 5: System structure diagram.

Table 3: Player tracking results for sequences 1, 2, and 3.

Items	Sequence 1	Sequence 2	Sequence 3
Total frames	58	36	20
Occlusions by players of different teams	3	1	1
Occlusions by players and referees	2	0	0
Occlusions by players of same teams	0	3	2
Correct matches after separation	2	1	1
Number of players entering the screen	0	1	1
Number of players leaving the screen	1	0	2

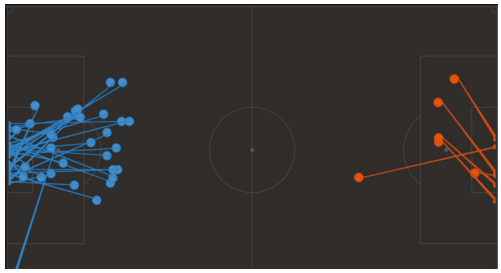

Figure 6: Soccer tactic visualization results.

The player tracking results of the above three video sequences show that the algorithm in this paper has correctly processed the occlusion judgment, separation, and matching and entry and exit of all players in these test sequences, which proves the effectiveness of the algorithm in this paper. According to the results of the above separation, this paper obtains the football visualization tactical results of the game, as shown in Figure 6.

5. Conclusion

This paper mainly studies target detection and tracking in football videos and contributes to the research of football tactical visualization. Multiobject tracking in football videos is a very challenging task. In this paper, a multitarget tracking algorithm based on deep learning and continuous conditional random fields is proposed for the difficulty of multitarget tracking, that is, retracking under occlusion conditions. The innovation of the algorithm is as follows: (1) a CNN for target displacement prediction is proposed, which no longer relies on the previous linear motion model or quadratic motion model, so that the multitarget tracking algorithm can be applied to more scenes. (2) For the first time in a multitarget tracking algorithm, a continuous conditional random field is used to model the asymmetric nature of the target relationship. In this paper, the tracking chain with high confidence is used to correct the tracking chain with high confidence, so that the robustness of the multitarget tracking algorithm is further improved. At the same time, the CNN for target displacement prediction can be cascaded with the continuous conditional random field for end-to-end training, which greatly reduces the training difficulty. The parameters of the experiment in this paper are simple, and good results have been obtained on different datasets.

Data Availability

The datasets used during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest

The author declares that he has no conflict of interest.

References

[1] M. Taj and A. Cavallaro, *Multi-View Multi-Object Detection and Tracking*, Berlin Heidelberg, Computer Vision. Springer, 2010.

[2] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 34, no. 7, pp. 1409–1422, 2012.

[3] X. Zhao and D. Gong, "Tracking using motion patterns for very crowded scenes," in *European Conference on Computer Vision*, pp. 315–328, Springer, Berlin, Heidelberg, 2012.

[4] W. Hu, X. Li, W. Luo, X. Zhang, S. Maybank, and Z. Zhang, "Single and multiple object tracking using log-Euclidean Riemannian subspace and block-division appearance model," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 34, no. 12, pp. 2420–2440, 2016.

[5] L. Zhang and L. V. D. Maaten, "Preserving structure in model-free tracking," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 36, no. 4, pp. 756–769, 2014.

[6] J. Xing, H. Ai, L. Liu, and S. Lao, "Multiple player tracking in sports video: a dual-mode two-way Bayesian inference approach with progressive observation modeling," *IEEE Transactions on Image Processing*, vol. 20, no. 6, pp. 1652–1667, 2011.

[7] G. Duan, H. Ai, S. Cao, and S. Lao, "Group tracking: exploring mutual relations for multiple object tracking," in *European conference on computer vision*, pp. 129–143, Springer, Berlin, Heidelberg, 2012.

[8] K. X. Dai, G. H. Li, D. Tu, and J. Yuan, "Prospects and current studies on background subtraction techniques for moving objects detection from surveillance video," *Journal of Image & Graphics*, vol. 11, no. 7, pp. 919–927, 2016.

[9] N. Liu, *Segmentation and Tracking of Moving Objects in Sports Video*, Baoding: North China Electric Power University, 2006.

[10] V. Pallavi, J. Mukherjee, A. K. Majumdar, and Shamik Sural, "Graph-based multiplayer detection and tracking in broadcast soccer videos," *IEEE Transactions on Multimedia.*, vol. 10, no. 5, pp. 794–805, 2008.

[11] P. J. Figueroa, N. J. Leite, and R. M. L. Barros, "Tracking soccer players aiming their kinematical motion analysis," *Computer Vision and Image Understanding*, vol. 101, no. 2, pp. 122–135, 2006.

[12] W. Xu, Q. Zhao, Y. Wang, and X. Li, "Online learned player recognition model based soccer player tracking and labeling for long-shot scenes," *IEICE Transactions on Information and Systems*, vol. 97, no. 1, pp. 119–129, 2014.

[13] Q. Tran, A. Tran, T. B. Dinh, and D. Duong, "Long-view player detection framework algorithm in broadcast soccer videos," in *International Conference on Intelligent Computing*, pp. 557–564, Springer, Berlin, Heidelberg, 2011.

[14] H. S. Yoon, Y. J. Bae, and Y. Yang, "A soccer image sequence mosaicking and analysis method using line and advertisement board detection," *ETRI Journal*, vol. 24, no. 6, pp. 443–454, 2002.

[15] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," *IEEE Transactions on Image Processing.*, vol. 12, no. 7, pp. 796–807, 2003.

[16] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-speed tracking with kernelized correlation filters," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 37, no. 3, pp. 583–596, 2015.

[17] S. H. Khatoonabadi and M. Rahmati, "Automatic soccer players tracking in goal scenes by camera motion elimination," *Image and Vision Computing*, vol. 27, no. 4, pp. 469–479, 2009.

[18] J. Van De Weijer, C. Schmid, J. Verbeek, and D. Larlus, "Learning color names for real-world applications," *IEEE Transactions on Image Processing*, vol. 18, no. 7, pp. 1512–1523, 2009.

[19] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 37, no. 9, pp. 1904–1916, 2015.

[20] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1820–1833, 2011.

[21] X. Z. Liu, *Video Image Sequence Target Tracking Algorithm and Its Application Research*, Central South University, Changsha, 2011.

[22] G. D. Cui, *Research on Player Detection and Tracking in Soccer Video*, Hebei Polytechnic University, Tianjin, 2009.

[23] F. L. Chang, L. Ma, and Y. Z. Qiao, "Human oriented multi-target tracking algorithm in video sequence," *Control and Decision*, vol. 22, no. 4, pp. 418–422, 2007.

[24] Q. Li, C. F. Shao, H. Yue, and C. G. Liu, "Pedestrian oriented multi-object tracking algorithm in video sequence," *Transactions of Beijing Institute of Technology*, vol. 33, no. 2, pp. 178–184, 2013.

[25] C. G. Liu, D. S. Cheng, J. F. Liu, J. H. Huang, and X. L. Tang, "Interactive particle filter based algorithm for tracking multiple objects in videos," *Acta Electronica Sinica*, vol. 39, no. 2, pp. 260–267, 2011.

[26] G. C. Liu and Y. J. Wang, "An algorithm of multi-target tracking based on improved particle filter," *Control and Decision*, vol. 24, no. 2, pp. 317–320, 2009.

[27] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr, "Fully-convolutional siamese networks for object tracking," in *European conference on computer vision*, no. 1-pp. 1–8, Springer, Cham, 2016.

[28] E. Morais, A. Ferreira, S. A. Cunha, R. M. L. Barros, A. Rocha, and S. Goldenstein, "A multiple camera methodology for automatic localization and tracking of futsal players," *Pattern Recognition Letters*, vol. 39, pp. 21–30, 2014.

[29] S. H. Bae and K. J. Yoon, "Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 3, pp. 595–610, 2018.

[30] T. D'Orazio, M. Leo, P. Spagnolo et al., "An investigation into the feasibility of real-time soccer offside detection from a multiple camera system," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 12, pp. 1804–1818, 2009.

[31] A. Mittal and L. S. Davis, "M2Tracker: a multi-view approach to segmenting and tracking people in a cluttered scene [J]," *International Journal of Computer Vision*, vol. 51, no. 3, pp. 189–203, 2003.

[32] J. Liu, X. Tong, W. Li, T. Wang, Y. Zhang, and H. Wang, "Automatic player detection, labeling and tracking in broadcast soccer video," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 103–113, 2009.

[33] M. Danelljan, G. Bhat, F. Shahbaz Khan, and M. Felsberg, "Eco: efficient convolution operators for tracking," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6931–6939, Honolulu, USA, 2017.

[34] M. Zhang, Q. Wang, J. Xing et al., "Visual tracking via spatially aligned correlation filters network," in *Proceedings of the European conference on computer vision*, pp. 469–485, Munich, Germany, 2018.

[35] Y. Sun, C. Sun, D. Wang, Y. He, and H. Lu, "Roi pooled correlation filters for visual tracking," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5776–5784, Long Beach, CA, USA, 2019.