Highly Accurate Dynamic Gesture Recognition Method Based on Edge Intelligence

Lu Changkai[1], Liu Yiwen[1,2,3], Gao Yanxia[1,2,3](✉), Shi Yuanquan[1,2,3], and Peng Xiaoning[1,2,3]

[1] School of Computer and Artificial Intelligence, Huaihua University, Huaihua 418000, China
3129437633@qq.com
[2] Key Laboratory of Wuling-Mountain Health Big Data Intelligent Processing and Application in Hunan Province Universities, Huaihua 418000, China
[3] Key Laboratory of Intelligent Control Technology for Wuling-Mountain Ecological Agriculture in Hunan Province, Huaihua 418000, China

Abstract. In recent years, gestures have been widely used in many fields. For example, human-computer interaction, virtual reality, gesture translation, etc. The reason for its rapid development is due to the emergence of deep learning and artificial intelligence under today's society. Due to dynamic gestural interactions, such large intelligent models are often characterized by many parameters, large sample size, frequent parameter updates, and high communication volume. Based on this feature, we propose a cloud-based edge design architecture approach for gesture recognition based on an improved YOLOv5 network model optimization by changing different gestures, background interference, which uses a 21-layer model for the neural network. With the use of edge intelligence, the computational accuracy can be improved by 10.6% over the traditional YOLOv5 with a MAP value of 93.3%. The final recall rate is also improved by 3.6%. The parameter model is only 43.6% of the original one. This shows the practicality and operability of using edge computing as a technique in gesture recognition, as well as the small improvement cost and obvious effect.

Keywords: Edge Intelligence · Gesture Recognition · Neural Networks · YOLOv5

1 Introduction

With the continuous development of the Internet, people's living standard has been improving. Gesture interaction [1–3] began to gradually enter our life. We began to have higher requirements for smart devices [4], from operating machines with remote controls; voice recognition [5]; playing games with gamepads. To now, through gestures can command the device to complete the functions we need to complete, by posing a variety of postures can make people have an immersive gaming experience.

At the same time, while the terminal device is greatly enhanced, it enables a large number of services and response requests to be residual at the terminal. This is really

© ICST Institute for Computer Sciences, Social Informatics and Telecommunications Engineering 2023
Published by Springer Nature Switzerland AG 2023. All Rights Reserved
Z. Xiao et al. (Eds.): ICECI 2022, LNICST 478, pp. 213–225, 2023.
https://doi.org/10.1007/978-3-031-28990-3_15

difficult to load for server cloud servers [6] and the cost is very huge. To make these new technologies a win-win in terms of cost and experience, we introduced the concept of edge intelligence, on top of which edge computing is added to solve this problem.

For different response requests generated by terminals, we propose a method to judge the direction of data processing, and propose a method to dynamically allocate and process data based on an initial judgment of the complexity of current data processing during a specific time period with a large number of users, many variables, complex network conditions, and low resolution efficiency. The simple and easy data are processed at the edge [7] and then returned directly to the terminal, which can save a lot of resources and bandwidth consumption in data processing and can lead to a good experience. To a certain extent, it can also ensure the user's privacy and improve the stability of the software without giving a bad experience or even unresponsive errors due to the high latency of the network or server. As application software becomes richer and more powerful, the amount of computing required is also increasing. If the traditional cloud computing method is used, it will lead to the need to constantly upgrade the server configuration, which is a considerable expense for enterprises. Therefore, it is crucial to use edge intelligence in this aspect of gesture recognition.

This study focuses on improving YOLOv5 by using this as the support model for this edge computing.

YOLOv5 [8–12], is a single-stage target detection algorithm that adds some new improvement ideas to YOLOv4 [13], which results in a great performance improvement in both speed and accuracy. The purpose of improving this model is to maximize the edge computing capability and provide applications with characteristic capabilities such as ultra-low latency, high bandwidth, and real-time access [14]; distributed clouds [15–20] mainly have the following capability features: distributed, low latency, high performance, safe and reliable, green and energy efficient, and open capabilities. Able to manage its cloud nodes independently, it can provide more business capabilities than a single location cloud service.

Edge intelligence [21–23] is defined as an open platform that combines network, compute, storage, and application core capabilities as a whole to provide services at the end close to the object or data source, with the goal of providing services on the side of the data input or endpoint. This is done so that the data is not affected by the latency issues of the application. In this way, an efficient and cost-saving approach can be achieved. As shown in Fig. 1, the basic data processing and analysis model for this project.

Edge computing for gestural interaction systems [24] has the following advantages. First, it reduces the cost of the server and reduces the bandwidth usage. The series of data generated during the recognition process will be partially computed at the edge, which will reduce the process of responding to server requests. Let most of its computation produce results on the user end device. Second, it is fast and reliable. The process of sending data waiting for the server response takes a long time and when the server is busy, there may also be a problem of request failure, which may cause a series of problems such as packet loss in a complex network environment. Finally, the stability of the system is enhanced. Malicious accesses are reduced, and the inclusion of edge computing can effectively reduce malicious attacks [25–29] and prevent most of the problems of wasted

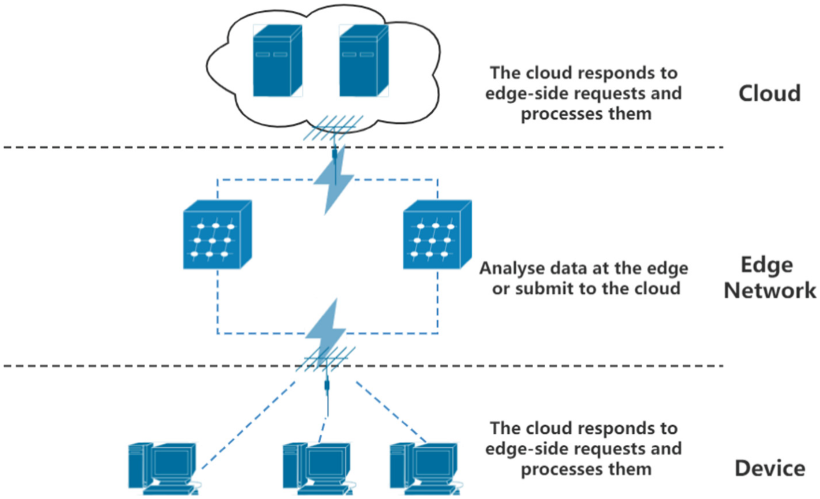

Fig. 1. Edge Intelligent Data Delivery Model

server resources. It also enhances confidentiality, and some confidential information can be processed at the edge without server processing, which effectively [30] improves the security of data.

2 Gesture Interaction Design

2.1 Holistic Approach

According to the requirements, the method design of the gesture recognition system based on edge computing is proposed, based on the architecture designed according to three layers: cloud - edge computing node - user terminal.

The server side is mainly responsible for user information management, big data analysis of recognition results and visual data display, model building, and response requests for special complex scenarios. The edge side is mainly responsible for collecting and analyzing the terminal data, and the preliminary analysis performed by the edge side can lead to the result or need to request the cloud side for further analysis and processing. And each time the recognition result log is transmitted back to the cloud. The cloud can analyze the data sent back by the edge layer so as to carry out the analysis of the recognition model, with the purpose that the recognition data provided to the edge layer next time can be more accurate, enhance the efficiency of recognition, and realize the process of continuous optimization of recognition.

2.2 Gesture Interaction Processing

Gesture interaction [31] requires large processor power [32] and storage space for images and videos, as well as high requirements for network stability. In gesture control, gesture-operated games, real-time machine translation, and in the interaction process, most human communication is not only delivered through language, voice, intonation, facial expressions, gestures, etc. This time delay often causes many problems, and if in for a simple gesture, it takes a lot of time to respond, then these applications will lose the value he originally should have. Resulting in a very poor user experience, or even the inability to use properly.

In this regard, it seems that interaction is often accompanied by complex computation as well as reasoning [33–36], and he is different from the previous recognition, which may be just a picture, and the picture is not like interaction, the picture needs to be collected and changed in real time. And once the lag occurs, then it will cause a lot of important information loss, incoherence and a series of problems. To solve these problems, it needs to have strong hardware and good algorithms as support.

In this problem, a set of cloud-based edge-end model is designed. First, the information collected in the terminal is first handed over to the edge end for judgment and processing, and when the edge end thinks that the task is within the processing capability of the edge equipment, then the request from the terminal is processed immediately and the processed information is transmitted back to the terminal in time. When the edge device checks that this recognition may require more computation or cannot be completed on the edge device, it forwards the request to the cloud for request response. This design makes it possible to make the valuable computing resources of the server, fully utilized.

Based on this design, more user requests will be completed at the edge for response, greatly improving efficiency. It also saves valuable bandwidth resources and reduces latency. There is also an improvement in security. These interactive devices, often collect a large amount of user information, and edge intelligence is deployed and trained on this side closer to the user, which can effectively reduce data hijacking [37, 38], tampering, and other problems caused by data during network transmission. The specific design is shown in Fig. 2.

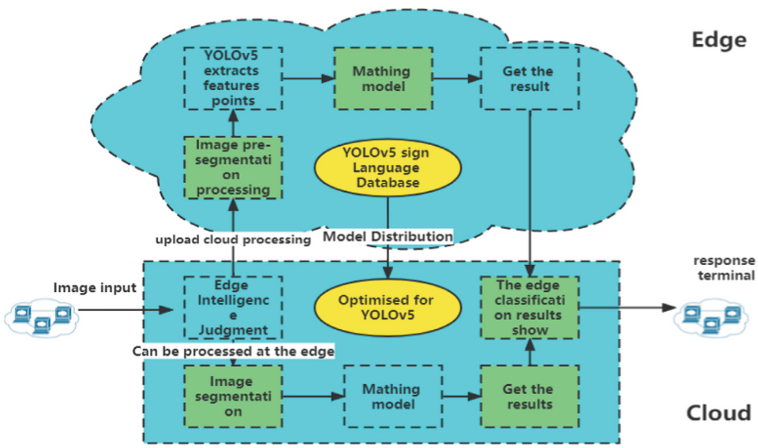

Fig. 2. Gesture recognition framework

The image is first input from the end device, uploaded to the edge device, and pre-analyzed on the edge device. If the initial analysis is relatively simple, it can be sent to the server at the edge for processing and response to execute the trained and optimized YOLOv5, and uploaded to the server when the analysis of that image may require a larger amount of operations or when the resources of the edge device cannot be satisfied. This is because the server [39] has a richer database and analysis model. Ultimately it is the edge server that responds to the data directly to the user. This design makes more rational use of valuable server resources, speeds up the response [40] time and requests of the whole system, and reduces to some extent the resource overhead of the server and the risk of server downtime caused by illegally conducted malicious attacks.

3 Intelligent Processing Solutions at the Edge

3.1 YOLOv5 Algorithm

In June 2020, the first version of the algorithm YOLOv5 was officially released, and its release has caused extensive research, discussion, and use in the field of computer vision. Unlike previous versions of the YOLO model, YOLOv5 is actually a model family containing four models (YOLOv5s, YOLOv5m, YOLOv5l, and YOLOv5x), including three main components, as shown in Fig. 3

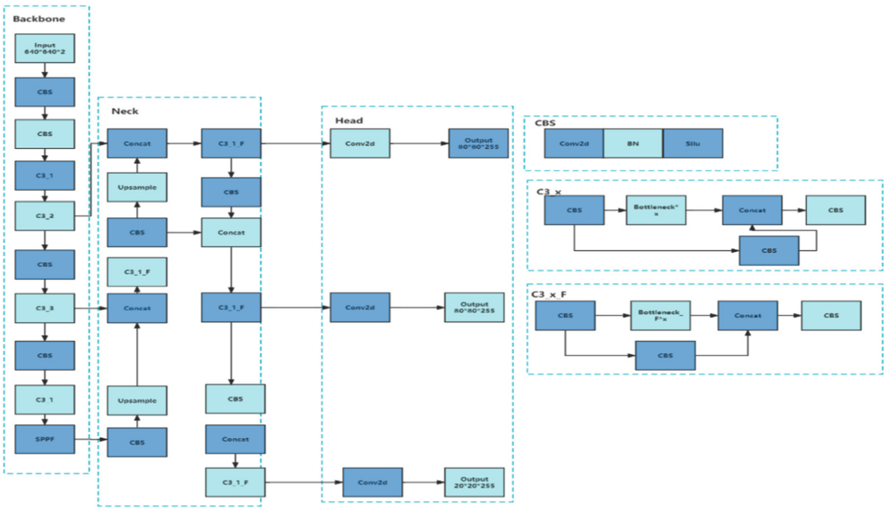

Fig. 3. Original YOLOv5 network structure

Backbone

The convolutional neural network that aggregates and extracts image features on different subcategories of images, including the Focus structure and the CSP structure.

Neck

Using the FPN + PAN structure, the network layer of blended image features is used to pass the feature images of relevant targets to the prediction layer.

Head

Has multiple prediction scales to perform target prediction on images, generate target bounding boxes and predict relevant classes. Among them, YOLOv5s is the smallest model in the YOLOv5 series, but it is still resource-consuming in some cases where the environment is complex and has more elements. For edge devices, the model size and computation are overwhelming, and in addition to that, memory reads and writes and loading large scale models will incur a huge additional overhead. This algorithm can be run on the server side with high performance. For edge-side devices with limited computing [41] power. Based on this algorithm, this paper proposes a faster detection algorithm based on a more lightweight approach to reduce resource overhead and improve efficiency [42] without sacrificing accuracy as much as possible. The efficiency of the model is better balanced with the availability.

3.2 Improved YOLOv5 Algorithm

In order to improve the accuracy as well as the performance of the gesture image detection algorithm for this situation. In this paper, the following improvements are made to YOLOv5.

Introduce a cooperative attention mechanism in the network. The reason for introducing this mechanism is to, divide the recognized images into different weight layers so that more useful features can be extracted and other distracting factors, such as extraneous factors of the environment, can reduce the interference with the originally main recognized objects.

A layer of complex target detection is added to the original three detection layers of different scales. It is dedicated to the corresponding recognition [43] of those images that are not recognized accurately.

At the output side, a boundary loss function is introduced to improve the problem of loss function in the original network [44].

3.3 Coordinate Attention

A Coordinate Attention block can all be considered as a computational unit with the purpose of enhancing feature representation in Mobile Network. He can take any intermediate feature tensor:

$$X = [x_1, x_2 \ldots, x_c] \in R^{c \times H \times w} \tag{1}$$

As input and by transforming the output with the same size as the tensor and with enhanced representation:

$$Y = [y_1, y_2 \ldots, y_c] \tag{2}$$

In order to describe CA attention more clearly, the SE block is discussed here first. Structurally, it is possible to divide the SE block into 2 steps Squeeze and Excitation, which are used for the input of global information and the adaptive Re-weight of the channel relationship, respectively. Conditional on the input X, the squeeze step of the cth channel can be expressed as:

$$z_c = \frac{1}{H * W} \sum\nolimits_{i=1}^{H} \sum\nolimits_{j=1}^{W} x_{c(i,j)} \tag{3}$$

where z_c is the output associated with the cth channel.

The input X comes from a convolutional layer with a fixed kernel size. The purpose of Excitation is to completely capture the dependencies between channels, which can be expressed as:

$$\widehat{X} = X \cdot \sigma(\hat{z}) \tag{4}$$

Coordinate Attention encodes channel relationships and long-term dependencies by precise location information, which is divided into two steps: Coordinate information embedding and Coordinate Attention generation, as shown in Fig. 4.

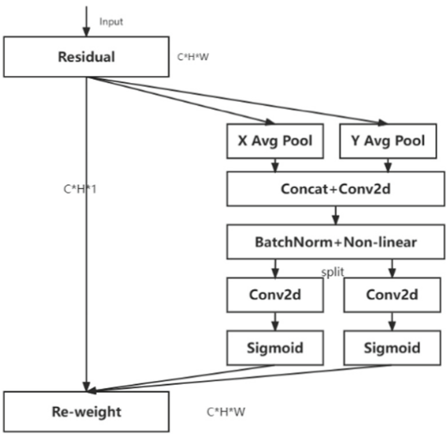

Fig. 4. Structure of the Coordinate Attention mechanism

Coordinate information embedding [6, 45], the global pooling approach is commonly used for global encoding of spatial information for channel attention encoding, but it makes it difficult to preserve location information because it compresses global spatial information into channel descriptors. To motivate the attention module to capture remote spatial interactions with precise location information, this paper decomposes the global pooling into a pair of one-dimensional feature encoding operations according to the following equation:

$$z_c = \frac{1}{H * W} \sum_{i=1}^{H} \sum_{j=1}^{W} x_{c(i,j)} \tag{5}$$

Specifically, given the input X, each channel is first encoded along the horizontal and vertical coordinates using a pooling kernel of size (H, 1) or (1, W), respectively. Thus, the output of the cth channel with height h can be expressed as:

$$Z_c^h(h) = \frac{1}{W} \sum_{0 \le i < W} x_c(h, i) \tag{6}$$

Similarly, it can be shown that the output of channel c with width w can be written as:

$$Z_c^w(w) = \frac{1}{H} \sum_{0 \le j < W} x_c(j, w) \tag{7}$$

The last is Coordinate Attention generation, after the transformations in the information embedding, this part performs the concatenate operation on the above transformations and then uses the convolutional transform function to transform them.

$$f = \delta(F_1(\left[z^h, z^w\right]) \tag{8}$$

$$g^h = \sigma(F_h(f^h)) \tag{9}$$

$$g^w = \sigma(F_w(f^w)) \tag{10}$$

Finally, the output Y of the Coordinate Attention Block can be written as:

$$y_c(i, j) = x_c(i, j) \times g_c^h(i) \times g_c^w(j) \tag{11}$$

3.4 Improved Overall Network Model

An improved overall network modelSeveral additional corresponding feature extraction layers are also added, that is, after the 17th layer of the network, continue to use the C3 model block and CONV module for feature extraction, and feature sampling in the 20th layer to further expand it. In the 21st layer, the sampled feature map of size 640*640 is fused with the feature map obtained from the second layer in the extraction network to obtain the detection of complex gestures.

4 Experimental Environment and Experimental Data

4.1 Training of Models

All experiments in this paper are executed under Windows 10 operating system, with AMD Ryzen 5 4600H processor; 16G memory and NVIDA GeForce GTX 1650 graphics card. This paper is based on the deep learning framework PyTorch1.10, and the built environment includes Anaconda3.0, python3.7, and CUDA11.1.1.

In this paper, precision, recall, and mean average precision metrics are used to synthesize the effect of improving the model, and the specific calculation methods of each metric are as follows.

$$Pre = \frac{TP}{TP + FP} \tag{12}$$

$$Re = \frac{TP}{TP + FN} \tag{13}$$

$$MAP = \frac{\sum_{i=1}^{k} AP_i}{k} \tag{14}$$

In the above equation, Pre denotes the precision of recognition [44], Re denotes the recall rate, and MAP denotes the mean average precision. In this experimental design, the average precision with a threshold of 0.5 is used as a measure, APi denotes the average precision of the ith category, TP denotes the number of positive samples predicted by positive samples, FP denotes the number of positive samples predicted by negative samples, FN denotes the number of positive samples predicted by negative samples, and k denotes the number of categories in the experiment. Comparing with the training data of the original YOLOv5 algorithm, we can see that the improved YOLOv5 algorithm has improved the MAP, precision, and recall in comparison with the original one. The MAP value is 93.3%, and the final recall rate is 87.3%, up from 83.7%, an improvement of 3.6%. The memory occupied by the model decreased from 7.8 MB to 3.4 MB. The experiment shows that all the metrics have improved and are better than the original YOLOv5's algorithm. And after 60 rounds, these metrics are basically converged, which proves the effectiveness of this algorithm. As shown in Figs. 5, 6 and 7.

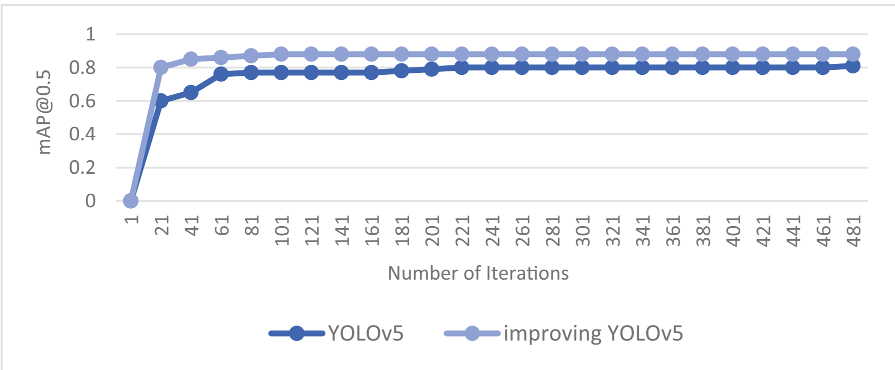

Fig. 5. MAP

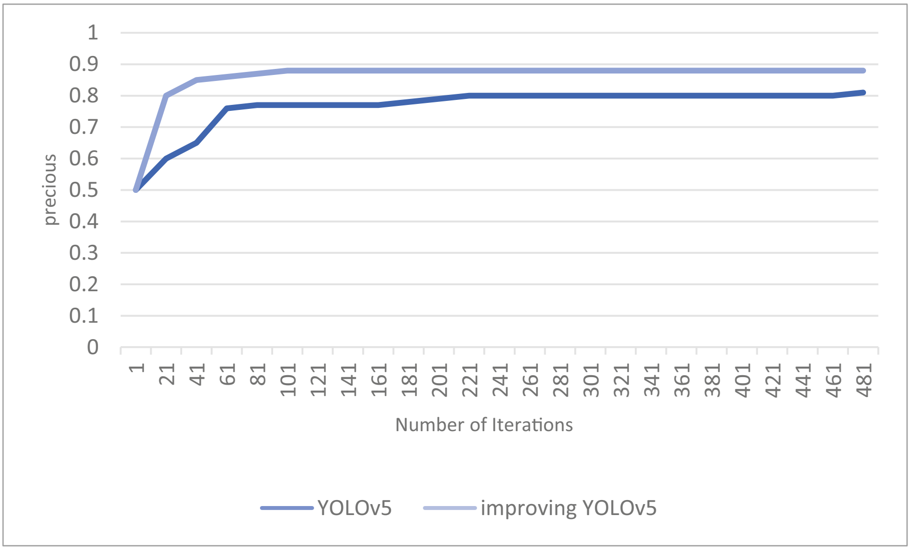

Fig. 6. Precious

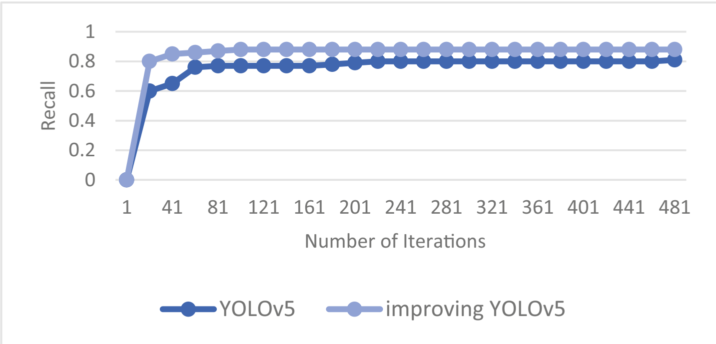

Fig.7. Recall

4.2 Comparison of Different Algorithms

In order to be able to show that the algorithm in this paper outperforms other algorithms, we use the same data set, on the same configuration of computers, with the same setup. They were trained with as few extraneous variables as possible to influence the environment, and the final results are shown in Fig. The control algorithm is YOLOv4 and SSD, which are commonly used for detection and target recognition. 4.8% accuracy improvement compared to the original YOLOv4, 83% of the original computation time, and only 5.1% of the original model size can be seen that the improved algorithm is still better than other [46] algorithms in general and has a better prospect for edge devices.

Table 1. Comparison of detection performance of each model

Algorithm	MAP	Recall/%	Times/s	Params/MB
SSD	86.3	77.9	0.0617	26.15
YOLOv4	88.2	81.6	0.01	61.58
Refine YOLOv5	92.5	87.9	0.00083	3.14

5 Conclusion

In this paper, we propose an improved YOLOv5 gesture recognition method for edge computing, which targets the problem of complex neural networks running on edge terminal devices with insufficient computing power. Improvements to YOLOv5 are made from the introduction of Coordinate Attention and the introduction of a feature detection layer. The feasibility is experimentally verified and the accuracy as well as efficiency is balanced to reduce the performance requirements, as well as the resource consumption.

In future practical engineering applications. The gesture recognition model should be dynamically adjusted to the actual scenario. When the gesture is not particularly complex and there is not much interference from external factors, the number of detection layers should be appropriately reduced to reduce the size of the model; or the number of detection layers can be appropriately increased to make the gesture recognition more accurate if the performance of the edge-end device is detected to be high.

This study is an experimental use case for gesture recognition, and the accuracy and effectiveness of the improved algorithm are verified. It also proposes a feasible research solution for the study of computer vision tasks under edge computing.

Acknowledgment. This work was supported in part by the Scientific research projects funded by the Department of education of Hunan Province (No. 21C0628, No. 20C1472 and No. 22C0668), the Huaihua University Double First-Class initiative Applied Characteristic Discipline of Control Science and Engineering(No. ZNKZN2021–10), the National Natural Science Foundation of China (No. 62172182), the Hunan Provincial Natural Science Foundation of China (No. 2020JJ4490), the Project of Hunan Provincial Social Science Foundation (NO. 21JD046), the Huaihua University Project (No. HHUY2019–25), the Philosophy and Social Science Achievement Evaluation Committee of Huaihua (No. HSP2022YB40) and the Science and Technology Innovation 2030 Special Project Sub-Topics(No. 2018AAA0102100). Hunan University Students' Innovation and Entrepreneurship Training Program (2021–3502).

References

1. Huiming, Z.: Design and implementation of a containerized edge computing platform for intelligent video surveillance scenarios. University of Electronic Science and Technology, MA thesis, pp.6–23 (2022)
2. Xiaoyue, M.: Research on the design of children's cognitive development and learning products based on gesture interaction: the example of Kids' Companion application. Electron. Compon. Inf. Technol. **6**(05), 83–87 (2022)

3. Lin, X.Y., et al.: Real-time interaction system for hovering true 3D display based on Leap Motion gesture recognition. Liq. Cryst. Display **37**(05), 654–659+546 (2022)
4. Zhang, W., et al.: A review of dynamic gesture understanding and interaction. J. Softw. **32**(10), 3051–3067 (2021)
5. Zeng, F., Li, Q., Xiao, Z., Havyarimana, V., Bai, J.: A price-based optimization strategy of power control and resource allocation in full-duplex heterogeneous macrocell-femtocell networks, IEEE Access, **6**, 42004–42013 (2018). -8
6. Jiang, H., Cao, H., Liu, D., Xiong, J., Cao, Z.: SmileAuth: using dental edge biometrics for user authentication on smartphones. Proc. ACM Interact. Mob. Wearable Ubiquitous Technol. **4**(3), 84:1–84:24 (2020)
7. Liu, D., Cao, Z., He, Y., Ji, X., Hou, M., Jiang, H.: Exploiting concurrency for opportunistic forwarding in duty-cycled IoT networks. ACM Trans. Sens. Netw. **15**(3), 31:1–31:33 (2019). -9
8. Lumeng, C., Yanyan, C., Min, H., Xinguang, X.: A flame detection method based on improved YOLOv5. Comput. Eng. 1–17(2022)
9. Wenbo, Z., et al.: An improved YOLOv5-based method for sugarcane stem node identification. J. Huazhong Agric. Univ. 1–9(2022)
10. Zhang, Z.M., Wu Z.: A dense pedestrian detection method based on improved YOLOv5. Appl. Sci. Technol. 1–7(2022)
11. Xu, H.-D., Ma, W., Yu, T., Xingxing, L., Zheng, Y.-J., Tian, Z.-W.: A YOLOv5 deep learning based method for tea shoot yield estimation. J. China Agric. Univ. **12**, 213–220 (2022)
12. Yang, G-L., Zhao, M., Huang, C., Huang, J-W.: Improvement of mask wearing detection algorithm by YOLOv5s. J. Heilongjiang Inst. Technol. (Compr. Ed.), **22**(10), 51–57 (2022)
13. Hao, S., Xingfa, D., Jun, W., Zhiyuan, C.: Improved YOLOv4-tiny-based lightweight on-campus pedestrian target detection algorithm. Comput. Eng. Appl. 1–12 (2022)
14. Yanhua, Z., et al.: MEC and blockchain empowered UAV-assisted IoT resource optimization. J. Beijing Univ. Technol. 1–909 (2022)
15. Liang, C., Kang, Y., Jianjun, Z.: Research on distributed cloud storage for electric safety apparatus management. Electr. Appl. **41**(02), 32–35 (2022)
16. Hao, C.: Research and application of online education system based on distributed cloud computing. East China Normal University, MA thesis, pp.10–27(2021)
17. Wang, X.: A distributed cloud-based approach to virtual machine template management. Metall. Autom. **45**, 169–176 (2021)
18. Shengli, G.: An analysis of distributed cloud data center solutions. China Financ. Comput. 96 (2022)
19. Xue, H.-M.: A distributed cloud computing data mining approach based on MapReduce. J. Anyang Normal Coll. **05**, 24–27 (2020)
20. Liu, Y, Zhang, N.: An integrated processing platform for urban and rural planning big data based on distributed cloud data center. J. Henan Urban Constr. Inst. **29**(03), 81–86+92 (2020)
21. Mao. J.: Deep reinforcement learning based collaborative caching for edge intelligence. East China Normal University, MA thesis, pp.63–70(2022)
22. Wenyong, D.: Exploring the application of integrated industrial Internet edge computing system based on edge intelligence plus network security. Autom. Expo **39**(01), 83–87 (2022)
23. Qiao, D.W., et al.: Edge intelligence: research advances and challenges. Radio Commun. Technol. **48**(01), 34–45 (2022)
24. Sowmya, M., et al.: SmartDefense: A distributed deep defense against DDoS attacks with edge computing. Comput. Netw. **209**, 108874 (2022). prepublish
25. Wei, Q., et al.: Research on face recognition method based on edge-cloud collaboration. Comput. Sci. **49**(05), 71–77 (2022)
26. Zifeng, W., et al.: Wider or deeper: revisiting the ResNet model for visual recognition. ArXiv abs/1611.10080 (2019)

27. Chengmin, L., et al.: Image recognition of peach tree pests based on multiscale attention residual network. J. Shandong Agric. Univ. (Nat. Sci. Ed.) **53**(02), 253–258 (2022)
28. Chen, Y., Dang, S.W., Nie, B.: A loopback detection algorithm based on ResNet model. Intell. Comput. Appl. **12**(08), 196–199 (2022)
29. Hu, Z., Zeng, F., Zhu, X., Fu, B., Jiang, H., Chen, H.: Computation efficiency maximization and QoE-provisioning in UAV-enabled MEC communication systems. IEEE Trans. Netw. Sci. Eng. **8**(2), 1630–1645 (2021)
30. Li, Z., et al.: A study on tree species identification based on improved ResNet34 network. Foreign Electron. Meas. Technol. **41**(07), 119–125 (2022)
31. Shuang, W.: Research and implementation of flower recognition algorithm based on machine learning. Chengdu: University of Electronic Science and Technology (2018)
32. Qian, C., Liu, D., Jiang, H.: Harmonizing energy efficiency and QoE for brightness scaling-based mobile video streaming. In: IWQoS, pp. 1–10 (2022)
33. Aifang, C.: Research on edge computing technology for optimization of target detection execution efficiency. Beijing Jiaotong Univ. 40–65 (2021)
34. Li, L.: Research on the theory and application of FDTD and its hybrid algorithm in microstrip antenna computing, Harbin Engineering University, PhD dissertation, pp.10–15(2015)
35. Liu, Y.: Research on coverage calculation and optimization methods for mobile communication networks. University of Science and Technology Beijing, PhD dissertation (2021)
36. Yang, Y.: Research on Internet inter-domain routing hijacking and its defense. Tsinghua University, PhD dissertation, pp.45–47 (2020)
37. Heng Li, L.-L., Yue, B.-S., Yung, S.-C.: Threat detection of hijacking attacks on vulnerable data under tenacious robust information. J. Xi'an Eng. Univ. **34**(06), 92–98 (2020)
38. Zhu, X., et al.: Toward accurate vehicle state estimation under non-Gaussian noises. IEEE Internet Things J. **6**(6), 10652–10664 (2019)
39. Liu, D., Hou, M., Cao, Z., He, Y., Ji, X., Zheng, X.: COF: exploiting concurrency for low power opportunistic forwarding. In: ICNP, pp. 32–42 (2015)
40. Jiang, H., Zhu, X., Li, Z., Xu, J., Zeng, F., Wang, D.: An energy-efficient framework for internet of things underlaying heterogeneous small cell networks. IEEE Trans. Mob. Comput. **21**(1), 31–43 (2022)
41. Liu, D., Wu, X., Cao, Z., Liu, M., Li, Y., Hou, M.: CD-MAC: a contention detectable MAC for low duty-cycled wireless sensor networks. In: SECON, pp. 37–45 (2015)
42. Hu, J., et al.: BlinkRadar: Non-intrusive driver eye-blink detection with UWB radar. In: Proceedings of IEEE ICDCS (2022)
43. Jiang, H., Dai, X., Xiao, Z., Iyengar, A.K.: Joint task offloading and resource allocation for energy-constrained mobile edge computing. IEEE Trans. Mob. Comput. (2022). https://doi.org/10.1109/TMC.2022.3150432
44. Ali, T.A.A., Xiao, Z., Sun, J., Mirjalili, S., Havyarimana, V., Jiang, H.: Optimal design of IIR wideband digital differentiators and integrators using salp swarm algorithm. Knowl Based Syst. **182**(15), 104834 (2019). https://doi.org/10.1016/j.knosys.2019.07.005
45. Zeng, Z., Li, Q., Xiao, Z., Havyarimana, V., Bai, J.: A Price-based optimization strategy of power control and resource allocation in full-duplex heterogeneous Macrocell-femtocell networks. IEEE Access **6**, 42004–42013 (2018)
46. Liu, D., Cao, Z., Hou, M., Rong, H., Jiang, H.: Pushing the limits of transmission concurrency for low power wireless networks. ACM Trans. Sens. Netw. **16**(4), 40:1–40:29 (2020)