

Camera-based Basketball Scoring Detection Using Convolutional Neural Network

Xu-Bo Fu¹ Shao-Long Yue² De-Yun Pan¹

¹Department of Public Physical and Art Education, Zhejiang University, Hangzhou 310058, China

²North China Electric Power University, Beijing 102206, China

Abstract: Recently, deep learning methods have been applied in many real scenarios with the development of convolutional neural networks (CNNs). In this paper, we introduce a camera-based basketball scoring detection (BSD) method with CNN based object detection and frame difference-based motion detection. In the proposed BSD method, the videos of the basketball court are taken as inputs. Afterwards, the real-time object detection, i.e., you only look once (YOLO) model, is implemented to locate the position of the basketball ball hoop. Then, the motion detection based on frame difference is utilized to detect whether there is any object motion in the area of the hoop to determine the basketball scoring condition. The proposed BSD method runs in real-time with satisfactory basketball scoring detection accuracy. Our experiments on the collected real scenario basketball court videos show the accuracy of the proposed BSD method. Furthermore, several intelligent basketball analysis systems based on the proposed method have been installed at multiple basketball courts in Beijing, and they provide good performance.

Keywords: Computer vision, convolutional neural network (CCN), frame difference, motion detection, object detection, real-time system.

Citation: X. B. Fu, S. L. Yue, D. Y. Pan. Camera-based basketball scoring detection using convolutional neural network. *International Journal of Automation and Computing*, vol.18, no.2, pp.266–276, 2021. <http://doi.org/10.1007/s11633-020-1259-7>

1 Introduction

Recently, with the development in artificial intelligence, computer vision and data mining, many intelligent methods have been applied to sports^[1–7]. On one hand, some methods are proposed to predict the outcomes of sports games and help people analyze sports at a macroscopic level. Owramipur et al.^[8] proposed to use Bayesian network to predict the results of football games. Miljkovic et al.^[9] formalized the problem of predicting the outcomes of basketball games in the national basketball association (NBA) league as a classification problem and use Naïve Bayes method to solve it. On the other hand, some methods are proposed to analyze the specific behaviors on the sports ground and help people understand sports at a microscopic level. Nakai et al.^[10] used the human pose estimation method, i.e., OpenPose^[11], to predict the shooting probability of basketball free throws. Liu and Carr^[12] proposed to use a random decision forest and a context-conditioned motion model to detect and track the players in sports games.

Research Article

Manuscript received May 20, 2020; accepted September 25, 2020; published online December 23, 2020

Recommended by Associate Editor Nazim Mir_Nasiri

Colored figures are available in the online version at <https://link.springer.com/journal/11633>

© Institute of Automation, Chinese Academy of Sciences and Springer-Verlag GmbH Germany, part of Springer Nature 2020

The object detection methods^[13–17] have recently developed rapidly with the convolutional neural network (CNN)^[18–22]. These methods have been widely applied in many application scenarios, such as smart video surveillance, autonomous driving, etc. In this paper, we introduce an object detection method into the area of basketball. We propose a camera-based basketball scoring detection (BSD) method with a you only look once (YOLO) basketball hoop detector and a frame difference scoring detector.

When we hold basketball games, the scoring tables or scorers are needed to keep records of the points. However, it is impractical to arrange scoring tables or scorers for all the basketball games, especially for the spare-time games. Therefore, the automatic scoring detection and counting is applicable for basketball courts. What's more, it is a social media age. The amateur players want to share their highlights on basketball courts, such as the moment of scoring, with the basketball APPs, such as “QI-UJI” and “REEE Camera”. Detecting the basketball scoring automatically will help the APPs to cut, upload, store and share the highlight videos. Considering the requirements on real-time performance and practicability, we adopt the simple and efficient methods in the proposed BSD model. As demonstrated in Fig. 1, the videos of the basketball court are taken as the input in the proposed BSD model. Afterwards, the real-time object detec-

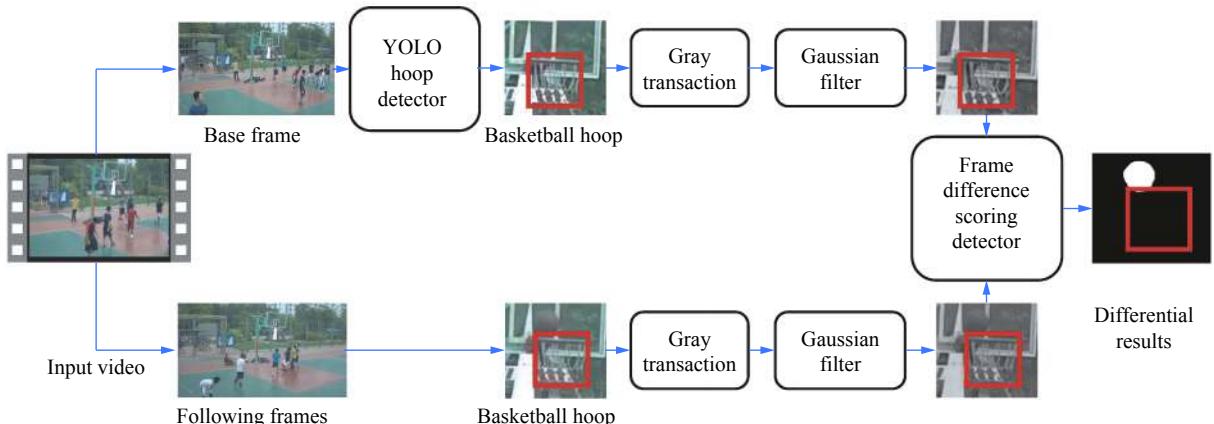


Fig. 1 Framework of the camera-based basketball scoring detection method

tion method, i.e., YOLO^[16, 17], is implemented to locate the position of the basketball hoop in the video. Then, the frame difference method^[23] is utilized to detect whether there is any object motion in the area of the hoop to determine the basketball scoring condition. The implementation of the YOLO method and the frame difference method makes sure the proposed model meets the key requirement, i.e., real-time running, when applied in real scenarios. The experiments and applications in practical basketball court scenarios verify the effectiveness of the proposed BSD method.

2 Related work

In recent years, the CNN based methods have been widely applied in computer vision research^[13–17, 24, 25]. As a fundamental task of computer vision, the object detection methods with CNN have developed rapidly. The regions with CNN features (RCNN)^[13] method is firstly proposed to use the selective search method to generate image regions of interest (RoI). Then, the CNN model is used to extract the visual features of all the generated regions. Finally, the support vector machine (SVM)^[26] classifier is used to determine the object categories in the image regions. Because the RCNN methods need to extract the CNN features of all generated regions, the entire process is very slow and far from being applied in real-time systems. The faster RCNN^[14] method is proposed to accelerate the process and promote the detection accuracy of the RCNN method. The faster RCNN model replaces the selective search with the region proposal network. Meanwhile, the utilization of the RoI pooling layers does not extract CNN features of all the generated image regions. Therefore, the operating efficiency of the faster RCNN method is much better than the RCNN method. However, the faster RCNN method is still far from real-time processing. The processes of RCNN and faster RCNN are two-stage, which contain region proposals and object classification.

In order to further accelerate the process, some one-

stage methods, e.g., the single shot multi-box detector (SSD) method^[15] and the YOLO method^[16, 17], are proposed. In the YOLO method, the category classifications and the object locations can be operated by a single convolutional neural network. The processes of location and classification are unified as one regression task. Compared to the two-stage methods, the detection accuracy of the one-stage methods is relatively inferior. However, the running efficiency of the one-stage methods is much faster, which meets the requirement of the real-time basketball scoring detection task.

Furthermore, Lin et al.^[27] proposed RetinaNet, which is a one-stage detector solving the class imbalance problem in a flexible manner. RetinaNet proposes focal loss to suppress the gradients of easy negative samples. A feature pyramid network is used to detect multi-scale objects at different levels of feature maps. Zhang et al.^[28] introduced RefineDet which is a one-stage detector. RefineDet proposes a cascaded optimization framework to refine the manually defined anchors and improve the anchor quality and final prediction accuracy significantly. Cai and Vasconcelos^[29] proposed cascade RCNN which adopts a similar idea as RefineDet by refining proposals in a cascaded manner. Law and Deng^[30] proposed a novel anchor-free framework CornerNet which detects objects as a pair of corners. CornerNet predicts class heatmaps, pair embeddings and corner offsets on each position of the feature maps to match the objects. Another anchor-free framework is CenterNet^[31] which combines the idea of center-based methods and obtains significant improvements compared to baseline methods. Ghiasi et al.^[32] proposed neural architecture search-feature pyramid network (NAS-FPN) which adopts neural architecture searching to find some new feature pyramid architectures. NAS-FPN consists of both top-down and bottom-up connections to fuse features with a variety of different scales. Similarly, EfficientNet^[33] uses a neural architecture search to design a detection network, which carefully balances network depth, width, and resolution. Zhu et al.^[34] presented a feature selection anchor free (FSAF) frame-

work which can be plugged into one-stage detectors with FPN structure. Fan et al.^[35] proposed a novel few-shot object detection network which aims at detecting objects of unseen categories with only a few annotated training examples. Dong et al.^[36] proposed CentripetalNet which uses centripetal shift to pair corner key points from the same instance. CentripetalNet adopts a cross-star deformable convolution network to conduct feature adaption to make the information more aware at the corners.

The frame difference method^[23, 37–40] has been widely used in different motion detection applications, which is robust and efficient for the scenarios with fixed position cameras. When there are moving objects in the videos, the gray scales of the corresponding pixels between consecutive frames will have differences. Therefore, we can calculate the difference map between the consecutive frames. The pixels of stationary objects are set as 0 in the difference map. The pixels of moving objects have gray scale variations. If the variations get larger than a set threshold, we can consider the object to be moving in the video. The calculation of the frame difference method is quite fast, which meets the real-time running requirement of the proposed BSD method.

3 Methodology

The framework of the proposed camera-based basketball scoring detection method is shown in Fig. 1. In order to guarantee that the BSD method is able to process the basketball video in real-time with satisfactory detection accuracy, the well developed YOLO object detection method and frame difference motion detection method are adopted.

3.1 YOLO hoop detector

When operating the BSD method, the video clips of the basketball court are taken as inputs to the model. Afterwards, the first frame of the video is used as the base frame to determine the position of the basketball hoop. The YOLO network is implemented as the hoop detector.

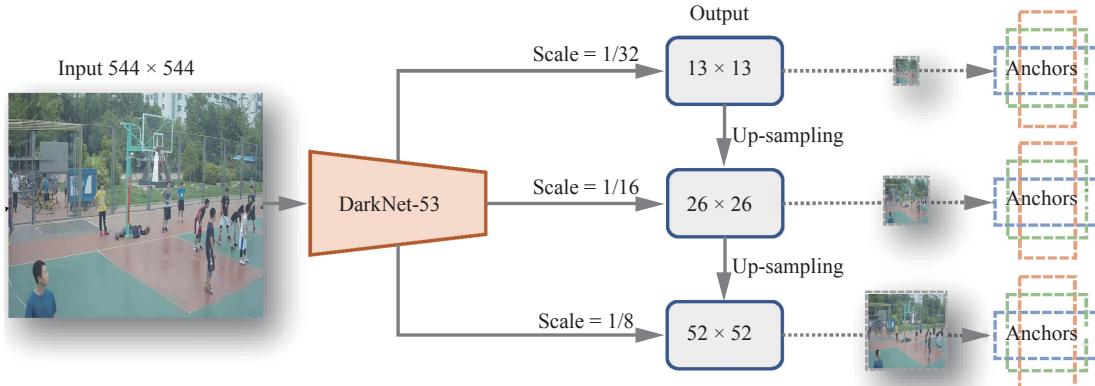


Fig. 2 Framework of the YOLO based basketball hoop detector. For each input image, there are three scales with three default anchors for detection.

The hoop detection results are demonstrated with red boxes in Fig. 1. Because the positions of cameras are fixed on the basketball courts, the hoop areas are stationary in the videos. Therefore, the hoop detector only needs to be operated on the base frame for one time. When training the YOLO hoop detector, the loss function $loss$, contains four parts, i.e., the classification loss $loss_{cls}$, the center coordinate loss $loss_{xy}$, the width-height coordinate loss $loss_{wh}$ and the confidence loss $loss_{conf}$.

The network structure of the adopted YOLO based basketball hoop detector is shown in Fig. 2. For each input image, there are three scales with three default anchors for detection, i.e., 9 anchor size in total. When the base frame of the video inputs the YOLO hoop detector, the image is divided into a $K \times K$ grid. We assume there are M object box boundary candidates in every cell. There is only one object, i.e., the basketball hoop, that we need to detect. Therefore, the binary cross entropy loss is implemented as the classification loss. Formally,

$$loss_{cls} =$$

$$-\sum_{i=0}^{K \times K} I_i^{hoop} \sum_{c \in class} \left[\hat{p}_i(c) \log(p_i(c)) + (1 - \hat{p}_i(c)) \log(1 - p_i(c)) \right] \quad (1)$$

where $I_i^{hoop} = 1$ if the hoop appears in cell i , otherwise $I_i^{hoop} = 0$. The classification candidate $class$ contains only one category. $\hat{p}_i(c)$ denotes the conditional probability for the hoop in cell i .

The center coordinate loss $loss_{xy}$ is implemented to locate the center position of the predicted boundary box of the basketball hoop. Formally,

$$loss_{xy} = \sum_{i=0}^{K \times K} \sum_{j=0}^M I_{ij}^{hoop} [(x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2] \quad (2)$$

where (x_i, y_i) indicates the ground truth center position of the basketball hoop boundary box. (\hat{x}_i, \hat{y}_i) is the predicted results of the center coordinates of the boundary box. $I_{ij}^{hoop} = 1$ if the j -th boundary box in the i -th cell is responsible for detecting the basketball hoop,

i.e., the intersection-over-union (IoU) between the ground truth hoop boundary box and the predicted boundary box is larger than 0.5. Otherwise, $I_{ij}^{hoop} = 0$.

The width-height coordinate loss $loss_{wh}$ is implemented to determine the width and the height coordinates of the hoop boundary box. Formally,

$$loss_{wh} = \sum_{i=0}^{K \times K} \sum_{j=0}^M I_{ij}^{hoop} (2 - w_i \times h_i) \times \left[(w_i - \hat{w}_i)^2 + (h_i - \hat{h}_i)^2 \right] \quad (3)$$

where the indicator I_{ij}^{hoop} works similarly to the one in (2). w_i and h_i indicate the width and the height of the ground truth hoop boundary box. \hat{w}_i and \hat{h}_i are the predicted results. The factor $2 - w_i \times h_i$ in $loss_{wh}$ is set for the hoops in small size. The smaller the hoops are, the larger the value of the loss becomes. It will benefit the small basketball hoops in the images.

Last but not the least, the confidence loss $loss_{conf}$ is implemented to measure the confidence if a basketball hoop is in the predicted boundary box. The confidence loss is also in the form of binary cross entropy loss, which is demonstrated as follows:

$$loss_{conf} = - \sum_{i=0}^{K \times K} \sum_{j=0}^M I_{ij}^{hoop} \left[\frac{\hat{C}_i \log(C_i)}{(1 - \hat{C}_i) \log(1 - C_i)} \right] - \lambda_{nohoop} \sum_{i=0}^{K \times K} \sum_{j=0}^M I_{ij}^{nohoop} \left[\frac{\hat{C}_i \log(C_i)}{(1 - \hat{C}_i) \log(1 - C_i)} \right]. \quad (4)$$

The confidence loss contains two items corresponding to two conditions, i.e., the hoop is in the predicted boundary box and the hoop is not in the predicted boundary box. Similar to (2) and (3), $I_{ij}^{hoop} = 1$ and $I_{ij}^{nohoop} = 0$ if the IoU between the ground truth hoop boundary box and the j -th predicted boundary box in the i -th cell is larger than 0.5. If the IoU between the ground truth hoop boundary box and the predicted boundary box is smaller than 0.5, $I_{ij}^{hoop} = 0$ and $I_{ij}^{nohoop} = 1$. \hat{C}_i denotes the confidence score of the j -th predicted boundary box in the i -th cell. λ_{nohoop} is the hyperparameter.

The entire loss function of the YOLO hoop detector is the sum of the above four loss functions. Formally,

$$loss = \lambda_{coord}(loss_{xy} + loss_{wh}) + loss_{conf} + loss_{cls} \quad (5)$$

where λ_{coord} is the hyperparameter to control the scale of $loss_{xy}$ and $loss_{wh}$.

3.2 Frame difference scoring detector

After locating the position of the basketball hoop, the frame difference scoring detector is operated just on the hoop areas between the base frame and the following frames. The detailed pipeline is shown in Fig. 3. There are two key operations through gating the forward images. Gate 1 is to make the values of each pixel into binary with a proper threshold, while Gate 2 means selecting the holes through ranking the size of all holes. After these steps, the largest hole is extracted for the final scoring prediction. Here, the pixel of the base frame in the video is denoted as $B(x, y)$. The pixel of the current frame is denoted as $F_n(x, y)$. Before the frame difference operation, the video frames are converted to gray images. Then, the Gaussian filter is used for noise reduction. Afterwards, the frame difference operates as follows:

$$D_n(x, y) = |F_n(x, y) - B(x, y)|. \quad (6)$$

Afterwards, a threshold T is set to obtain the binary result of the difference image. Formally,

$$D_{n'}(x, y) = \begin{cases} 1, & \text{if } D_n(x, y) > T \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

The pixels of the moving objects are marked as 1 on the binary image. The pixels of the stationary objects are marked as 0. The connected component analysis operates afterwards to obtain the final frame difference results.

Lastly, the model determines the result of the basketball shot by detecting whether the moving object, i.e., the ball, goes through the red box in Fig. 1, i.e., the basketball hoop.

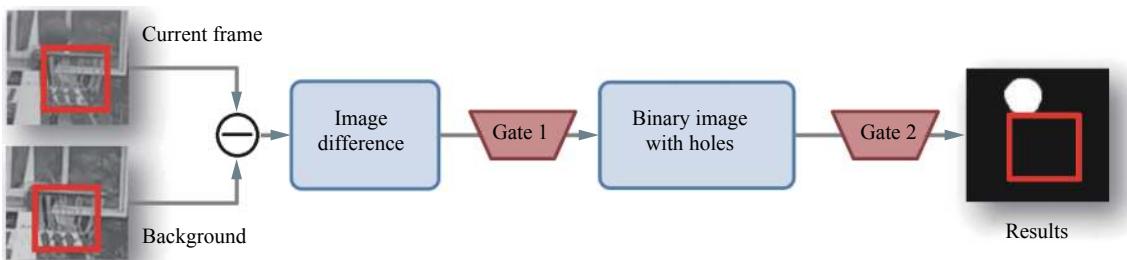


Fig. 3 Pipeline of frame difference scoring detector. Gate 1 is to make the values of each pixel into binary with a proper threshold, while Gate 2 means selecting the holes through ranking the size of all holes. After these steps, the largest hole is extracted for final scoring prediction.

4 Datasets

In order to train the proposed basketball scoring detection model, we have collected a basket and hoop image dataset. The dataset contains the photos of the basketball court, the surveillance images of the basketball court and the screenshots of basketball games. The example images of the dataset are shown in Fig. 4. This dataset contains 4000 images, in which 3500 images are used for training the YOLO hoop detector and 500 images are used for testing. In the dataset, 3638 images contain only one basketball hoop. Meanwhile, 362 images contain at least two basketball hoops. The images are labeled with the data format of PASCAL VOC Challenge^[41]. When testing the entire BSD method, five long basketball videos which are captured in real scenarios are used. The videos are resized to 1280×720 . The five test videos contain 44 basketball scoring moments. The representative frames of the test videos are shown in Fig. 5.



Fig. 4 Data examples of the basket and hoop image dataset



Fig. 5 Representative frames of the test videos

5 Experiments

5.1 Experimental details

During the experiments, the YOLO basketball hoop detector is pre-trained on the Microsoft COCO^[42] dataset. Then the YOLO basketball hoop detector is finetuned on the collected basket and hoop image dataset. The images are resized to 544×544 to input the YOLO basketball hoop detector. During the training process of the hoop detector, the learning rate of the model is set to 1×10^{-4} for the first 20 training epochs. Then the learning rate is set to 1×10^{-6} for the following 30 training epochs. The batch-size of the model in the training process is set to 6. The grid number K in the equations is set to 13, 26 and 52, respectively in the experiments. The non-maximum suppression^[43] is utilized to obtain the final basketball hoop detection results. The number of the object box boundary candidate in every cell, i.e., M , is set as 3 in the experiments. In (5), $\lambda_{coord} = 5$. For the confidence

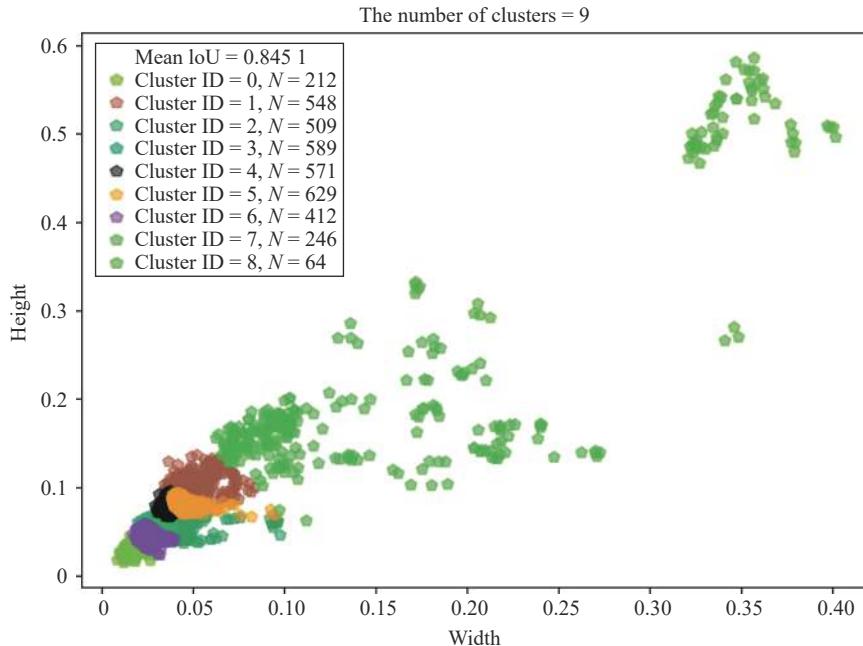


Fig. 6 Anchor size selection for YOLO training. We implement the k -means clustering on all anchors across the training set. After clustering, we rank the 9 cluster centers according to their sizes and set them as the default anchors for training.

loss, $\lambda_{nohoop} = 0.5$.

As the YOLO method has defined the default anchors, therefore, the anchor size selection is crucial. To this end, we implement the k -means clustering on all anchors across the training set. After clustering, there are 9 anchor groups, as shown in Fig. 6. Then, we rank the 9 cluster centers according to their sizes and set them as the default anchors for training.

The experimental platform is based on the Nvidia GTX TitanX GPU and the Intel i7-9750 CPU.

5.2 Experiments of the BSD method

The quantitative experimental results of the YOLO basketball hoop detector on the collected basket and hoop

image dataset are shown in Table 1. The quantitative experimental results of the BSD method on the five long test videos are shown in Table 2.

As shown in Table 1, the YOLO basketball hoop detector can process 30 frames per second (FPS) on the Nvidia GTX TitanX GPU in the experiments. The detector is active only for the first frame of the input video. Therefore, the YOLO detector is fast enough for the real-time processing^[44]. As to the hoop detection accuracy, the average precision at $\text{IoU} = 0.5$ (AP_{50}) is used as the metric. For the AP_{50} metric, if the intersection-over-union (IoU) between the hoop detection result and the ground hoop annotation area is larger than 0.5, we can consider the detection result is correct. The AP_{50} of the YOLO hoop detector on the basket and hoop image dataset is 92.59%. The receiver operating characteristic (ROC)

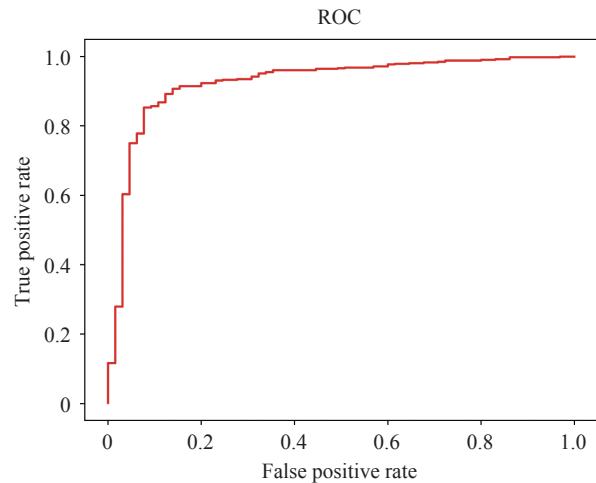


Fig. 7 ROC curve of the YOLO based basketball hoop detector

Table 1 Quantitative experimental results of the YOLO basketball hoop detector

Method	FPS	AP_{50} (%)
YOLO hoop detector	30	92.59

Table 2 Quantitative experimental results of the basketball scoring detection method

Video	Ground truth scoring	Detected scoring	FPS	Accuracy (%)
1	10	9	80	90.00
2	12	11	80	91.67
3	9	8	80	88.89
4	10	8	80	80.00
5	3	3	80	100.00
All	44	39	80	88.64

curve is shown in Fig. 7. It shows the YOLO based basketball hoop detector is robust for practical application.

The quantitative experimental results of the entire basketball scoring detection method are demonstrated in Table 2. The BSD method is able to process 80 frames per second, which meets the demand of real-time processing. The proposed model is tested on the five basketball videos with 44 scoring moments. The scoring detection accuracy on all the videos is 88.64%.

Some unsuccessful basketball hoop detection examples on the collected dataset are shown in Fig. 8. As we can see from the unsuccessful examples, the background of some images is complicated and the view angle of some hoops varies considerably.

The qualitative results of the basketball hoop detection on the five test long videos are shown in Fig. 9. The basketball hoop is marked with yellow boxes by the hoop detector. As is demonstrated, the YOLO basketball hoop detector successfully detects the locations of the basketball hoops for all the real scenario videos.

The sample results of the frame difference scoring detector are shown in Fig. 10. The left images are the ori-

ginal frames of the test videos. The right images are the corresponding frame difference results.

5.3 Ablation study of different input image size

In order to demonstrate the influence of different sizes of input images to the YOLO basketball hoop detector, we change the image size to test the model on the collected basket and hoop image dataset. The quantitative results are shown in Table 3.

As demonstrated in Table 3, the YOLO basketball hoop detector with the input image size of 544×544 achieves the best performance on the basket and hoop image dataset. Therefore, we select 554×544 as the input image size of the basketball hoop detector and the basketball scoring detection system.

5.4 Ablation study of different basketball hoop detection methods

In order to compare the efficiency and effectiveness of



Fig. 8 Unsuccessful examples of basketball hoop detection

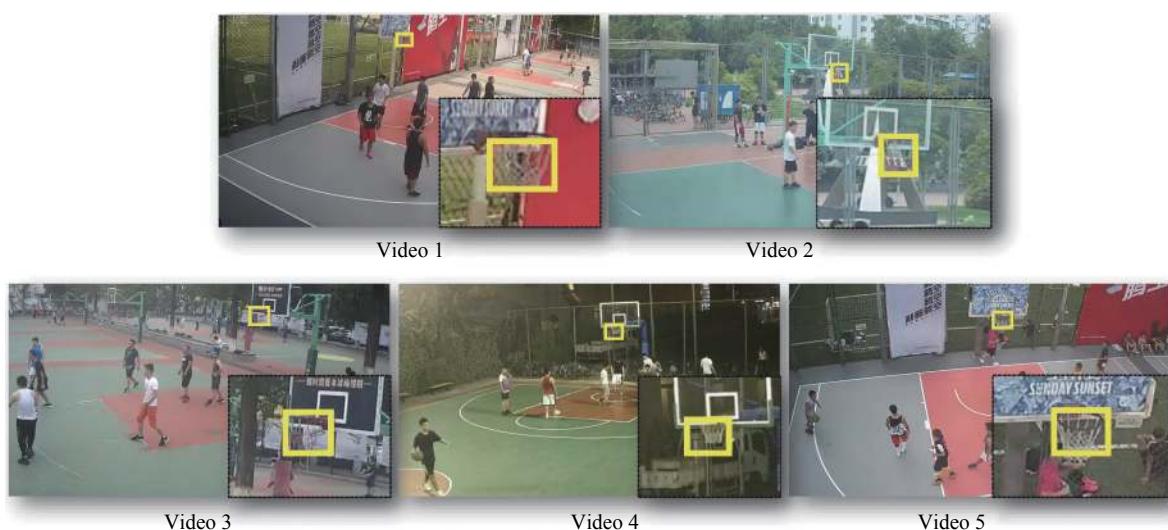


Fig. 9 Results of basketball hoop detection on real scenario videos

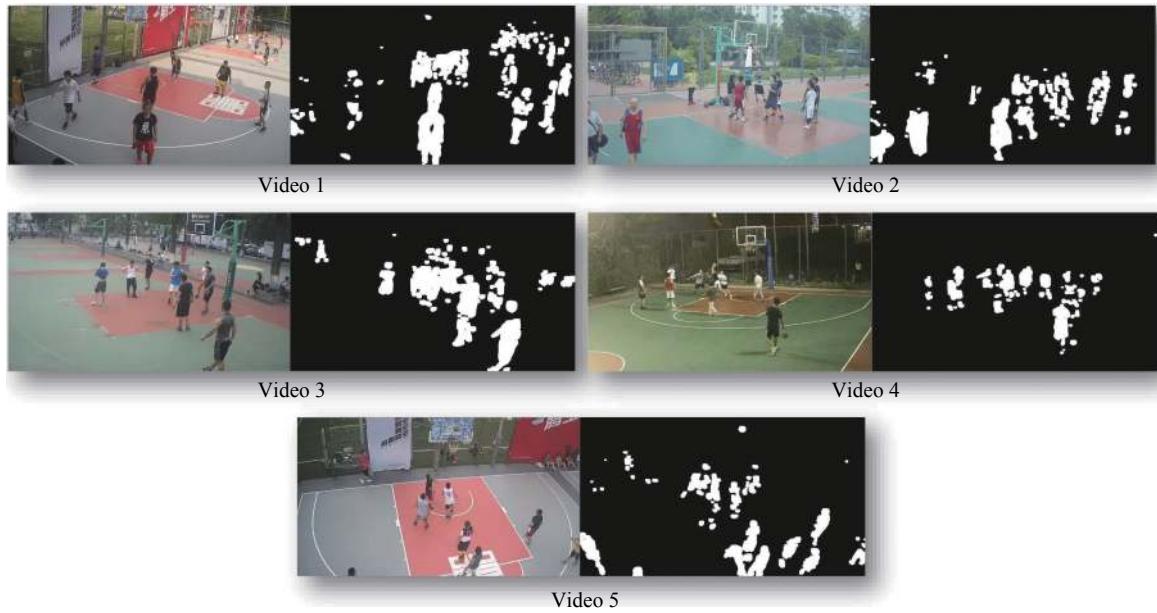


Fig. 10 Sample results of the frame difference scoring detector

Table 3 Quantitative experimental results of the YOLO basketball hoop detector with different input image size

Input image size	AP_{50} (%)
256×256	70.26
288×288	80.75
320×320	84.82
352×352	88.11
384×384	92.04
416×416	91.60
448×448	91.59
480×480	91.26
512×512	92.11
544 × 544	92.59
576×576	92.33
608×608	92.10

Table 4 Ablation study of different basketball hoop detector

Method	Detection method category	FPS	AP_{50} (%)
Faster RCNN detector	Two-stage	0.80	89.26
SSD detector	One-stage	1.25	91.30
YOLO detector	One-stage	30	92.59

the YOLO basketball hoop detector in the proposed model, the ablation study between the YOLO hoop detector and the other detection methods is implemented. The representative two-stage object detection method, i.e., faster RCNN, and another representative one-stage meth-

od, i.e., SSD, are used for the ablation study. The quantitative results are shown in Table 4.

As demonstrated by the ablation study results, the YOLO based detector achieves superior basketball hoop detection accuracy to the faster RCNN based detector and the SSD based detector. Moreover, the YOLO based detector achieves much faster results on running efficiency than the other two detectors, which makes sure that the entire model is a real-time system.

As shown in Table 4, the SSD based basketball hoop detector achieves approximated detection accuracy to the YOLO based detector. But, the running efficiency is much inferior to the YOLO based detector.

5.5 Applications of the BSD system

The proposed method has been installed at multiple basketball courts in Beijing. The examples of the BSD system user interface are shown in Fig. 11. The system is used for automatic scoring detection and acts as an intelligent scorer for the games.

As shown in the bottom line of Fig. 11, the player detection function is added to the proposed BSD system, which is also a preparation for further analysis of the players on court.

6 Conclusions

In this paper, a basketball scoring detection method is proposed. The proposed BSD method contains a YOLO basketball hoop detector and a frame difference scoring detector. The model takes the videos of basketball game as input and detects the scoring condition automatically.



Fig. 11 Interface of the BSD system. The top is the scoring detection demo, while the bottom is the player activity analysis window. The whole BSD system could provide comprehensive understandings from raw videos, which could contribute to the training of the team and help the player perform better.

The BSD method can process the basketball videos in real-time with satisfactory scoring detection accuracy.

In the future, larger scale datasets will be collected to further promote the BSD method. More efficient and effective detection models will be applied in the BSD method. What's more, the BSD method will contain more basketball analysis functions.

Acknowledgements

This work was supported by Research on Educational Science Planning in Zhejiang Province (No. 2019SCG195), "13th Five Year Plan" Teaching Reform Project of Zhejiang University and Shandong Provincial Key Research and Development Program (Major Scientific and Technological Innovation Project) (No. 2019JZZY010119).

References

- [1] G. Thomas, R. Gade, T. B. Moeslund, P. Carr, A. Hilton. Computer vision for sports: Current applications and research topics. *Computer Vision and Image Understanding*, vol. 159, pp. 3–18, 2017. DOI: [10.1016/j.cviu.2017.04.011](https://doi.org/10.1016/j.cviu.2017.04.011).
- [2] T. B. Moeslund, G. Thomas, A. Hilton. *Computer Vision in Sports*, Cham, Germany: Springer, 2014. DOI: [10.1007/978-3-319-09396-3](https://doi.org/10.1007/978-3-319-09396-3).
- [3] R. P. Schumaker, O. K. Solieman, H. Chen. *Sports Data Mining*, Boston, USA: Springer, 2010. DOI: [10.1007/978-1-4419-6730-5](https://doi.org/10.1007/978-1-4419-6730-5).
- [4] M. Haghigiat, H. Rastegari, N. Nourafza. A review of data mining techniques for result prediction in sports. *Advances in Computer Science: An International Journal*, vol. 2, no. 5, pp. 7–12, 2013.
- [5] C. K. Leung and K. W. Joseph. Sports data mining: Pre-
- [6] A. McCabe and J. Trevathan. Artificial intelligence in sports prediction. In *Proceedings of the 5th International Conference on Information Technology: New Generations*, IEEE, Las Vegas, USA, pp. 1194–1197, 2008. DOI: [10.1109/ITNG.2008.203](https://doi.org/10.1109/ITNG.2008.203).
- [7] H. Novatchkov and A. Baca. Artificial intelligence in sports on the example of weight training. *Journal of Sports Science & Medicine*, vol. 12, no. 1, pp. 27–37, 2013.
- [8] F. Owramipur, P. Eskandarian, F. S. Mozneb. Football result prediction with Bayesian network in Spanish League-Barcelona team. *International Journal of Computer Theory and Engineering*, vol. 5, no. 5, pp. 812–815, 2013. DOI: [10.7763/IJCTE.2013.V5.802](https://doi.org/10.7763/IJCTE.2013.V5.802).
- [9] D. Miljkovic, L. Gajic, A. Kovacevic, Z. Konjovic. The use of data mining for basketball matches outcomes prediction. In *Proceedings of the 8th IEEE International Symposium on Intelligent Systems and Informatics*, IEEE, Subotica, Serbia, pp. 309–312, 2010. DOI: [10.1109/SISY.2010.5647440](https://doi.org/10.1109/SISY.2010.5647440).
- [10] M. Nakai, Y. Tsunoda, H. Hayashi, H. Murakoshi. Prediction of basketball free throw shooting by openpose. In *Proceedings of New Frontiers in Artificial Intelligence*, Springer, Yokohama, Japan, pp. 435–446, 2018.
- [11] Z. Cao, G. H. Martinez, T. Simon, S. E. Wei, Y. A. Sheikh. OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019. DOI: [10.1109/TPAMI.2019.2929257](https://doi.org/10.1109/TPAMI.2019.2929257).
- [12] J. C. Liu and P. Carr. Detecting and tracking sports players with random forests and context-conditioned motion models. *Computer Vision in Sports*, T. B. Moeslund, G. Thomas, A. Hilton, Eds., Cham, Germany: Springer, pp. 113–132, 2014. DOI: [10.1007/978-3-319-09396-3_6](https://doi.org/10.1007/978-3-319-09396-3_6).

- [13] R. Girshick, J. Donahue, T. Darrell, J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Columbus, USA, pp. 580–587, 2014. DOI: [10.1109/CVPR.2014.81](https://doi.org/10.1109/CVPR.2014.81).
- [14] S. Q. Ren, K. M. He, R. Girshick, J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *Proceedings of Advances in Neural Information Processing Systems*, Montreal, Canada, pp. 91–99, 2015.
- [15] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, A. C. Berg. SSD: Single shot MultiBox detector. In *Proceedings of the 14th European Conference on Computer Vision*, Springer, Amsterdam, The Netherlands, pp. 21–37, 2016. DOI: [10.1007/978-3-319-46448-0_2](https://doi.org/10.1007/978-3-319-46448-0_2).
- [16] J. Redmon, S. Divvala, R. Girshick, A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Las Vegas, USA, pp. 779–788, 2016. DOI: [10.1109/CVPR.2016.91](https://doi.org/10.1109/CVPR.2016.91).
- [17] J. Redmon and A. Farhadi. YOLOv3: An incremental improvement, [Online], Available: <https://arxiv.org/abs/1804.02767>, 2018.
- [18] A. Krizhevsky, I. Sutskever, G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems*, Lake Tahoe, USA, pp. 1097–1105, 2012.
- [19] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *Proceedings of the 3rd International Conference on Learning Representations*, San Diego, USA, 2014.
- [20] C. Szegedy, W. Liu, Y. Q. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich. Going deeper with convolutions. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Boston, USA, pp. 1–9, 2015. DOI: [10.1109/CVPR.2015.7298594](https://doi.org/10.1109/CVPR.2015.7298594).
- [21] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, Z. Wojna. Rethinking the inception architecture for computer vision. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Las Vegas, USA, pp. 2818–2826, 2016. DOI: [10.1109/CVPR.2016.308](https://doi.org/10.1109/CVPR.2016.308).
- [22] K. M. He, X. Y. Zhang, S. Q. Ren, J. Sun. Deep residual learning for image recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Las Vegas, USA, pp. 770–778, 2016. DOI: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90).
- [23] N. Singla. Motion detection based on frame difference method. *International Journal of Information & Computation Technology*, vol. 4, no. 15, pp. 1559–1565, 2014.
- [24] K. Aukkapinyo, S. Sawangwong, P. Pooyoi, W. Kusakunniran. Localization and classification of rice-grain images using region proposals-based convolutional neural network. *International Journal of Automation and Computing*, vol. 17, no. 2, pp. 233–246, 2020. DOI: [10.1007/s11633-019-1207-6](https://doi.org/10.1007/s11633-019-1207-6).
- [25] A. X. Li, K. X. Zhang, L. W. Wang. Zero-shot fine-grained classification by deep feature learning with semantics. *International Journal of Automation and Computing*, vol. 16, no. 5, pp. 563–574, 2020. DOI: [10.1007/s11633-019-1177-8](https://doi.org/10.1007/s11633-019-1177-8).
- [26] C. Cortes, V. Vapnik. Support-vector networks. *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995. DOI: [10.1007/BF00994018](https://doi.org/10.1007/BF00994018).
- [27] T. Y. Lin, P. Goyal, R. Girshick, K. M. He, P. Dollar. Focal loss for dense object detection. In *Proceedings of IEEE International Conference on Computer Vision*, IEEE, Venice, Italy, pp. 2999–3007, 2017. DOI: [10.1109/ICCV.2017.324](https://doi.org/10.1109/ICCV.2017.324).
- [28] S. F. Zhang, L. Y. Wen, X. Bian, Z. Lei, S. Z. Li. Single-shot refinement neural network for object detection. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 4203–4212, 2018. DOI: [10.1109/CVPR.2018.00442](https://doi.org/10.1109/CVPR.2018.00442).
- [29] Z. W. Cai and N. Vasconcelos. Cascade R-CNN: Delving into high quality object detection. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Salt Lake City, USA, pp. 6154–6162, 2018. DOI: [10.1109/CVPR.2018.00644](https://doi.org/10.1109/CVPR.2018.00644).
- [30] H. Law and J. Deng. CornerNet: Detecting objects as paired keypoints. In *Proceedings of the 15th European Conference on Computer Vision*, Springer, Munich, Germany, pp. 765–781, 2018. DOI: [10.1007/978-3-030-01264-9_45](https://doi.org/10.1007/978-3-030-01264-9_45).
- [31] K. W. Duan, S. Bai, L. X. Xie, H. G. Qi, Q. M. Huang, Q. Tian. CenterNet: Keypoint triplets for object detection. In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE, Seoul, South Korea, pp. 6568–6577, 2019. DOI: [10.1109/ICCV.2019.00667](https://doi.org/10.1109/ICCV.2019.00667).
- [32] G. Ghiasi, T. Y. Lin, Q. V. Le. NAS-FPN: Learning scalable feature pyramid architecture for object detection. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Long Beach, USA, pp. 7029–7038, 2019. DOI: [10.1109/CVPR.2019.00720](https://doi.org/10.1109/CVPR.2019.00720).
- [33] M. X. Tan and Q. V. Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, ICML, Long Beach, USA, 2019.
- [34] C. C. Zhu, Y. H. He, M. Savvides. Feature selective anchor-free module for single-shot object detection. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Long Beach, USA, pp. 840–849, 2019. DOI: [10.1109/CVPR.2019.00093](https://doi.org/10.1109/CVPR.2019.00093).
- [35] Q. Fan, W. Zhuo, C. K. Tang, Y. W. Tai. Few-shot object detection with attention-RPN and multi-relation detector. In *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE, Seattle, USA, 2019. DOI: [10.1109/CVPR42600.2020.00407](https://doi.org/10.1109/CVPR42600.2020.00407).
- [36] Z. W. Dong, G. X. Li, Y. Liao, F. Wang, P. J. Ren, C. Qi. CentripetalNet: Pursuing high-quality keypoint pairs for object detection. [Online], Available: <https://arxiv.org/abs/2003.09119>, 2020.
- [37] C. H. Zhan, X. H. Duan, S. Y. Xu, Z. Song, M. Luo. An improved moving object detection algorithm based on frame difference and edge detection. In *Proceedings of the 4th International Conference on Image and Graphics*, IEEE, Sichuan, China, pp. 519–523, 2007. DOI: [10.1109/ICIG.2007.153](https://doi.org/10.1109/ICIG.2007.153).
- [38] D. A. Migliore, M. Matteucci, M. Naccari. A revaluation of frame difference in fast and robust motion detection. In *Proceedings of the 4th ACM International Workshop on Video Surveillance and Sensor Networks*, Santa Barbara, USA, pp. 215–218, 2006. DOI: [10.1145/1178782.1178815](https://doi.org/10.1145/1178782.1178815).
- [39] Y. Li, Z. X. Sun, B. Yuan, Y. Zhang. An improved method for motion detection by frame difference and background subtraction. *Journal of Image and Graphics*, vol. 14, no. 6,

- pp. 1162–1168, 2009. (in Chinese)
- [40] X. F. Ji, Q. Q. Wu, Z. J. Ju, Y. Y. Wang. Study of human action recognition based on improved spatio-temporal features. *International Journal of Automation and Computing*, vol. 11, no. 5, pp. 500–509, 2014. DOI: [10.1007/s11633-014-0831-4](https://doi.org/10.1007/s11633-014-0831-4).
- [41] M. Everingham, S. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, A. Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, vol. 111, no. 1, pp. 98–136, 2015. DOI: [10.1007/s11263-014-0733-5](https://doi.org/10.1007/s11263-014-0733-5).
- [42] T. Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, C. L. Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the 13th European Conference on Computer Vision*, Springer, Zurich, Switzerland, pp. 740–755, 2014. DOI: [10.1007/978-3-319-10602-1_48](https://doi.org/10.1007/978-3-319-10602-1_48).
- [43] A. Neubeck and L. van Gool. Efficient non-maximum suppression. In *Proceedings of the 18th International Conference on Pattern Recognition*, IEEE, Hong Kong, China, pp. 850–855, 2006. DOI: [10.1109/ICPR.2006.479](https://doi.org/10.1109/ICPR.2006.479).
- [44] Q. Y. Gu and I. Ishii. Review of some advances and applications in real-time high-speed vision: Our views and experiences. *International Journal of Automation and Computing*, vol. 13, no. 4, pp. 305–318, 2016. DOI: [10.1007/s11633-016-1024-0](https://doi.org/10.1007/s11633-016-1024-0).



Xu-Bo Fu received the B.Sc. degree in physical education and training from the Physical Education Department, Zhejiang University, China in 2006, the M.Sc. degree in physical education and training from the Physical Education Department, Zhejiang University, China in 2008. He is an associate professor in the Public Sports and Art Department, Zhejiang University, China. He offers junior and football intermediate courses for undergraduate students. He also published five papers in related fields. He hosted or participated in 4 types of projects. Among them, he presided over the “Regional Sports Economic Industry Layout and Institutional Adjustment Research: Taking the

Yangtze River Delta as an Example”. He participated in the “Practical Research on Public Physical Education Model with Non-Austrian Projects” project and won the second prize of Zhejiang Teaching Achievements. In 2010, he got the “Sunshine Sports Award” by the Sports Association of the Higher Industrial School directly under the Ministry of Education. In 2010, he was awarded as the “Outstanding Physical Education Teacher” by the Hangzhou Sports Association. In 2013, he won the second prize of Zhejiang University Quality Teaching Award.

His research interest is school physical education.

E-mail: fycc@zju.edu.cn (Corresponding author)

ORCID iD: 0000-0001-7973-299X



Shao-Long Yue received the B.Sc. degree in automation from School of Electrical and Electronic Engineering, Shandong University of Technology, China in 2018. He is currently a master student in control engineering at School of Control and Computer Engineering, North China Electric Power University, China.

His research interests include pattern recognition, computer vision and machine learning.

E-mail: slyue@ncepu.edu.cn



De-Yun Pan received the B.Sc. degree in physical education basketball from Beijing Sport University, China in 1999. He also is a national basketball referee. He is currently the director of the Sports Training Center of the Public Sports and Art Department of Zhejiang University, China. He is the deputy director of the Basketball Committee of the Chinese University Sports Association, and the director of the Coaching Committee of the Zhejiang University Sports Association. He is the director of the Coaching Committee of the Zhejiang University Sports Association, the deputy director of the Zhejiang Basketball Association Coaching Committee and member of the Youth Committee of the Chinese Basketball Association.

His research interest is physical education.

E-mail: zdpy@126.com

Citation: X. B. Fu, S. L. Yue, D. Y. Pan. Camera-based basketball scoring detection using convolutional neural network. *International Journal of Automation and Computing*, vol.18, no.2, pp.266–276, 2021. <https://doi.org/10.1007/s11633-020-1259-7>

Articles may interest you

Localization and classification of rice-grain images using region proposals-based convolutional neural network. *International Journal of Automation and Computing*, vol.17, no.2, pp.233-246, 2020.

DOI: [10.1007/s11633-019-1207-6](https://doi.org/10.1007/s11633-019-1207-6)

Hybrid dynamic neural network and pid control of pneumatic artificial muscle using the pso algorithm. *International Journal of Automation and Computing*, vol.17, no.3, pp.428-438, 2020.

DOI: [10.1007/s11633-019-1196-5](https://doi.org/10.1007/s11633-019-1196-5)

Text-mining-based fake news detection using ensemble methods. *International Journal of Automation and Computing*, vol.17, no.2, pp.210-221, 2020.

DOI: [10.1007/s11633-019-1216-5](https://doi.org/10.1007/s11633-019-1216-5)

A practical approach to representation of real-time building control applications in simulation. *International Journal of Automation and Computing*, vol.17, no.3, pp.464-478, 2020.

DOI: [10.1007/s11633-018-1131-1](https://doi.org/10.1007/s11633-018-1131-1)

h state estimation for stochastic markovian jumping neural network with time-varying delay and leakage delay. *International Journal of Automation and Computing*, vol.16, no.3, pp.329-340, 2019.

DOI: [10.1007/s11633-016-0955-9](https://doi.org/10.1007/s11633-016-0955-9)

Pid neural network decoupling control based on hybrid particle swarm optimization and differential evolution. *International Journal of Automation and Computing*, vol.17, no.6, pp.867-872, 2020.

DOI: [10.1007/s11633-015-0917-7](https://doi.org/10.1007/s11633-015-0917-7)

Energy guarantee scheme for real-time systems with energy harvesting constraints. *International Journal of Automation and Computing*, vol.16, no.3, pp.354-368, 2019.

DOI: [10.1007/s11633-018-1166-3](https://doi.org/10.1007/s11633-018-1166-3)



WeChat: IJAC



Twitter: IJAC_Journal