Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2024 SCPE. Volume 25, Issues 4, pp. 2973–2984, DOI 10.12694/scpe.v25i4.2929

APPLICATION OF MULTI-OBJECTIVE EVOLUTIONARY ALGORITHMS FOR MULTIDIMENSIONAL SENSORY DATA PREDICTION AND RESOURCE SCHEDULING IN SMART CITY DESIGN

LIYA LIU*

Abstract. Multidimensional sensory data prediction and resource scheduling are paramount challenges in the design of smart cities. This paper delves into the utilization of multi-objective evolutionary algorithms to enhance the accuracy and efficiency of target detection through optimized YOLO_v3 network models. By integrating the YOLO_v3 model with the K-means++ algorithm for Anchor_Box generation, the novel approach exhibits superior adaptability and flexibility, particularly in handling variable-sized feature pattern mappings. This adaptability better caters to the detection of targets of diverse sizes, thus elevating the performance and precision of target detection algorithms. To further scrutinize the YOLO-v3 joint algorithm's performance in urban traffic detection, P-R curves were plotted for various loss types on the NEU-DET dataset. Comparative analysis of these curves highlights the optimized algorithms' superiority in detecting various types of losses in urban model completeness. Additionally, practical application analysis revealed that the optimized monitoring results outperform the detection time of the original YOLO-v3_means++ network model on FP_GA. Notably, post-processing with C-FENCE can reduce average single-frame image detection time to 2.01 seconds, while convolutional degree-level fusion with the BN layer cuts it down to 2.25 seconds. In summary, the FP_GA-based YOLO-v3_means++ network algorithm offers superior detection capabilities, and the multi-objective evolutionary algorithm's optimization of the YOLO-v3 model enhances target detection performance and precision.

Key words: depth-based learning network; multi-objective evolutionary algorithm; YOLO-v3_means++; multi-dimensional perception; smart city design

1. Introduction.. With the deepening development of economic globalization, every scientific and technological development and progress of human society will have a profound impact on urban development [1]. In recent years, the concept of a smart city has attracted widespread attention from all walks of life, and major cities have been increasing their efforts in the development of smart cities and smart landscapes. People all hope to rely on smart city construction to make their lives more efficient and convenient. Fundamentally, the smart city concept is people-oriented [2], advocating the organic integration of information technology and knowledge to promote the development of urban wisdom and innovation. In the multi-dimensional perception design of the smart city, the public transportation-oriented urban development model (T-O-D) can be integrated into the design of transportation infrastructure. In the design practice, the traffic relationship on the street should be clarified first. This design helps to highlight the humanistic environment of the city and effectively improve the multi-dimensional design effect of the smart city [3].

Smart city refers to an urban development model that applies advanced technological means such as information technology and the Internet of Things (IoT) to comprehensive data collection, analysis, and management of the city to realize the efficient use of urban resources, intelligent services, and improved quality of life. In the construction of a smart city, target degree detection is an important technology [4-5], which is used to monitor various objects and events in the city in real time. FPGA is a programmable hardware device with parallel processing and high-performance computing capability. FPGA can be used to accelerate the execution of algorithms in target degree detection tasks, providing real-time performance and low latency [6-7]. Since smart cities need to process a large amount of data and make decisions in real-time, FPGA can provide efficient computational power so that the target degree detection system can quickly and accurately identify and track various targets in the city, such as urban traffic detection. Monitoring and analyzing the flow of vehicles on the road in real time is of great significance for the construction of smart cities and other aspects [8]. Since the con-

*School of Urban Construction and Engineering, Wenhua College, Wuhan, Hubei 430074, China (Liya_Liu23@outlook.com)

cept of target degree detection was put forward, many scholars [9-11] have been plowing deeper and deeper into this field, and the basic theory of target degree detection class of algorithms ushered in vigorous development but was always limited by the hardware level. With the development of science and technology, hardware can gradually support the algorithms in the arithmetic demand, target degree detection algorithms are also widely used in various fields of life, such as face recognition, intelligent transportation, industrial detection, etc. [12].

In terms of traditional target detection algorithms, there are two mainstream directions of target degree detection algorithms based on deep-type learning3: the first is the two-stage deep-type learning target degree detection algorithms based on candidate frames represented by the R_C-NN46 series of algorithms [13], and the second is the one-stage deep-type learning target degree detection algorithms based on regression methods represented by the YOLO_7 series of algorithms. These depth-based learning target degree detection algorithms are essentially large convolutional neural architecture networks containing millions of neural unit connections, which require more than a billion operations to process at a time [14].

Multi-objective evolutionary algorithms are widely used in smart cities to optimize energy consumption, traffic management, and environmental monitoring. In terms of energy consumption, algorithms can be used to reduce energy consumption by adjusting the brightness of street lights or scheduling power supply plans. In traffic management, it can optimize traffic light timing, bus routes or parking space allocation. In environmental monitoring, air quality, noise pollution or water quality changes can be predicted to provide a basis for policy making. Multi-objective evolutionary algorithms can help smart cities achieve efficient and sustainable development. FP_GAs is often used in image processing due to their two main features of real-time pipelined operations and high real-time performance. FP_GAs provide a pipelined structure that matches well with product-based neural architecture network algorithms [15]. Tang et al. [16] designed a highly efficient DNN training velocimeter called E-F-Tra with a unified channel-level parallelism-based convolutional kernel, which allows for end-to-end training on a resource-limited, low-power edge-level FP_GAs for end-to-end training. A data-based reconfiguration method with intra-block sequential memory allocation and weight-based reuse is developed. To achieve high energy efficiency on edge FP_GAs, an analytical model of computational and memory resources for automatic scheduling is developed. High computational efficiency is provided by employing dynamic tiling, level fusion, and datatype layout optimization. A new generalized SA is designed to handle multidirectional convolution efficiently. The framework was tested using three complex C-NNs: Open-U-Net-E [17], and the optimization of the architecture achieved a 2.3x performance improvement. Sait et al [18] proposed a C-NN gas pedal and its automated design methodology, which employs a meta-heuristic approach to partition the available FP_GA-funded sources for the design of a Multi-CLP-type gas pedal. Its proposed design tool uses simulated annealing-type and forbidden search-type algorithms to find the number of Cs and their corresponding configurations required to achieve optimal performance on a given target FP_GA device [19]. Literature [20] deploys a speeder on an FP_GA that combines sparse Winograd convolution, a small set of pulsed arrays, and a layout design with a plannable memory to achieve a better performance performance. However, the approach does not sufficiently consider the advancement of memory technology, and the performance may be further improved by rationally optimizing the memory. Literature [21-22] designed the YOLO-v3_means++ model gas pedal, which greatly reduces off-chip class access through binary-type weights and low-bit-type activation operations. To reduce the computational complexity, this gas pedal employs Winograd-C_NN and maximizes the data reuse with a row buffer structure [23]. Multi-objective evolutionary algorithms find equilibrium solutions between multiple objectives by simulating biological evolution and are suitable for dealing with the complex challenges of predicting sensory data and resource scheduling in smart cities due to their ability to deal with conflicting objectives, uncertainty, and large-scale problems and to accelerate the search process

2. Heterogeneous FP_GA Architecture. To achieve efficient urban traffic detection and analysis, it is important to embed target degree detection class algorithms into composite systems. In this regard, porting the YOLO-v3_means++ algorithm to the FP_GA platform is a key research task.YOLO-v3_means++ is a real-time target degree detection class algorithm, and by porting it to FP_GA, it can take full advantage of the acceleration of the hardware [24-28] to realize real-time detection and analysis of the urban traffic to provide fast response data support for traffic management and intelligent transportation system to provide fast response data support. Porting the YOLO-v3_means++ algorithm to the FP_GA platform also helps to meet the low-power requirements of embedded systems.FP_GAs are reconfigurable and can be customized according

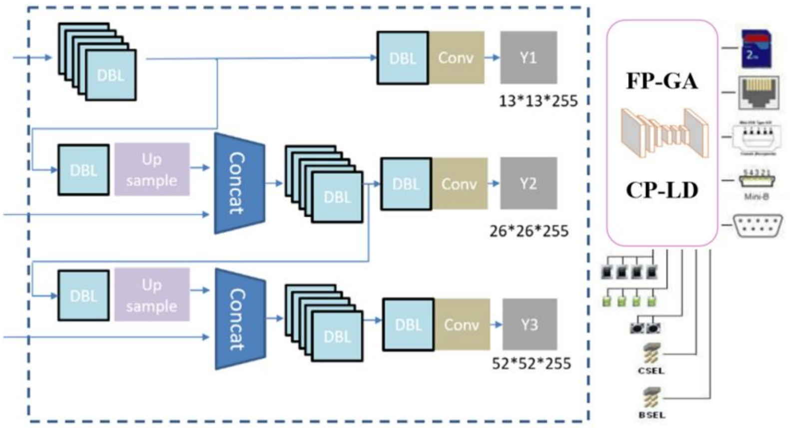

Fig. 2.1: Financial data based on mobile port photos and PDF invoices

to the needs of specific application scenarios to reduce power consumption and improve energy efficiency. By porting YOLO-v3_means++ to FP_GA, low-power urban traffic detection, and analysis can be realized while maintaining high detection performance, which is suitable for some embedded application scenarios with high energy consumption requirements.

2.1. Pre-processing of data and information rate enhancement of data classes. In smart cities, sensory data prediction and resource scheduling are closely related. Through real-time monitoring and prediction of various types of sensory data, we can more accurately understand the city's operating conditions and demand, so as to optimize the allocation of resources. The first step in the process of accelerating target degree detection class algorithms for FP_GA hardware should be to analyze the requirements of the target degree detection task, i.e., identifying objects and their application scenarios.

We focus on four types of sensory data: video surveillance, sound, sensors, and social media. These data sources are diverse and include public cameras, environmental monitoring stations, social media platforms, etc. They provide real-time, comprehensive information for city management and help in accurate decision-making.

Next, the target degree detection class algorithm should be optimized by software optimization. According to the characteristics of the optimized target degree detection class algorithm, the hardware structure is customized. Finally, the performance of the whole hardware speeder should be evaluated. The hardware acceleration of the target degree detection class algorithm based on FP_GA can be started from software optimization and hardware optimization[24-25].

To port YOLO-v3_means++ to FP_GA[26-29], it is necessary to reasonably utilize the logic resources of FP_GA. The data format of YOLO-v3_means++ is a single degree of correctness floating-point number, and in the process of advancing and advancing, because of the multiply-add operation involving floating-point numbers, the more the number of bits is, the more the logic resources are consumed, which is not conducive to the porting of the algorithm. Therefore, it is necessary to consume as little resources as possible while ensuring the correct degree. Moreover, by optimizing the network structure as well as post-processing, the process of advancing and promoting YOLO-v3_means++ can also be accelerated. This chapter first introduces the principle of YOLO-v3_means++ target degree detection network and the network structure performs the half-correct degree floating-point quantization of the network model, fuses the convolutional degree level and the BN layer in the network to accelerate the advancement and push forward, and reduces the complexity of the post-processing algorithm by using the C-FENCE algorithm. The YOLO-v3_means++ algorithm is combined with the FP-GA's The principle of composition and its network design idea is shown in Figure 2.1.

In the post-processing stage of target degree detection, YOLO-v3_means++ uses a non-maximal value suppression operation to get the best target frame. For the different kinds of detected target frames are sorted

Table 2.1: Comparison of cutting algorithm results

mould
Input: B= {b1..... bN},S= {S1.... SN}
1: D-02: while B is empty do
3: m←argmaxS
4: M ←bm
5: drum: bb-m
6: for bi in B do
7: if you(M, bi) > Nt then
8: B←B-bi S←S-Si
9: end10: end
11: end
12: return D, S end

from high to low by the confidence degree, respectively, to get a target frame with the maximum confidence degree, assuming that the limit value of 1OU (intersection and concatenation ratio) is 0.5 at this time, and calculating the 10U of the target frame with high confidence degree and the rest of the frames, if it is greater than the limit value, then it is determined that at this time the two target frames are recognized to be the same target, and the target frame is deleted; if it is smaller than the limit value, then it is determined that at this time the two target boxes do not belong to the same target. Repeat the process for the remaining target frames until all target frames are processed. The pseudo-code of the N-MS algorithm is shown in Table 2.1 of the algorithm entry.

Since the model detection interval range boxes have different sizes and positions, the directly calculated M_H_D distance does not have a unified metric, and therefore is not comparable, so it is necessary to unify the coordinates before calculating the M_H_D distance, transforming the coordinates between 0 and 1. When evaluating the performance of multi-objective evolutionary algorithms, we focus on metrics such as accuracy, efficiency, robustness, scalability and diversity. These metrics comprehensively assess the accuracy and robustness of the algorithms, which are crucial for their practical application in smart city design[30−31]. The specific process is as follows:

$$\begin{gathered} X=\{x_1,x_2,p_1,p_2\} \\ Y=\{y_1,y_2,q_1,q_2\} \\ Nx_i=\frac{x_i-\min(X)}{\max(X)-\min(X)} \\ Ny_i=\frac{y_i-\min(Y)}{\max(Y)-\min(Y)} \\ NP_{(u,v,m,n)}=|Nx_1-Np_1|+|Nx_2-Np_2|+|Ny_1-Nq_1|+|Ny_2-Nq_2| \\ WP_{(u,v,m,n)}=\frac{NP_{(u,v,m,n)}}{c} \end{gathered} \tag{2.1}$$

After the unification operation of all coordinate pairs, the coordinate points have values between 0 and 1, and the M_H_D distance of any pair of intersecting bounding-degree index frames is less than 2. Therefore, if the P-value of any two bounding-degree index frames is less than 2, it can be determined that they belong to the same group, referring to the same object, or one or more high-density pairs. C-FENCE obtains optimally weighted proximity of the detection interval range frames by using the M_H_D distance derived after the unification of coordinates to divide by its confidence score, as shown in Figure. 2 until it has processed all the pairs. H_D distance obtained by dividing the confidence score by the M_D distance obtained after the harmonization of coordinates, to obtain the weighted proximity of the detection interval range frame, and recursively repeat the process, as shown in Figure 2.2, until all the boundedness index frames are processed, and the optimal target index frame is obtained.

YOLO-v3 is a target degree detection class algorithm improved and enhanced based on YOLO_v2, the basic idea of YOLO-v3 can be divided into two parts, firstly, a series of candidate frames are generated on the input image according to a certain rule, which are the regions that may contain the target, and they are

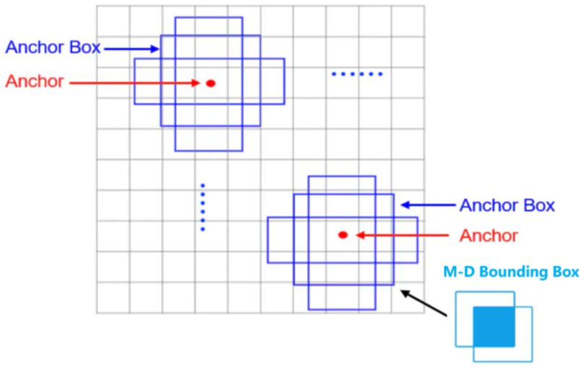

Fig. 2.2: Schematic of the MD Boundary Recognition Frame

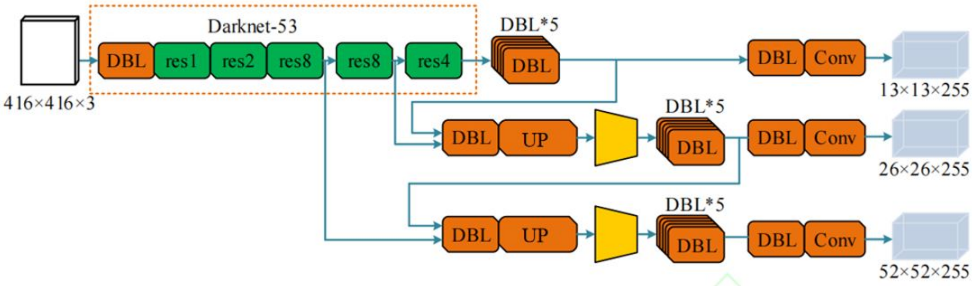

Fig. 2.3: Schematic of the MD Boundary Recognition Frame

categorized into positive samples and negative samples by annotating them with the real frames, the positive samples are those candidate frames that are completely overlapped with the real frames, while the negative samples are those candidate frames that have some deviation from the real frames [1]. overlap with the real frame, while the negative samples are the candidate frames that have some deviation from the real frame [1]. Secondly, the K-mean++ convolutional network is used to extract features from the candidate regions and perform location localization and type recognition, and the candidate regions are inputted into the convolutional neural architecture network to obtain the feature representation related to the target. These detection results are compared with the labels of the real frames to determine whether the target is correctly detected.YOLO-v3 uses Dark-Net-53 instead of Dark-Net-19 in YOLO-v3_means++ for the feature indexing network.Dark-Net-53 is a fully convolutional network structure that consists of multiple convolutional degree levels of 1? and 3? layers, each of which is followed by a batch uniformization layer and an activation layer. Unlike YOLO-v3_means++, the Dark-Net-53 network does not have pooling and fully connected layers but performs the down sampling operation by convolution with step size 2. After 5 down samplings, the size of the feature pattern map is reduced to 1/32 of the original image. The Dark-Net-53 network also introduces a residual group block structure with shortcut connections between the convolutional degree levels. This structure effectively reduces the difficulty of the trained deep class network and enables the network to converge better. The network structure is shown in Figure 2.3.

Assuming that an image undergoes the YOLO-v3_means++ network model to generate n prediction interval range frames, both the N-MS algorithm and the C-FENCE algorithm need to store n detection interval range frames and their corresponding parameter information, so the space complexity is the same. (n), the

Table 2.2: Algorithm under detection interval range box parameter prediction

Moulid

Input.
while B+empty dobs, ss←0,0
optimal C-FENCEIp for bi, s in B, S do C-FENCE by+0
except b= bi nb,nb,←normalize(b,b)
P← proximity(nb,nb)
If <2 then
C-FENCE ↔ C-FENCE U proximity
End if
If C-FENCE < optimal
C-FENCE then
optimal C-FENCE ←C-FENCE

specific prediction code program is shown in Table 2.2. For the N-MS algorithm, the degree of time-course complexity is as follows:

1. Traverse the confidence level of n prediction interval range frames with a time-range composite level of 0(n);
2. Sort the prediction interval range boxes in ascending order of confidence with a time-range composite degree of O(nlogn);
3. Select the prediction interval range box with the highest level of confidence and place it in the results with a time-range composite degree of 0(1);
4. Calculate the area of overlap between the remaining predicted interval range frames and the predicted interval range frame with the highest confidence level, and delete those whose overlap is greater than a set limit value, with a time-range composite degree of O(n);
5. Repeat steps 3 and 4 until all prediction interval range boxes have been processed with an algorithmic complexity of O(n2).

3. Optimization of Y-K-means++ fusion algorithm under multi-objective. FP_GA-based YOLO-v3_means++ algorithm hardware speeder model YOLO-v3_means++ The network model has a total of 22 layers. In the actual operation, the input of the current layer comes from the output of the previous layer, so the network is operated layer by layer. In which to realize multi-dimensional perception, we use a multi-objective evolutionary algorithm as shown in Figure 3.1. In which to reduce the computation time in the YOLO-v3 network, the study employs a depth-separable convolution method to improve the residual group block structure. The K-means++ algorithm improves the quality and stability of clustering by optimizing the initial center of mass selection. When generating the Anchor_Box, the algorithm ensures a reasonable configuration of Anchor points to better accommodate targets of different sizes and shapes. This not only improves the accuracy and efficiency of target detection, but also enhances the adaptability and flexibility of the algorithm. Therefore, the K-means++ algorithm plays a key role in generating the Anchor_Box, which provides an effective method for solving the multidimensional sensory data prediction and resource scheduling problems in smart cities. This method reduces the computational complexity by reducing the number of parameters in the convolution operation, and the study introduces the K-means++ algorithm in the residual group block [29-31]. Such a structure is effective in reducing the computational volume of the model, but also able to extract more information about the feature pattern of the target, which improves the degree of detection correctness, and through this optimization, it is possible to increase the speed of the algorithm's fulfillment procedure while maintaining the degree of accuracy.In optimizing the YOLO_v3 network model, we set the following criteria: to improve the accuracy and efficiency of target detection, as well as to enhance the model's adaptability and flexibility to different data.

After using the K-means algorithm and K-means++ algorithm to classify the data in aggregated type, the study can show their aggregated classification results by plotting a two-dimensional coordinate graph. In the

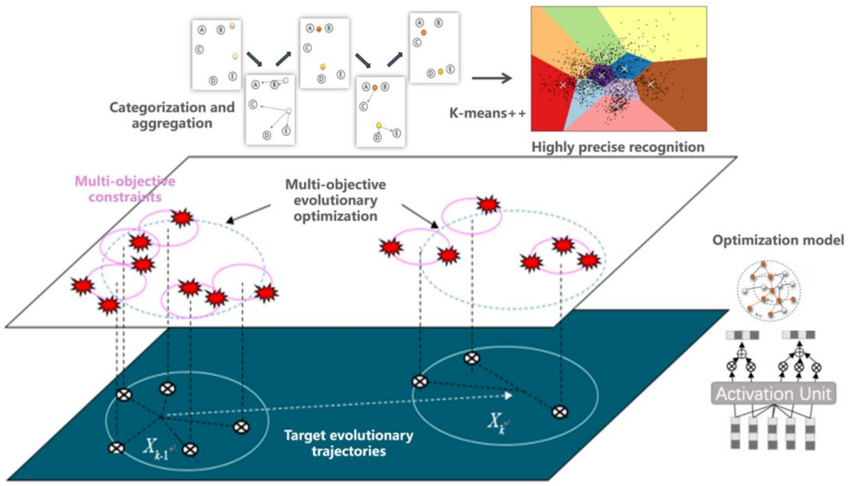

Fig. 3.1: Specific process of joint evolutionary optimization algorithm under multi-objective

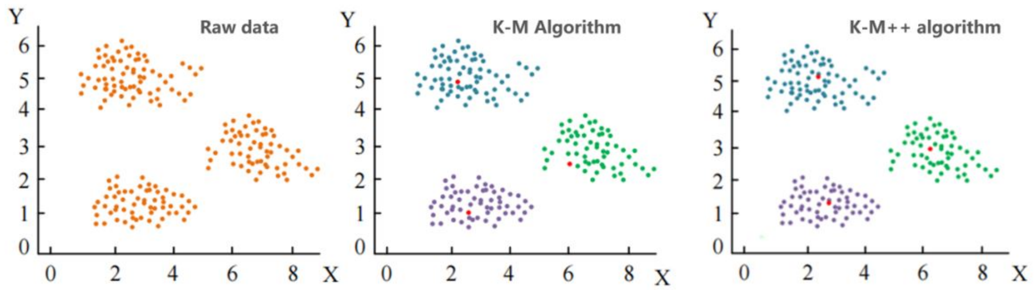

Fig. 3.2: Comparison of the degree of accuracy of different algorithms for aggregated classification under multi-objective

graph, each data point represents a sample object, while different colors indicate different aggregated classification groups. A comparison of the degree of accuracy of the two algorithms' aggregation type classification is shown in Figure 5.

According to Figure 3.2, it can be found that under the K-means++ algorithm, the aggregation-type classification results demonstrate a higher degree of recognition accuracy. This is because the K-means++ algorithm is more ingenious in selecting the initial aggregation-type classification centers, which effectively avoids the influence of the initial aggregation-type classification centers on the aggregation-type classification results by setting the initial aggregation-type classification centers farther apart from each other. As a result, the K-means++ algorithm can better capture the intrinsic structure of the data, making the aggregation type classification results more accurate.

The width placement, height, and area of Anchor_Box generated by the K-means algorithm and K-means++ algorithm are compared when the number of aggregated classification centers is 9. Details are shown in Table 2.1. After assigning the aggregated classification results to the 3 feature pattern map group layers, the differences of Anchor_Box in each size feature pattern map are compared. It can be observed that the Anchor_Box generated by the K-means++ algorithm has more differences between the different sizes of feature pattern map group layers.

Table 3.1: Differences in Algorithmic Aggregate Classification

box-shaped	Feature size	32*32	16*16	8*8
average width	K-m	33	53	127
	K-m++	29	85	118
average height	K-m	48	114	126
	K-m++	49	107	131
average size	K-m	1417	5364	15987
	K-m++	1444	5892	17062

According to Table 3.1, it can be seen that the Anchor_Box generated by the K-means++ algorithm has a greater difference between the layers of different sizes of feature pattern mapping groups. This means that the Anchor_Box generated by the K-means++ algorithm has a better degree of adaptation and flexibility in detecting the degree of targets of different sizes. In contrast, the K-means algorithm generates Anchor_Box with relatively small differences between different sizes of feature pattern map group layers. This is because targets of different sizes may require different sizes of Anchor_Box for accurate detection. By using the Anchor_Box generated by the K-means++ algorithm, it can better meet the detection needs of targets of different sizes, thus improving the performance and accuracy of the target degree detection class of algorithms. Thus, the K-means++ algorithm plays a key role in optimization in YOLO-v3.

In evaluating the performance of the YOLO-v3 algorithm for urban traffic detection, we delve into the importance of the P-R curve.The P-R curve, or precision-recall curve, is a key tool for measuring the performance of target detection algorithms. By plotting the P-R curve, we can visually compare the performance of different algorithms in dealing with various types of losses.The larger the area under the P-R curve, the better the detection ability of the algorithm. This characteristic provides us with a quantitative criterion for evaluating the performance difference between different algorithms when processing urban traffic data. On the NEU-DET dataset, we plotted the P-R curves of different algorithms for detailed comparison.It can be clearly seen that there is a significant difference in the area underneath the three curves. The largest area is the third curve, which represents the optimized algorithm showing strong performance in urban model integrity detection. In contrast, the area under the second curve is the smallest, indicating a relatively weak performance.Further analysis reveals that the loss type represented by the third curve achieves an optimal balance between precision and recall. This means that the optimized algorithm can achieve a high recall while maintaining high precision. This property is particularly important in urban traffic detection, as we want the algorithm to detect as many targets as possible while reducing false detections.

In contrast, the loss type represented by the second curve performs poorly at detection, with low precision, even at high recall. This may be due to the algorithm's difficulty in effectively identifying and classifying different types of information when dealing with urban traffic data. By analyzing the P-R curve in detail, we can clearly see the advantages of the optimized algorithm in urban traffic detection. This analysis not only helps us to understand the performance differences of different algorithms, but also provides directions for further improvement and optimization of the algorithms. The multi-objective evolutionary algorithm shows strong adaptability and robustness in dealing with complex and variable urban traffic data, which provides strong support for promoting the development of smart cities.

4. Pilot test results. Through the previous joint method, we can realize the application of multidimensional sensory data prediction and resource scheduling under various domains of smart cities. In this section, the application analysis in practice will be carried out, the processor used for model training is: Intel_Xeon_Gold_5218_CPU, the memory is 6, the kernel value is 8 cores, the graphics card is NVIDIA_DV_RTX 2080 Ti, the operating system is Windows 10_64-bit, and this training is based on the deep-type learning framework pytorch_1.7 The experimental environment is python_3.7, and the GPU acceleration software is C_UDA_10.2 and CUDNN_7. Considering that the YOLO-type algorithm itself uses the VOC dataset, the urban traffic detection dataset is constructed according to the format of the VOC dataset, and the part of

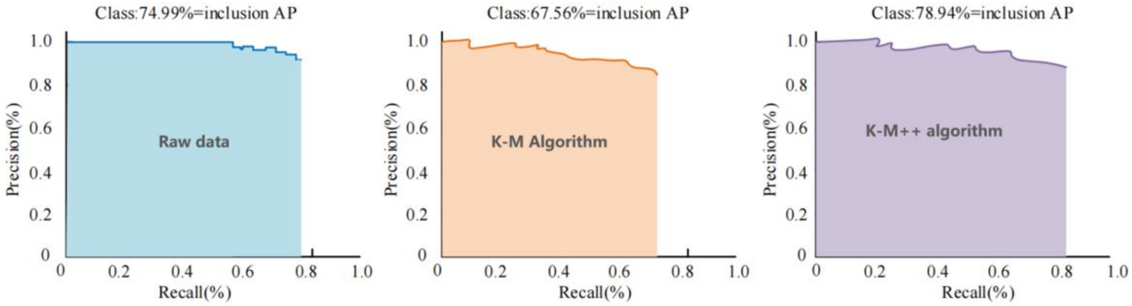

Fig. 3.3: Comparison of the degree of accuracy of different algorithms for aggregated classification under multi-objective

Fig. 4.1: Illustration of test applications such as city maps, city groundwater, city highway traffic, etc.

the image in the dataset is the schematic diagram of the practical application results as follows, which shows that we can see that the city maps, city groundwater, urban highway traffic, urban building layout, urban power layout, and other test applications can be seen. Through the optimized monitoring results (as shown in Figure 4.1), it can be seen that the ability to test the results of this paper, can be a very clear reflection of the city where we need certain detection goals.

To improve the advancing and pushing speed of the YOLO-v3_means++ network model, and to verify the feasibility of these two methods. As shown in Figure 4.2, it can be seen from the figure that the detection time of the original YOLO-v3_means++ network model for a single-frame image on FP_GA is about 2.2-2.8s, and the average detection time is about 2.4s. After the fusion of the convolutional degree level with the BN layer, the average detection time for a single-frame image is reduced to 2.25 s. While the average detection time for a single-frame image is about 2.01 s when C-FENCE is used as the post-processing algorithm, it appears that some of the detection time exceeds that of the original YOLO-v3_means++ network model. This is because when there is a partial overlap between two targets of the same class, the N-MS algorithm will directly filter

Fig. 4.2: Schematic of test applications for city maps, urban groundwater, urban highway traffic, etc.

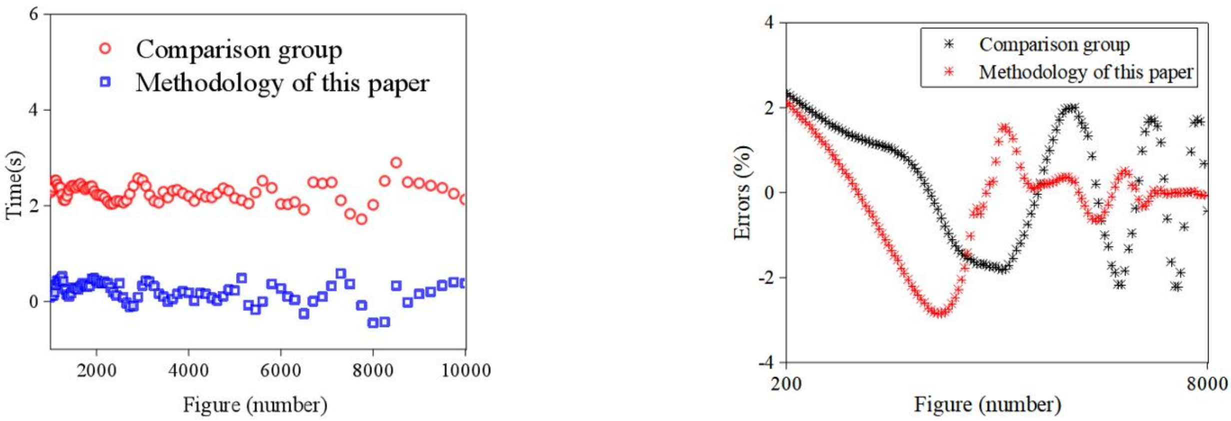

Fig. 4.3: The success rate of each algorithm's operation

out the target frame with less confidence in the same class of targets, while the C-FENCE algorithm adopts an aggregated classification to avoid leakage detection, but increases the processing time. Secondly, it is also necessary to analyze and compare the detection results of GPU and FPGA. To present the detection results more intuitively, this paper draws the target frames predicted by the network and labels the target categories with their confidence levels through the Open_CV tool. Figure 4.3 shows the comparison of the two sets of detection results for two different platforms. By comparing and analyzing the detection results in Figure 4.3, it can be seen that in terms of the accuracy of the target frame and the confidence level of the target, the detection results of GPU are slightly better than those of FPGA. This is because, in the process of porting the YOLO-v3_means++ network, this paper utilizes the half-correctness floating-point number to quantize the YOLO-v3_means++ network, which results in part of the loss of correctness, although the loss of correctness is more than the loss of correctness. degree loss, although the loss of the correct degree is less, the error will still be reflected in the final detection result as the computation volume increases. In the detection results of the FP_GA platform using the C-FENCE method as the post-processing algorithm in Figure 4.3, although the target confidence level is also slightly lower than that of the GPU platform, the target frame is more in line with the target model itself, which is because the C-FENCE algorithm considers the target's confidence level and weighted proximity while calculating the target frame instead of using the confidence level as the judging criterion only. The experimental results show that the FP_GA-based, YOLO-v3_means++ network can better meet the detection needs.

5. Conclusions and discussions. The paper focuses on the application of multi-objective evolutionary type algorithms for multi-dimensional sensory data prediction and resource scheduling in smart city design.

The conclusions of the related simulations and tests are as follows:

1. YOLO_v3 network model jointly with the K-means++ algorithm to generate Anchor_Box, the new algorithm has greater differences between different sizes of feature pattern map group layers, which indicates that the algorithm has a better degree of adaptability and flexibility to better meet the needs of detection of different sizes of targets and improve the performance and accuracy of the target degree detection class of algorithms.
2. To further analyze the performance of YOLO-v3 joint algorithms for urban traffic detection, the P-R curves of the various types of algorithms for some of the loss types on the NEU-DET dataset are plotted. The optimized algorithms are found to be stronger for detecting different types of losses in urban model completeness detection.

Practical application analysis revealed that the optimized monitoring results were compared with the detection time of the original YOLO-v3_means++ network model on FP_GA. We found that the average single-frame image detection time is reduced to 2.25 s after fusion of the convolutional degree level with the BN layer, while the average single-frame image detection time is about 2.01 s when using C-FENCE as a post-processing algorithm.

The results of the tested recognition applications show that this fusion algorithm based on FP_GA's YOLO-v3_means++ network can better meet the detection needs and that optimizing the YOLO-v3 network model using a multi-objective evolutionary algorithm can improve the performance and accuracy of the target degree detection class of algorithms.This study provides valuable insights for smart city planners and designers.Future research could be further extended to different types of sensory data, such as radar, infrared, etc., to provide a more comprehensive perception of the city. Meanwhile, changes and improvements in evolutionary algorithms, such as genetic algorithms and particle swarm optimization, are explored to find more efficient and accurate solutions. In addition, research could focus on the combination of multi-objective evolutionary algorithms with other advanced technologies, such as deep learning, reinforcement learning, etc., in order to promote the sustainable development of smart cities.

REFERENCES

[1] Zhenqiang, L. Watershed governance and reconstruction under the perspective of transformation of resource-oriented cities: An example of Pingdingshan Jiaodian wisdom urban landscape and water conservancy project. *China Municipal Engineering.* **2022**, 33-35
[2] Zichen, Z. Analyzing the utilization practice of smart city concept in urban landscape design. *Modern Horticulture.* **2021**, 128-129
[3] Zou, Z., Chen, K., Shi, Z. & Others Object detection in 20 years: a survey. *Proceedings Of The IEEE.* (2023)
[4] Liu, F., Guo, M. & Xiangjun, W. Small target degree detection of convolutional neural architecture networks based on cross-scale fusion. *Advances In Lasers And Optoelectronics.* **58**, 213-221 (2021)
[5] Zhengjun, S. Design and realization of vehicle counting system based on YOLO-v3 [D]. (Northern Nationalities University,2019)
[6] Girshick, R., Donahue, J., Darrell, T. & Others R-C-NN. *Region-based.* pp. 2-9 (2014)
[7] Girshick, R. Fast r-C-NN[C]//Proceedings of the IEEE international conference on computer Vision.
[8] Ren, S., He, K., Girshick, R. & Others Faster r-C-NN: Towards real-time object detection with region proposal networks. *Advances In Neural Information Processing Systems.* **28** (2015)
[9] Shao, Y., Zhang, D., Chu, H. & Others A review of YOLO object detection based on deep learning. *Journal Of Electronics And Information.* **44** pp. 1-12 (2022)
[10] Hinton, G., Krizhevsky, A. & Sutskever, I. Imagenet classification with deep convolutional neural networks. *Advances In Neural Information Processing Systems.* **25** pp. 1106-1114 (2012)
[11] Liu, Y. & War, Y. A review of deep learning-based algorithms for small target degree detection class [D]. *Computer Engineering And Applications.* **57**, 37-48 (2021)
[12] Viola, P. & Jones, M. Robust real-time face detection. *International Journal Of Computer Vision.* **57** pp. 137-154 (2004)
[13] Hongpeng, Y., Bo, C., Yi, C. & Zhaodong, L. A review of vision-based target degree detection and tracking. *Journal Of Automation.* **42**, 1466-1489 (2016)
[14] Felzenszwalb, P., Girshick, R., McAllester, D. & Others Object detection with discriminatively trained part-based models. *IEEE Transactions On Pattern Analysis And Machine Intelligence.* **32**, 1627-1645 (2009)
[15] He, K., Zhang, X., Ren, S. & Others Spatial pyramid pooling in deep convolutional networks for Visual recognition. *IEEE Transactions On Pattern Analysis And Machine Intelligence.* **37**, 1904-1916 (2015)
[16] Liu, W., Anguelov, D., Erhan, D. & Others SSD: Single shot multibox detector[C]//Computer Vision-ECCV 2016: Ih European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part I 14. (Springer International

Publishing,2016)

[17] Mingsong, C., Zegong, Z., Ranran, W. & Yongrong, W. Research on Vehicle Behavior Decision Making Based on Deep Reinforcement Learning Algorithm []] Journal of Guilin University of Electronic Science and Technology, 2022, 42(01):29-35.

[18] Tang, Y., Zhang, X., Zhou, P. & Others E-F-Tra: Enable efficient on-device C-NN training on FPGA through data reshaping for online adaptation or personalization. *ACM Transactions On Design Automation Of Electronic Systems (TODAES)*. **27**, 1-36 (2022)

[19] Basalama, S., Sohrabizadeh, A., Wang, J. & Others FlexC-NN: An End-to-End Framework for Composing C-NN Accelerators on FPGA. *ACM Transactions On Reconfigurable Technology And Systems*. **16**, 1-32 (2023)

[20] Sait, S., El-Maleh, A., Altakrouri, M. & Others Optimization of FPGA-based C-NN accelerators using metaheuristics. *The Journal Of Supercomputing*. **79**, 4493-4533 (2023)

[21] Huang, Y., Shen, J., Wang, Z. & Others A high-efficiency FPGA-based accelerator for convolutional neural networks using Winograd algorithm[C]//Journal of Physics: Conference Series. *Iop Publishing*. **1026**, 12-019 (2018)

[22] Shi, F., Li, H., Gao, Y. & Others Sparse winograd convolutional neural networks on small-scale systolic arrays. (2018), arXiV preprint

[23] Nguyen, D., Nguyen, T., Kim, H. & Others A High-Throughput and Power-Efficient FPGA Implementation of YOLO C-NN for Object Detection. *IEEE Transactions On Very LargeScale Integration (VLSI) Systems*. pp. 1-13 (2019)

[24] Lian, X., Liu, Z., Song, Z. & Others High-Performance FPGA-Based C-NNAccelerator With Block-Floating-Point Arithmetic. *IEEE Transactions On Very LargeScale Integration (VLSI) Systems*. pp. 1874-1885 (2019)

[25] Alwani, M., Chen, H., Ferdman, M. & Others Fused-layer C-NN accelerators[C]//: The 49th Annual IEEE/ACM International Symposium on Microarchitecture. *IEEE Press*. **22** (2016)

[26] Courbariaux, M., Hubara, I., Soudry, D. & Others Binarized neural networks: training deep neural networks with weights and activations constrained to +1 or -1[J]. (2016), http://arxiV, org/abs/Learning

[27] Venieris, S. & Bouganis, C. Latency-driven design for FPGA-based convolutional neural networks [C]//2017 27th International Conference on Field Programmable Logic and Applications. *FPL)*. pp. 1-8 (2017)

[28] Xinshuai, L. & Yang, G. Research on graduate students' performance portrait in colleges and universities based on improved K-means algorithm. *Shaanxi Education (Higher Education)*. **2023**, 49-51

[29] Yun, D. Research on the application of K-means aggregated classification algorithm for network security monitoring based on big data. *Information Record Material*. **24**, 140-142 (2023)

[30] Zhao, C. Research on bridge crack identification based on improved K-means aggregated classification segmentation method. *Heilongjiang Transportation Science And Technology*. **46**, 122-124 (2023)

[31] Ma, J., Luo, X., Wang, Y. & Others Research on weed detection based on improved YOLO-v3. *Internet Of Things Technology*. **13**, 11-16 (2023)

Edited by: Zhengyi Chai
Special issue on: Data-Driven Optimization Algorithms for Sustainable and Smart City
Received: Nov 29, 2023
Accepted: Apr 25, 2024