DOI: 10.2478/eces-2023-0029 ECOL CHEM ENG S. 2023;30(2):275-282

Weijie HU[1*], Youfei ZHOU[1], Junying LU[1], Jun SHENG[1] and Zechen JIN[1]

REMOTE SENSING DETECTION AND RESOURCE UTILISATION OF URBAN SEWAGE SLUDGE BASED ON MOBILE EDGE COMPUTING

Abstract: Improper disposal of municipal sewage sludge poses a significant threat to effective environmental protection. With the continuous advancement of artificial intelligence technology and the Internet of Things (IoT), remote sensing detection technology is emerging as a promising research avenue to address this issue. However, the current state of real-time detection technology is inadequate, hindering comprehensive and stable monitoring operation. Additionally, the rational use of network resources remains suboptimal. To address this challenge, this study proposes a resource optimisation technology for the current insufficient intelligent monitoring system of urban sewage sludge. By leveraging IoT and wireless technology, water meter data can be collected with minimal earth construction compared to traditional PLC collection. This is followed by utilising Faster R-CNN to plan the network transmission of sewage remote sensing information resources. Finally, the architecture collection module's scalability is enhanced by incorporating edge computing and reserving sensor ports to meet future plant expansion demands. The experiment demonstrates the significant potential of this technology in application and resource optimisation. In actual parameter tracking tests, the proposed method effectively monitors sewage sludge, providing policy guidance and measure optimisation for relevant authorities, ultimately contributing to pollution-free urban development.

Keywords: wastewater, edge computing, faster R-CNN, resourceful management

Introduction

In recent years, a substantial quantity of sludge treatment has exerted immense pressure on urban environmental protection worldwide, due to the expansion of urban sewage treatment capacity. Presently, the global urban sludge recycling industry primarily employs harmless treatment technologies, it is noteworthy that wastewater treatment facilities must allocate 20 % to 50 % more funds to appropriately dispose of the sludge, thereby representing a substantial cost. However, following safe wastewater treatment, municipal sludge can be viewed as a renewable resource with extensive utilisation potential. The application and innovation of green technology has received extensive attention [1, 2].

Edge computing leverages proximity to provide users with an enhanced experience [3]. Although local high-performance devices exhibit high-speed capabilities in image recognition, the lack of device performance often limits their ability to perform

[1] Design Institute No. 3, Shanghai Municipal Engineering Design and Research Institute (Group) Co., Ltd., Shanghai 200092, China, ORCID: WH 0009-0001-9726-3389, YZ 0009-0000-8034-5129, JL 0009-0000-4622-7524, JS 0009-0006-5641-0924, ZJ 0009-0008-5904-7918

[*] Corresponding author: huweijie_888@163.com

high-parametric neural network computations. In the absence of clear criteria for resource allocation and utilisation, the mobile edge computing technology has emerged as a viable solution for reasonably allocating remote sensing signals and resources from an information technology perspective. By storing image processing resources in a distributed fashion, mobile edge computing rationally calculates and distributes computing resources, designs the results and detection technology of incoming images, and feeds back to relevant departments in image format [4].

In this study, we present an innovative approach that promises to significantly enhance the utilisation rate of social resources and offer pragmatic recommendations for managing municipal sewage sludge. The principal contributions are outlined as follows: We propose a resource allocation algorithm, grounded in *Region-based Convolutional Network* (Faster-R-CNN), for remote sensing image allocation of municipal sewage sludge treatment. This algorithm rationalises the allocation of data processing for remote sensing image detection technology and leverages shared convolution features to vastly augment the pace of target detection. We apply edge computing technology to smart equipment components to integrate isolated equipment information and improve the model's pruning effect through the deployment of a static migration scheme.

Related works

Remote sensing monitoring

CNN can learn more information on the raw spectral data with higher accuracy. Luo et al. [5] demonstrated that 1D-CNN is capable of learning more information on raw spectral data with higher accuracy. Their experiments showed that the 1D-CNN model is less sensitive to noise than traditional methods, making it more suitable for analysing strong noise Raman spectral signals in complex field measurements. Moreover, CNN eliminates the need for complex process and artificial feature extraction, making the analysis process simpler and the results more accurate [6].

SPP-Net, as proposed in [7], replaces the final pooling layer of the conventional convolutional neural network with Spatial Pyramid Pooling (SPP). Fast R-CNN, on the other hand, substitutes the final pooling layer of the traditional CNN model with a Region of Interest (RoI) pooling layer. In addition, border regression is introduced to the network, and feature sharing is employed to reduce repetitive computation, resulting in a significant improvement in the speed of target recognition [8]. Despite these advances, Fast-RCNN still relies on Selective Search to extract regions of interest, rendering it unsuitable for real-time processing. Moreover, both R-CNN and Fast R-CNN necessitate multiple steps during training and testing, including candidate frame extraction, feature extraction, SVM, and SVR, all of which are independent processes that require intermediate results to be saved separately. This results in high space and computational complexity. Fortunately, Faster R-CNN integrates candidate frame extraction and convolutional neural network classification, thus enhancing the recognition accuracy and computational efficiency [9].

Edge computing

Edge computing, as designed in [10], aims to transfer computing power from centralised cloud computing servers to edge nodes in close proximity to the user end. This shift yields two significant improvements to existing cloud computing models [11]. Firstly, edge nodes can pre-process large data volumes before transmitting them to a central server in the cloud. Secondly, they can reduce the burden on cloud servers by leveraging computational power at the edge.

However, the practical deployment of deep convolutional neural network algorithms in edge computing devices is largely hindered by three factors [12]. Firstly, there is a limitation on the size of network weight parameters. The impressive detection capability of convolutional neural networks relies on millions of weight parameters that require a significant amount of storage and operation space. Secondly, there is a limitation on the memory required during runtime. During inference operations, the convolutional neural network's computational operations require significant memory allocation. Finally, there is a limitation on the number of computational operations. Convolutional operations on high-resolution images can be very computationally intensive, and a convolutional neural network with substantial depth may take several minutes to process an image on an edge device. Consequently, real-time applications cannot meet high real-time requirements.

System design

Overall structure

The edge server selects tasks based on configuration and task information, taking preferences into account. The management server is responsible for summarising all current choices and providing feedback according to the Quality of Service (QoS) of the MEC system.

Additionally, we acknowledge that remote sensing image processing entails a sequence of operations such as radiometric correction, geometric correction, image enhancement, projection transformation, mosaic, feature extraction, classification, and various thematic processing of remote sensing images to achieve the desired goal [13].

Water meter information collection based on IoT

Sensors are the principal means of acquiring information in both natural and industrial settings. Plant equipment is outfitted with sensors and A/D converters so that computer systems can capture equipment status information. This allows for equipment analysis based on the data collected. Therefore, the development of modern intelligent equipment requires a vast number of highly functional and high-quality sensors. The sensors, along with A/D converters, serve as core components of the data acquisition module, enabling seamless data acquisition.

Furthermore, when using the data acquisition module, it is crucial to pay attention to the sampling frequency. If the highest frequency of the equipment signal is denoted as f, then, according to Shannon's sampling theorem, the sampling frequency must be set to $2f$ to ensure optimal signal collection [14]. The lower computer data acquisition board uses DMA (Direct Memory Access) to send and receive data, thus making the high-frequency signal collection points more and collecting a large amount of data.

Typically, the system would send the acquired data directly to the host computer, which could result in data loss and errors, thereby negatively impacting subsequent research and analysis. To mitigate this problem, the acquired data is first transferred to a buffer pool or memory space, which is reserved in advance in the module specifically for data storage. This ensures the integrity of the data to a certain extent and enables the detection of the size of the transmitted data in the memory space, as well as the monitoring of whether the output data is consistent with the input.

Faster R-CNN-based remote sensing monitoring of wastewater

Faster R-CNN employs the Region Proposal Network (RPN) as a substitute for the Selective Search approach found in previous R-CNN and Fast-R-CNN models. This technique facilitates sharing of convolutional features across the entire map with the target detection network, thereby dramatically lowering computational complexity. RPN is a fully convolutional network, which can accurately forecast both the target candidate frame and the target score of the input image. By training RPN end-to-end, it generates high-quality region candidate frames.

Simultaneously predict k = 9 candidate boxes for each 3 × 3 size sliding window. The position regression layer outputs 4k position coordinates of k candidate boxes, that is, coordinate codes of k candidate boxes. The classification layer outputs 2k target scores for the candidate boxes, i.e. the estimated probability that k candidate boxes are targets/non-targets.

Edge gateway design

The utilisation of edge computing technology in smart equipment components enables the integration of information from previously isolated devices. Latency is divided into computational latency and communication latency. T_m represents the computation latency of the edge server, and the communication latency is represented by T_s as shown in Equation (1) and Equation (2):

$$T_m = \frac{F_l}{f_s} \tag{1}$$

$$T_s = \frac{P_{size}}{u_s} + \frac{P_{result}}{d_s} \tag{2}$$

where F_l denotes the total number of floating points required by the mobile CPU to complete the computational task, and is a fixed parameter of the wheat fertility monitoring model, P_{size} and P_{result} denotes the size of uploaded data and received data. Energy consumption can be divided into two categories: computational energy consumption and communication energy consumption. In this context, the decision is solely based on the energy consumption of the mobile device, thus only accounting for the energy consumption attributed to the mobile device.

This paper employs compression rate and floating point operations to evaluate the effect of model pruning [15]. The compression rate indicates the local storage efficiency of the material prediction model, whereas floating point operation demonstrates the acceleration effect. The computation of the parameter quantity of the original network model is as follows:

$$\|w^i\| = N_i \times N_{i+1} \times K_i \times K_i + N_{i+1} \tag{3}$$

where K_i denotes the kernel_high and kernel_wight of the *i-th* kernel_high and kernel_wight of the convolutional kernel, the fully connected layer is equivalent to $K_i = 1$ of the convolution kernel, the compression rate of the model parameters before and after pruning P is calculated as:

$$P = \frac{\sum_{i=1}^{L} |w^i|}{\sum_{i=1}^{L} |w*^i|} = \frac{\sum_{i=1}^{L} N_i \times N_{i+1} \times K_i \times K_i}{\sum_{i=1}^{L} N_i{}' \times N_{i+1}{}' \times K_i \times K_i} \tag{4}$$

Floating-point operations are used as a criterion to evaluate the computational complexity of the model. The floating-point operations are shown in Equation (5), which mainly include addition and multiplication, where the fully connected layer can be considered as $W_i = K_i = H_i = 1$.

$$FPs = \sum_{i=1}^{L} (N_i \times K_i^2 + 1) \times N_{i+1} \times H_{i+1} \times W_{i+1} \tag{5}$$

Experiment and analysis

Experimental setup

Utilising the present data of a reclaimed water plant, which is stored within a relatively closed DCS database system, the archived historical data is queried. Wastewater treatment process data is recorded every hour, generating approximately 6400 samples. In this study, ten monitoring data of the plant's wastewater treatment process were selected for the construction of a water quality predictive control model. Prior to constructing the model, anomalous data values were identified and eliminated. Through the mean interpolation method, the small amount of missing data was filled in by replacing it with the average value of the data sample. In order to facilitate the training and optimisation of the model, the experimental data need to be normalised and converted into data between 0 and 1. Different input characteristic values under the same order of magnitude can effectively reduce the impact of the feature dimension on the model. The data set is divided into two parts, namely the training data set and the test data set. The distribution ratio between the training data set and the test data set is about 8 : 2.

Algorithm training

To leverage deep neural networks for data classification and processing within the resource-constrained embedded devices of the Internet of Things, this manuscript proffers a data mining methodology based on edge computing. The test plan is predominantly centred on this functional prerequisite. This segment outlines the design of the testing strategy. Firstly, branches are appended to the convolutional neural network, namely Alexnet and ResNet, and the neural network with the native architecture is classified on the same dataset to compare operational efficacy. Secondly, the edge computing scene is simulated to validate the distribution of the segmentation points of the convolutional neural network amidst edge and cloud, under variable conditions.

The paper conducts an initial investigation to compute the request density of ten users and adopts a dynamic allocation strategy for task requests. As shown in Figure 1, the positions generated in the dynamic planning of task requests are variable. The ordinate represents the request density and the abscissa represents the different locations

In these scenarios, the execution time tends to vary at the edge level. Typically, 90 % of the total resource capacity is allocated to the cloud level while only 10 % is reserved for the edge level. For smaller instances, the number of users typically ranges from 5 to 20. The number of VM instances for each resource type is then determined based on the demand from these users. To ensure that the available capacity for each resource type is proportional to the demand, the algorithm calculates the comparison of revenues. As the number of users increases, the algorithm is able to provide appropriate processing capabilities.

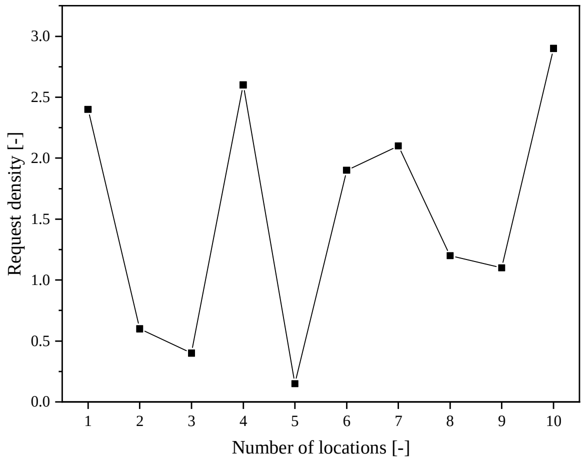

Fig. 1. Dynamic variation of random request density

System testing

The experiment aims to evaluate the impact of network transfer volume on the overall latency of the model recognition service. The total local time, network transmission time, and server processing time are recorded as a round of segmentation tests, and the CPU usage fluctuation of the edge server while running the model recognition service is recorded. The difference of CPU usage before and after running is used as a reference. The experiments are conducted under the same conditions, using the network in the same time period, and 10 experiments are conducted to ensure objectivity. The results obtained are presented in Figure 2 and Table 1.

Table 1

Model comparison results

Model	**1D-CNN**	**SVM**	**RF**	**Ours**
Delay	102.20	116.72	180.27	94.20
CPU growth	8.32	19.23	33.50	6.50
Number of split points	10	10	10	10

The recognition latency of the static migration deployment of the model is significantly reduced, achieving its lowest point at breakpoint 5. The CPU growth experiences a steady decline as the duration handed over to the server to run grows shorter with each preceding breakpoint. A comparison between the values of time and transfer volume demonstrates a strong correlation between the performance of the model's static migration deployment and the number of parameters transferred. The comparison of the latency results graph reveals that as the structural computation of the model shifts to the mobile side, the overall recognition latency not only fails to decrease but also experiences a slight increase. Thus, it can be inferred that the performance inequality between the server side and the mobile side, in addition to the amount of network transmission, also contributes to the recognition latency.

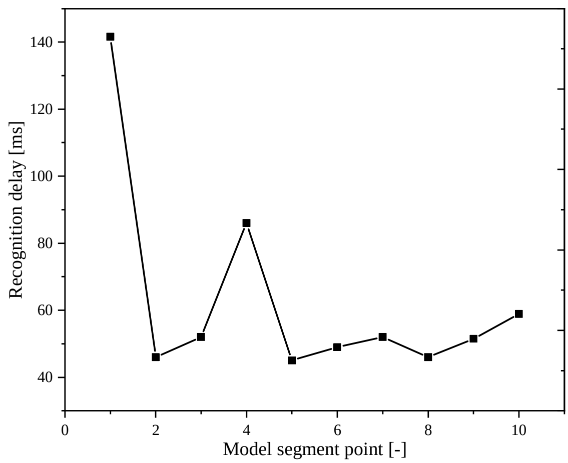

Fig. 2. Identification delay of model segmentation points

The abscissa of Figure 2 represents the model segmentation point and the ordinate represents the recognition delay. In comparison to the model's overall deployment to the server, the static migration scheme remarkably decreases recognition latency, alleviates server load pressure, and optimises the utilisation of both mobile and server computing resources. Nevertheless, in real-world scenarios such as wastewater monitoring, network speed conditions tend to fluctuate constantly. Consequently, the static migration scheme of the monitoring model may not provide timely responses to such changes in the environment. In the event of poor network speed conditions, the feature map of wastewater treatment recognition using the static migration scheme to the server may not achieve the desired effect, thereby reducing the scheme's robustness. As a result, the static migration scheme is best suited for models with a high parameter scale.

Conclusion

With the sustainable development of smart cities and the continuous progress of information technology, intelligent control of wastewater treatment has emerged as a highly effective method of managing wastewater treatment, and thus plays an increasingly pivotal role. In this paper, a model scheme was proposed that exhibited superior performance in practical testing. The scheme leverages both deep and shallow features, and proposes a target detection method oriented to scale change, enabling better prediction of effluent quality and operational control of the controlled variables, and ultimately facilitating predictive control of wastewater treatment. By filtering original data through edge computing to retain only useful data, and uploading the filtered data to the cloud computing centre for data analysis, not only can processing efficiency be improved, but bandwidth pressure can also be significantly reduced. The model can effectively reduce the security risks associated with wastewater treatment, minimise cost loss, improve the operation level, increase the efficiency of wastewater treatment, and ensure the stability of wastewater treatment operations. Given the static migration scheme's high stability, this paper aims to introduce dynamic migration strategies in the static migration deployment of the model in the future, thereby further enhancing its decision-making capabilities.

Acknowledgements

This work was supported by:

1. Research and Engineering Demonstration on Integrated Technology of Municipal Sludge Low Carbon Treatment (H20220011), Ministry of Housing and Urban-Rural Development of the People's Republic of China (MOHURD).
2. Study on Carbon Emission Effect of Sludge Core Treatment Process in Large and Medium-sized Urban Wastewater Treatment Plants (21230731100), Science and Technology Commission of Shanghai Municipality.
3. Carbon Emission Reduction Technology of sludge digestion treatment in Urban Wastewater Treatment Plants (21YJKF-21), Shanghai Construction Group Co., Ltd.

References

[1] Wu CH. An empirical study on discussion and evaluation of green university. Ecol Chem Eng S. 2021;28(1):75-87. DOI: 10.2478/eces-2021-0007.

[2] Wu CH, Tsai SB, Liu W, Shao XF, Sun R, Wacławek M. Eco-technology and eco-innovation for green sustainable growth. Ecol Chem Eng S. 2021;28(1):7-10. DOI: 10.2478/eces-2021-0001.

[3] Chen J, Ran X. Deep learning with edge computing: a review. Proc IEEE. 2019;107(8):1655-74. DOI: 10.1109/jproc.2019.2921977.

[4] Sodhro AH, Luo Z, Sangaiah AK, Baik SW. Mobile edge computing based QoS optimization in medical healthcare applications. Int J Information Manage. 2019;45:308-18. DOI: 10.1016/j.ijinfomgt.2018.08.004.

[5] Luo R, Popp J, Bocklitz T. Deep learning for Raman spectroscopy. A review. Analytica. 2022;3(3):287-301. DOI: 10.3390/analytica.3030020.

[6] Kong F, Hu K, Li Y. A spectral-spatial feature extraction method with polydirectional CNN for multispectral image compression. IEEE J Selected Topics Appl Earth Observations Remote Sensing. 2022;15:2745-58. DOI: 10.1109/jstars.2022.3158281.

[7] Wang X, Wang S, Cao J. Data-driven based tiny-YOLOv3 method for front vehicle detection inducing SPP-net. IEEE Access. 2020;8:110227-36. DOI: 10.1109/access.2020.3001279.

[8] Su Y, Li D, Chen X. Lung nodule detection based on faster R-CNN framework. Computer Methods Programs Biomedicine. 2021;200:105866. DOI: 10.1016/j.cmpb.2020.105866.

[9] Chen Y, Wang H, Li W. Scale-aware domain adaptive faster r-cnn. Int J Computer Vision. 2021;129(7):2223-43. DOI: 10.1007/s11263-021-01447-x.

[10] Li E, Zeng L, Zhou Z. Edge AI: On-demand accelerating deep neural network inference via edge computing. IEEE Trans Wireless Communications. 2019;19(1):447-57. DOI: 10.1109/twc.2019.2946140.

[11] Krestinskaya O, James AP, Chua LO. Neuromemristive circuits for edge computing: a review. IEEE Trans Neural Networks Learning Systems. 2019;31(1):4-23. DOI: 10.1109/tnnls.2019.2899262.

[12] Khan MZ, Harous S, Hassan SU. Deep unified model for face recognition based on convolution neural network and edge computing. IEEE Access. 2019;7:72622-33. DOI: 10.1109/access.2019.2918275.

[13] Zhang J, Yang X, Xie N. An online auction mechanism for time-varying multidimensional resource allocation in clouds. Future Generation Computer Systems. 2020;111:27-38. DOI: 10.1016/j.future.2020.04.029.

[14] Holub M, Andrs O, Kovar J, Vetiska J . Effect of position of temperature sensors on the resulting volumetric accuracy of the machine tool. Measurement. 2020;150:107074. DOI: 10.1016/j.measurement.2019.107074.

[15] Lee SY, Tsou C, Huang PW. Ultra-high-frequency radio-frequency-identification baseband processor design for bio-signal acquisition and wireless transmission in healthcare system. IEEE Trans Consumer Electronics. 2019;66(1):77-86. DOI: 10.1109/tce.2019.2956627.