International Journal of Distributed Systems and Technologies
Volume 14 • Issue 2

An Effective Volleyball Trajectory Estimation and Analysis Method With Embedded Graph Convolution

Guanghui Huang, Zhengzhou Tourism College, China*

ABSTRACT

Volleyball trajectory prediction and analysis based on deep learning has become a hot topic in sports video research. However, due to a large amount of calculation in video processing and the fast speed of volleyball movement with the target scale changing rapidly, these challenges lead to low performance. To this end, this paper proposes an effectively variant YOLOv4 framework to predict and analyze the volleyball trajectory based on video sequences. In the proposed framework, the authors adopt the pre-trained YOLOv4 to select some proposal regions with a high confidence score. Then, the authors embed graph convolution to effectively aggregate deep features. Moreover, to improve the detection and localization capacity of small targets, they introduce a new loss function by modeling the target area with Gaussian distribution. The experimental results show that the proposed framework can effectively prompt the performance of volleyball detection.

KEYWORDS

Gaussian Modeling, Small Targets, Volleyball Trajectory Analysis, YOLOv4

1. INTRODUCTION

Multimedia technology has been widely used in intelligent sports. Computer vision and video analysis systems have higher accuracy and real-time than human eyes, which can quickly capture moving objects, and record various motion data of the objects (Li et.al, 2021; Xiao et.al, 2020). As a new intelligent analysis technology, it can automatically analyze image sequences and judge video content without human intervention to achieve fully automatic target detection, tracking, recognition, judgment, recording and emergency disposal (Liu et.al,2020; Jiao et.al, 2020; Gao et al. 2022) instead of traditional feature selection (Zheng et al. 2018; Zheng et al. 2021) or feature extraction (Zhu et al. 2022) technologies.

To effectively realize volleyball trajectory estimation and analysis, Yamato et. Al (1992) adopted the motion, color, texture and other features of two-dimensional small-area blocks to identify different kinds of balls (Yamato et.al, 1992). Lipton (1998) utilized spatial subtraction to detect and track moving

DOI: 10.4018/IJDST.317936 *Corresponding Author

objects in a real video stream (Lipton,1998). In order to further improve the detection efficiency, Rowley et.al (2006) et al. used the information from the optical flow field of moving objects. Each pixel was represented by optical flow. The flow vector formed a block with consistent motion, and the feature was represented by a multivariable Gaussian mixture (Rowley et.al, 2006). Although these methods have made good achievements, there are some deficiencies in the accuracy of recognition.

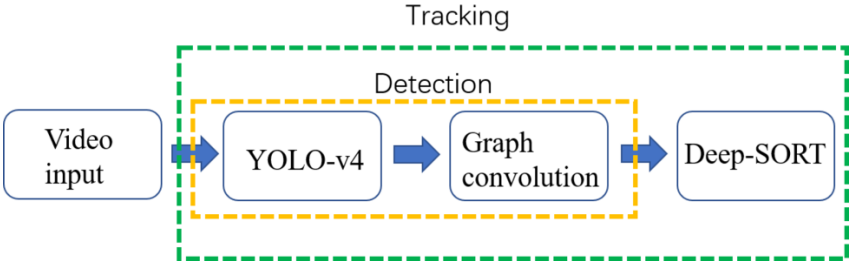
Recently, deep learning-based object detection methods have achieved great success in volleyball trajectory estimation and analysis (Zhao et.al, 2019; Wu et.al, 2020). Due to fast movement, the volleyball size changes rapidly which leads to low detection accuracy. So, one of the major challenges in volleyball detection is small object detection. To alleviate the effect of small object detection, some works exploit multi-scale features. Tan et.al (2020) designed a weighted bi-directional feature pyramid network to fuse multi-scale features. Moreover, by exploiting a compound scaling method, some efficient object detectors are proposed (Tan et.al, 2020). Zhao et.al (2019) designed a multi-level feature pyramid network for scale variation issues. By exploiting feature fusion modules and thinned U-shape modules, this model can effectively detect different scale objects (Zhao et.al, 2019).

Although the use of multi-scale feature information can improve the accuracy of recognition, it will lead to more computational complexity of the whole model. To this end, Tian et.al (2019) used an anchor box-free mechanism to construct a one-stage object detection deep network. Reducing the number of predefined anchor boxes can effectively avoid the computation. Moreover, small targets only occupy a small area of the image. So selecting sparsely multi-scale features from just a few regions is becoming a hotspot to reduce the computational complexity in small object detection.

As the small target, it can provide less useful feature information and cause further loss of information during processing. However, using the contextual information of the objects can improve the accuracy of small object detection. Simonyan et al. (2014) extended the receptive field by using multiple small convolution kernels instead of large convolution kernels. This kind of mechanism can increase the depth and achieve higher detection accuracy (Simonyan et.al, 2014). Law et al (2018) proposed a novel ordinary convolution operation to expand the size of the convolution kernel and make better full use of the contextual information of the samples (Law et al, 2018). Zhao et.al (2021) leveraged multi-scale features by a graph feature pyramid network which can realize the interaction between features of different scales. Graph convolution can use fewer model parameters to extract feature contextual information from graph structure data (Zhao et.al, 2021). However, these models expand the perceptual field of the convolution kernel to obtain contextual information will lead to too much redundant information. Moreover, the graph convolution network constructs the graph data based on the whole image. While the background usually occupies most regions, we try to introduce a novel attention mechanism to make the network pay more attention to important regions for volleyball detection.

Inspired by the above research, we design a novel graph-embedded YOLO framework for different scale volleyball detection which is shown in Fig 1. Firstly, we adopt a pre-trained YOLOv4 framework to select regions of interest (ROIs) based sparse sampling mechanism. Secondly, we construct object

Figure 1. The overview of object detection and tracking



graph data by exploiting ROIs. By embedding lightweight graph convolution layers, we can exploit contextual information for enhancing small-scale object detection. The experimental results show that the proposed model shows better performance in the volleyball detection task. Our contributions are as follows:

(1) We design a novel volleyball detection framework by exploiting graph convolution and pre-trained YOLOv4. This framework achieves higher performance in different scale volleyball detection.
(2) We construct object graph data from a pre-trained YOLOv4 framework by selecting ROIs based on a sparse sampling mechanism. This mechanism can effectively locate object-associated regions and reduce computation costs.
(3) By embedding graph convolution, we can effectively use better use of contextual information which is crucial for small object detection. The results show that graph convolution operation can effectively improve detection accuracy.

The rest of the paper is organized as follows. Section 2 reviews the related works of object detection. In Section 3, the main method of jointing graph convolution and the YOLOv4 deep network are described in detail. The experiments and results are presented in Section 4. Section 5 gives the conclusions.

2. RELATED WORKS

2.1 Small Object Detection

Existing deep-learning based object detection methods are divided into two categories: two-stage and one-stage detectors. The former firstly generated many regional proposals. These proposals are fed into the deep networks for feature extracting, classification and location. Some classic models, i.e., R-CNN, SPPNet, Fast R-CNN, Faster R-CNN, FPN, and R-FCN, have obtained higher detection accuracy with a lower speed (Zhang et.al, 2021; Dai et.al, 2021).

The latter directly predicts the category probability and position coordinate value of objects. Some representative models are SSD, and a variant of YOLO shows higher speed compared to two-stage detectors. However, small object detection is more challenging in many computer vision tasks (Li et.al, 2017). To this end, some works exploit different mechanisms from the following aspects: expanding resolution (Cai et.al, 2016; Kong et.al, 2016), data augmentation (Kisantal et.al, 2019; Liu et.al, 2016), scale-aware training strategy (Li et.al, 2019; Lin et.al, 2017) and integration context information (Chen et.al, 2016; Chen et.al, 2017). In this paper, we leverage graph convolution to mine the context information among object-associated regions.

2.2 Sparse Sample Mechanism

As small objects only occupy a small area of the picture, only a small portion of features play an important role in small object detection. If the features of the whole picture are used for small object detection, it will increase the computational complexity. Moreover, the detection accuracy will be easily interfered with by background noise. Recently, several works designed a sparse sampling mechanism on feature maps to reduce computation. Najibi et.al (2019) proposed the AutoFocus algorithm which was exploited to crop object-related regions at coarse scales (Najibi et.al, 2019). Yang et.al (2022) designed a novel sparse sampling mechanism in one-stage detectors named QueryDet which utilizes a multi-scale feature from the coarse predictions (Yang et.al, 2022). Roh et.al (2021) proposed a sparse DETR network that can adaptively select informative tokens for small object detection (Roh et.al, 2021). In this paper, we joint the sparse sampling mechanism with pre-trained YOLO to reduce computation.

3. GRAPH CONVOLUTION NETWORK

Graph convolution network is a useful tool for feature extraction of graph-structured data in recent years (Li et.al, 2019). Graph convolution operation can effectively mine the dependencies among nodes in graph data (Veličković et.al, 2017). There are two kinds of graph convolution operation: spectral-based and spatial-based (Gong et.al, 2019). With the help of the Fourier formula, the transformation formulas of the Fourier transform on the graph (from the spatial domain to the spectral domain) and inverse Fourier transform on the graph (from the spectral domain to the spatial domain) is defined in spectral-based graph convolution. The application of spatial-based graph convolution is similar to that of convolution in deep learning. Its core is to combine the information of neighbor nodes, for example, the simplest nonparametric convolution method can add the hidden states of all directly connected neighbor nodes to update the hidden state of the current node (Ying et.al, 2018). In this paper, we construct object graph data based on some ROIs, and then the graph convolution layers are embedded for feature extraction.

4. THE PROPOSED METHOD

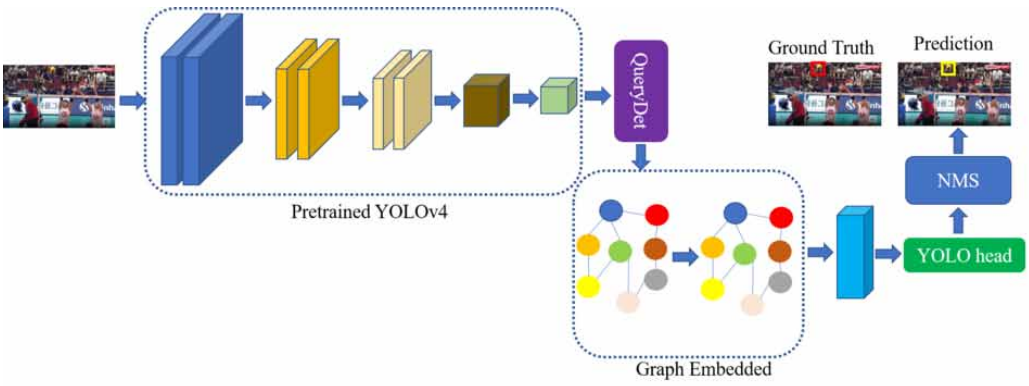
In this section, we show the implementation details of the proposed GE-YOLO framework which is shown in Fig 2 for volleyball detection. We first make an overview of pre-trained YOLO architecture with a sparse sampling mechanism. Then, we construct object graph data based on the result of the sparse sampling mechanism. By exploiting graph-embedded layers, the context information can be effectively mined for different scales of volleyball detection. To effectively realize the volleyball trajectory estimation and analysis, we introduce the Deep-sort algorithm with the GE-YOLO framework.

(1) Pretrained YOLOv4 with sparse sampling

YOLOv4 adds many practical skills based on traditional YOLO to achieve the best balance between detection speed and accuracy. YOLOv4's uniqueness lies in 1. It is an efficient and powerful object detection network. It enables each of us to use the GPU to train a fast and accurate object detector. This is a piece of good news for us who can't afford high-performance video cards! 2. Using a large number of advanced techniques on object detection performance, which is very conscientious! 3. it is more effective and more suitable for training on a single GPU; These improvements include some methods, i.e., CBN, PAN, SAM, etc.

The YOLOv4 framework consists of three parts: backbone, neck and head. The backbone of YOLOv4 is mainly composed of CSPParknet53, which is mainly composed of five residual layers

Figure 2. The architecture of GE-YOLO for volleyball detection



named resblock_body which has a special convolution operation to reduce the resolution and extract the feature information of image data. The neck is mainly composed of SPP and PANet. The main function of SPP is to increase the effectiveness of the receptive field. The main function of PANet is to convert the extracted feature information into coordinates, categories and other information. It is mainly composed of up-sampling and down-sampling. The head continues to follow the Yolov3 detection head. The main function of the three detection heads is to calculate the loss function (the loss function is mainly composed of three parts: (1) positioning loss (2) confidence loss (3) classification loss). After getting the output and comparing it with the real data label, and then reshape the data format as required, and activate the original grid coordinates accordingly.

To effectively detect volleyball, we adopt the pre-trained YOLOv4 on the COCO dataset. The COCO dataset has the "sports ball" classification task, which includes the volleyball category. We adopt the COCO to pre-trained the YOLOv4 for volleyball detection. When the volleyball image is fed into the pre-trained YOLOv4, we adopt a sparse sampling mechanism named AutoFocus to select object-associated features. In this paper, the output of the last convolution layer CSPDarknet53 is used to locate ROIs.

(2) Object graph data construction and embedded graph convolution layers

By introducing the sparse sampling mechanism, we can reduce the computation based on the object-associated ROIs. As the distribution of these ROIs in the picture is irregular. How to exploit these ROIs for different scales of volleyball detection is more challenging. The graph convolution network has a natural advantage that extracts features from unstructured data. In this paper, we leverage graph convolution networks for different feature extracting. Firstly, we construct object graph data based on these ROIs. Formally, the object graph which includes vertices and edges set, i.e., V and E set. The object graph can be defined as $g\left(V,E,A\right)$, where $v_i \in V$ denotes a node, and $e_{ij} = (v_i, v_j) \in E$ denotes an edge. A denotes the adjacency matrix of the object graph, which is defined as follows:

$$A_{ij} = \begin{cases} 1, & if \quad (v_i, v_j) \in E \\ 0, & otherwise \end{cases} \tag{1}$$

To effectively extract features from object graph data, we introduce spectral convolution operation which transforms data from the spatial domain to the spectral domain for processing according to spectrum theory and convolution theorem. Based on spectrum theory, we need to define the graph Laplacian matrix which can be calculated as follows:

$$\hat{L} = D - A \tag{2}$$

where $D_{ii} = \sum_j A_{ii}$.

Furthermore, the normalized graph Laplacian matrix L is defined as:

$$L = I_N - D^{-\frac{1}{2}} A D^{-\frac{1}{2}} \tag{3}$$

where I_N is an identity matrix.

Given an N-dimension graph signal, the convolution operation to the graph domain can be defined as follows:

$$\hat{x} = F(x) = U^T x \tag{4}$$

Where U is calculated from matrix decomposition $L = U\Lambda U^T$.

The spectral graph convolution is calculated as follows:

$$\left(x * h\right) = F^{-1}\left(F(x) \odot F(h)\right) = U\left((U^T h) \odot (U^T x)\right) \tag{5}$$

where h is the graph filter and $\odot$ is the Hadamard product.

To improve the efficiency of calculation, the parameterized diagonalization graph filtered $g_\theta \in R^{N\times N}$ is defined as:

$$g_\theta = diag\left(\theta\right) \tag{6}$$

where $\theta = \left(\theta_1, \theta_2, \ldots, \theta_N\right)$.

The spectral graph convolution operation is simplified as follows:

$$g_\theta * x = Ug\left(\Lambda\right)U^T x \tag{7}$$

where Λ is the parametric diagonal matrix. By introducing the activation function σ, the spectral graph convolution layer can be calculated as follows:

$$Y_o = \sigma\left(Ug\left(\Lambda\right)U^T X\right) \tag{8}$$

Where Y_o and X are the output and input of graph convolution layers, respectively. In this paper, we adopt 1-order ChebConv graph convolutional operation to further reduce computation. The output of the ChebConv graph convolutional layer is calculated as follows:

$$Y_o = \tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}X\Theta \tag{9}$$

Where Θ is the parameter matrix. $\tilde{A} = I_N + A$, and $\tilde{D} = \sum_j \tilde{A}_{ij}$.

According to the conclusion (Kipf et.al, 2016; Wang et.al, 2021), the number of embedded graph convolution layers that can obtain the best performance is two or three. In this paper, we embedded two ChebConv graph convolutional layers to extract features from object graph data.

(3) Volleyball trajectory estimation and analysis

The SORT (simple online and real-time tracking) algorithm has shown better performance in object tracking. The idea of the SORT is very simple. Firstly, the object is located in each frame through a

detector, such as fast-R-CNN, or YOLO. Then, the object's position is predicted in the adjacency frame through a Kalman filter. By calculating the IOU between the predicted position and the actual detected position in the next frame, the similarity of objects in two adjacent frames can be obtained. Finally, the Hungary matching algorithm is exploited to achieve the corresponding ID of adjacent frames.

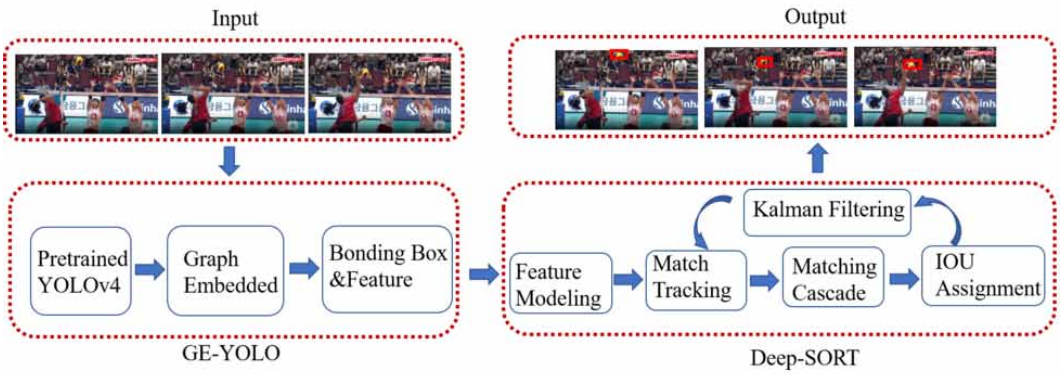
The SORT algorithm uses a simple Kalman filter to deal with the correlation of frame-by-frame data and the Hungary algorithm to measure the correlation. This simple algorithm achieves good performance at a high frame rate. However, the SORT ignores the surface features of the detected object, it will be accurate only when the uncertainty of object state estimation is low. Because only the overlapping area of the box is calculated here, if two objects are occluded, the ID exchange will occur. The Deep-SORT sort is an improved algorithm based on the idea of SORT. The reason why Deep-SORT is proposed is to reduce ID exchange by modifying area matching to feature matching. In Deep SORT, more reliable measures are used to replace the correlation measures, and CNN is used to train in large-scale pedestrian data sets and extract features, which has increased the network's robustness to loss and obstacles.

To estimate the volleyball trajectory, the existing Deep-SORT is an algorithm for object tracking, which evolved from SORT (simple online and real-time tracking). It uses the slow Kalman filter to predict the motion track of the detected object, and the Hungarian algorithm matches them with the newly detected object. Deep-SORT is easy to use and fast, becoming a popular algorithm for AI object detection and tracking. In this paper, we use the Deep-SORT algorithm to realize volleyball trajectory estimation and analysis with the proposed GE-YOLO. The framework is shown in Fig 3. The GE-YOLO network is used to detect volleyball in the video. After the video frames are input, it first enters the pre-trained YOLOv4 object detection network, and features are extracted through Darknet-53; secondly, we use a sparse sampling mechanism to locate ROIs and construct object graphs. By embedding graph convolution layers, the contextual information can be effectively mined for different-scale object detection. Thirdly, the predicted box information is input into the Deep-SORT algorithm for object feature modeling, matching and tracking; finally, output the results. The following figure shows the design of the algorithm process.

5. EXPERIMENTAL

In this section, we perform pre-trained experiments on MS COCO (Microsoft Common Objects in Context) dataset (Lin et.al, 2014). The MS-COCO data set is a data set built by Microsoft. It contains detection, segmentation, key points and other tasks. In the detection task, the MS-COCO data set contains 91 categories in total (80 categories are used for detection tasks, including the sports ball category). Each of the 82 classifications has more than 5000 instance objects, which helps to better learn the location information of each object. The number of objects in each category is also far more than in the

Figure 3. The architecture of volleyball trajectory estimation and analysis



PASCAL VOC dataset. Compared with the PASCAL VOC data set, the images in the COCO include natural images and common object images in life. The background is complex, the number of objects is large, and the size of objects is smaller, So the task on the COCO dataset is more difficult.

Compared with ImageNet, the COCO dataset does not have so many classifications, but there are more instance objects of each classification than ImageNet. COCO has 91 classifications, of which 82 classifications each have more than 5000 instance objects. These help to better learn the location information of each object, and the number of objects in each category is far more than the PASCAL VOC dataset. The MS COCO dataset has more object scene images than other datasets and can significantly improve the details of model learning. In the MS COCO detection task, we only use the pre-trained YOLOv4 weights and focus on volleyball detection.

We test the influence of graph-embedded improvement on the accuracy of the detector on the MS COCO (test-dev 2017) dataset. Then, we still compare the effectiveness of different tracking algorithms. At last, we show the result of volleyball trajectory estimation in a volleyball video by exploiting the proposed GE-YOLO and Deep-SORT.

1. Experimental setup

In MS COCO object detection experiments, some hyper-parameters settings are as follows: the training steps and the initial learning rate are set as 500,50 and 0.01. During training, we make the initial learning rate multiply with a factor of 0.1 at the 400,000 and the 450,000 steps, respectively; The mini-batch size is set as 8. In all our experiments, we only use one GPU for training. All experiments are trained with a 1080Ti or 2080Ti GPU. Other experimental settings are according to the paper (Bochkovskiy et.al, 2020). Moreover, in our proposed GE-YOLO framework, we adopt the pre-trained YOLOv4 weight for CSPParknet53, and these weights are frozen during network training.

2. Results of different object detectors

We compare different object detectors, i.e., YOLOv4, LRF, EFGRNet, YOLOv3, SSD, M2det, HSD, and SAPD, with the proposed GE-YOLO in terms of Backbone, AP, AP50, AP75, APS, APM, APL. We set the batch size as 1 and do not use the tensorRT mechanism. All the results are shown in Table 1.

Compared to other models, it confirms that YOLOv4, which uses many tricks, achieved better performance in terms of accuracy and speed. The AP and FPS of YOLOv4 are 43.0% and 31M. But YOLOv4 needs to perform object detection on the full image which leads to more computation. To

Table 1. The results of different models on the MS COCO dataset

Method	Backbone	AP(%)	AP_{50}(%)	AP_{75}(%)	AP_S(%)	AP_M(%)	AP_L(%)	FPS
GE-YOLO	CSPDarknet-53	46.8	66.2	48.3	26.1	48.4	57.5	34(M)
YOLOv4	CSPDarknet-53	43.0	64.9	46.5	24.3	46.1	55.2	31(M)
LRF	ResNet-101	37.3	58.5	39.7	19.7	42.8	50.1	31.3(M)
EFGRNet	ResNet-101	33.2	53.4	35.4	13.4	37.1	47.9	47.6(P)
YOLOv3	Darknet-53	31.0	55.3	32.3	15.2	33.2	42.8	45(M)
SSD	VGG-16	25.1	43.1	25.8	6.6	25.9	41.4	43(M)
M2det	VGG-16	33.5	52.4	35.6	14.4	37.6	47.6	33.4(M)
HSD	VGG-16	33.5	53.2	36.1	15.0	35.0	47.8	40(P)
SAPD	ResNet-50	41.7	61.9	44.6	24.1	44.6	51.6	14.9(P)

further reduce computation, we introduce a sparse sampling mechanism to select object-associated features. Moreover, we construct object graphs based on ROIs. By embedded graph convolution, contextual information can be effectively mined. The result of the proposed GE-YOLO shows the best performance. Compared to YOLOv4, the AP of GE-YOLO increases by 3.8%. Moreover, the speed of GE-YOLO is also fast than YOLOv4. Because the graph convolution can use fewer parameters to extract deep features. In our proposed GE-YOLO, we only use two graph convolution layers. All the results show that our proposed GE-YOLO model shows better performance in terms of accuracy and speed.

3. Results of different scales of volleyball detection

In this section, we compare the detection results of volleyball at different scales with our proposed GE-YOLO and other object detectors, i.e., YOLOv4, LRF, EFGRNet, and YOLOv3. All the results are shown in Fig. 4.

In Fig.3, we give the detection results for different scale volleyball. The red, blue, yellow, purple and pink boxes denote our proposed GE-YOLO, YOLOv4, LRF, EFGRNet, and YOLOv3, respectively. From the upper left corner to the lower right corner, the size of the volleyball is getting smaller and smaller. The experimental results show that when the size of the volleyball is large, all detectors can correctly detect the volleyball, although some of the detector frames will have some offset errors. When the size of the volleyball is small, some detectors cannot detect the volleyball correctly. The proposed GE-YOLO in this paper can make correct detection for volleyball of different scales. It further shows the robustness of the proposed algorithm to size.

4. Volleyball trajectory estimation and analysis

The goal of this paper is to realize volleyball trajectory estimation and analysis. In this section, we joint the proposed GE-YOLO framework with Deep-SORT for volleyball location and track tracking. The final effect is shown in Fig.5and Fig.6.

Figure 4. Comparison of different scales of volleyball detection

We selected the video of Japanese vs Thailand in the final of the 2019 Women's Volleyball Asian Championship. From the results, the proposed GE-YOLO can effectively detect volleyball and estimate the trajectory of volleyball.

5. Effects of different layer of graph layers

In this paper, we embedded two graph layers to exploit contextual information for small object detection. In this subsection, we give some experiments to validate related settings. We conduct experiments on the MS COCO dataset. We set the number of graph convolution layers as 1, 2, 3, 4 and 5. All the results are shown in Table 2. As the results shown in Table, when the number of graph convolution layer equal 2, the performance is the best.

Figure 5. The result of volleyball trajectory estimation in video 1

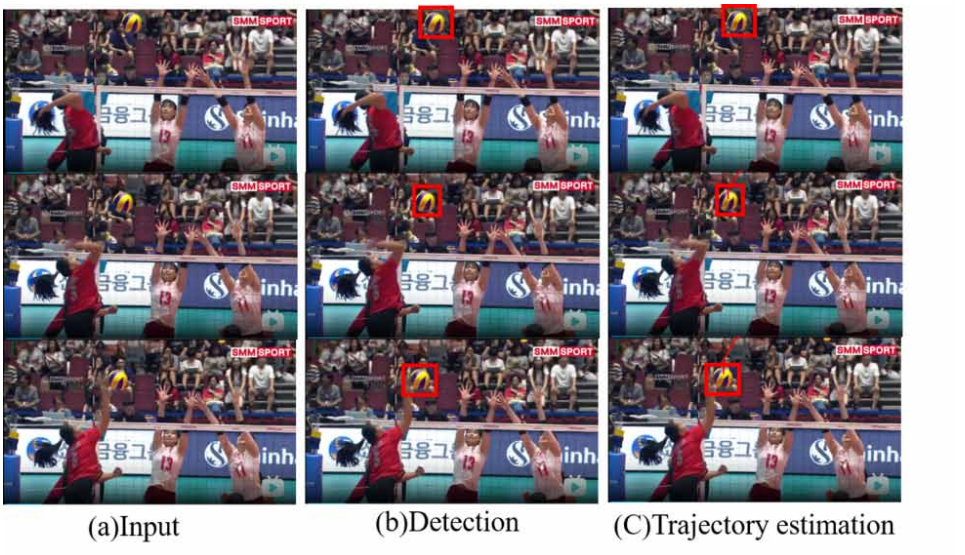


Figure 6. The result of volleyball trajectory estimation in video 2

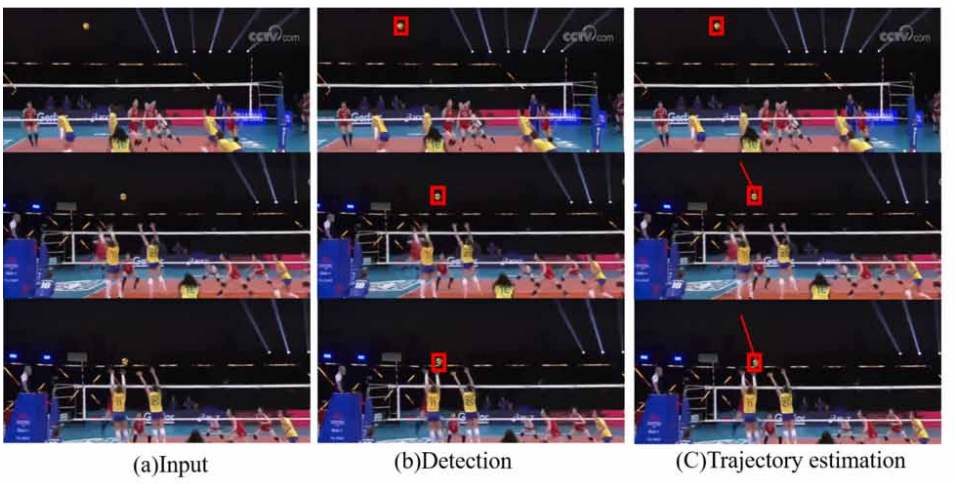


Table 2. Effects of different number graph convolution layers on the MS COCO dataset

Number	1	2	3	4	5
AP(%)	45.1	46.8	46.3	46.1	45.7

6. CONCLUSION

In this paper, we propose a novel graph-embedded object detector GE-YOLO for volleyball detection and trajectory estimation. Specifically, we reduce the computation by introducing a sparse sampling mechanism in our proposed framework. Based on object-associated features, we construct object graph data. Moreover, two Chevbev graph convolution layers are embedded object detectors for mining contextual information. The experimental results on the MS COCO dataset show that our proposed model achieves better performance compared to YOLOv4 and other object detectors. In addition, the GE-YOLO is more robust to detect different scales of volleyball compared to other detectors. At last, we joined the GE-YOLO with the Deep-SORT algorithm to estimate the trajectory of volleyball in the video sequence. Experimental results on volleyball videos in real scenes demonstrate the superiority of our models.

In future work, we will focus on the following two aspects: (1) Exploiting the complementary of multi-modal information has attracted great attention in recent years. So, designing an efficient multimodal fusion deep network will be a good option for FER. (2) In order to deploy deep networks on resource constrained devices, binary networks have become a better choice.

ACKNOWLEDGMENT

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

REFERENCES

Bochkovskiy, A., Wang, C. Y., & Liao, H. Y. M. (2020). *Yolov4: Optimal speed and accuracy of object detection*. arXiv preprint arXiv:2004.10934.

Cai, Z., Fan, Q., Feris, R. S., & Vasconcelos, N. (2016, October). A unified multi-scale deep convolutional neural network for fast object detection. In *European conference on computer vision* (pp. 354-370). Springer.

Chen, C., Liu, M. Y., Tuzel, O., & Xiao, J. (2016, November). R-CNN for small object detection. In *Asian conference on computer vision* (pp. 214-230). Springer.

Chen, L. C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *40*(4), 834–848.

Dai, Z., Cai, B., Lin, Y., & Chen, J. (2021). Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 1601-1610). IEEE.

Gao, X., Niu, S., Wei, D., Liu, X., Wang, T., Zhu, F., & Sun, Q. et al. (2022). Joint Metric Learning-Based Class-Specific Representation for Image Set Classification. *IEEE Transactions on Neural Networks and Learning Systems*.

Gong, L., & Cheng, Q. (2019). Exploiting edge features for graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 9211-9219). IEEE.

Jiao, L., Zhang, R., Liu, F., Yang, S., Hou, B., Li, L., & Tang, X. (2021). New generation deep learning for video object detection: A survey. *IEEE Transactions on Neural Networks and Learning Systems*.

Kipf, T. N., & Welling, M. (2016). *Semi-supervised classification with graph convolutional networks.* arXiv preprint arXiv:1609.02907.

Kisantal, M., Wojna, Z., Murawski, J., Naruniec, J., & Cho, K. (2019). *Augmentation for small object detection.* arXiv preprint arXiv:1902.07296.

Kong, T., Yao, A., Chen, Y., & Sun, F. (2016). Hypernet: Towards accurate region proposal generation and joint object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 845-853). IEEE.

Law, H. D. J. C. (2018). Detecting objects as paired keypoints. Lecture Notes in Computer Science, 765-781.

Li, C., & Cui, J. (2021). Intelligent sports training system based on artificial intelligence and big data. *Mobile Information Systems*.

Li, G., Muller, M., Thabet, A., & Ghanem, B. (2019). Deepgcns: Can gcns go as deep as cnns? In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 9267-9276). IEEE.

Li, J., Liang, X., Wei, Y., Xu, T., Feng, J., & Yan, S. (2017). Perceptual generative adversarial networks for small object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1222-1230). IEEE.

Li, Y., Chen, Y., Wang, N., & Zhang, Z. (2019). Scale-aware trident networks for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 6054-6063). IEEE.

Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2017). Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2117-2125). IEEE.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., . . . Zitnick, C. L. (2014, September). Microsoft coco: Common objects in context. In *European conference on computer vision* (pp. 740-755). Springer.

Lipton, A. (1998). Moving target detection and classification from real-time video. *Proc. of the 1998 Workshop on Applications of Computer Vision*.

Liu, D., Li, Y., Lin, J., Li, H., & Wu, F. (2020). Deep learning-based video coding: A review and a case study. *ACM Computing Surveys*, *53*(1), 1–35. doi:10.1145/3368405

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016, October). Ssd: Single shot multibox detector. In *European conference on computer vision* (pp. 21-37). Springer.

Najibi, M., Singh, B., & Davis, L. S. (2019). Autofocus: Efficient multi-scale inference. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 9745-9755). IEEE.

Roh, B., Shin, J., Shin, W., & Kim, S. (2021). *Sparse detr: Efficient end-to-end object detection with learnable sparsity*. arXiv preprint arXiv:2111.14330.

Rowley, H. A., Jing, Y., & Baluja, S. (2006). *Large scale image-based adult-content filtering*. Academic Press.

Simonyan, K., & Zisserman, A. (2014). *Very deep convolutional networks for large-scale image recognition.* arXiv preprint arXiv:1409.1556.

Tan, M., Pang, R., & Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 10781-10790). IEEE.

Tian, Z., Shen, C., Chen, H., & He, T. (2019). Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 9627-9636). IEEE.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., & Bengio, Y. (2017). *Graph attention networks*. arXiv preprint arXiv:1710.10903.

Wang, H., Tang, P., Li, Q., & Cheng, M. (2021). Emotion Expression With Fact Transfer for Video Description. *IEEE Transactions on Multimedia*, *24*, 715–727.

Wu, X., Sahoo, D., & Hoi, S. C. (2020). Recent advances in deep learning for object detection. *Neurocomputing*, *396*, 39–64.

Xiao, N., Yu, W., & Han, X. (2020). Wearable heart rate monitoring intelligent sports bracelet based on Internet of things. *Measurement*, *164*, 108102. doi:10.1016/j.measurement.2020.108102

Yamato, J., Ohya, J., & Ishii, K. (1992, June). Recognizing human action in time-sequential images using hidden Markov model. In CVPR (Vol. 92, pp. 379-385). Academic Press.

Yang, C., Huang, Z., & Wang, N. (2022). QueryDet: Cascaded sparse query for accelerating high-resolution small object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* (pp. 13668-13677). IEEE.

Ying, Z., You, J., Morris, C., Ren, X., Hamilton, W., & Leskovec, J. (2018). Hierarchical graph representation learning with differentiable pooling. *Advances in Neural Information Processing Systems*, 31.

Zhang, G., Cui, K., Wu, R., Lu, S., & Tian, Y. (2021). PNPDet: Efficient few-shot detection without forgetting via plug-and-play sub-networks. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision* (pp. 3823-3832). IEEE.

Zhao, G., Ge, W., & Yu, Y. (2021). GraphFPN: Graph feature pyramid network for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision* (pp. 2763-2772). IEEE.

Zhao, Q., Sheng, T., Wang, Y., Tang, Z., Chen, Y., Cai, L., & Ling, H. (2019, July). M2det: A single-shot object detector based on multi-level feature pyramid network. *Proceedings of the AAAI Conference on Artificial Intelligence*, *33*(01), 9259–9266.

Zhao, Z. Q., Zheng, P., Xu, S. T., & Wu, X. (2019). Object detection with deep learning: A review. *IEEE Transactions on Neural Networks and Learning Systems*, *30*(11), 3212–3232.

Zheng, W., Chen, S., Fu, Z., Zhu, F., Yan, H., & Yang, J. (2021). Feature selection boosted by unselected features. *IEEE Transactions on Neural Networks and Learning Systems*.

Zheng, W., Xu, C., Yang, J., Gao, J., & Zhu, F. (2018). Low-rank structure preserving for unsupervised feature selection. *Neurocomputing*, *314*, 360–370.

Zhu, F., Gao, J., Yang, J., & Ye, N. (2022). Neighborhood linear discriminant analysis. *Pattern Recognition*, *123*, 108422.