Hindawi
Mathematical Problems in Engineering
Volume 2022, Article ID 4992547, 14 pages
https://doi.org/10.1155/2022/4992547

Research Article

Prediction Model of Urban Street Public Space Art Design Indicators Based on Deep Convolutional Neural Network

Yang Li,[1] Jing Wu,[1] and Lingli Cao[2]

[1] *Henan College of Transportation, Zhengzhou 450000, Henan, China*
[2] *North China University of Water Resources and Electric Power, Zhengzhou 450000, Henan, China*

Correspondence should be addressed to Lingli Cao; cll@ncwu.edu.cn

Received 8 February 2022; Revised 17 March 2022; Accepted 26 March 2022; Published 28 April 2022

Academic Editor: Gengxin Sun

Copyright © 2022 Yang Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Object detection in public spaces on urban streets has always been an important research topic in the field of computer vision networks. Due to the complex and changeable scene in the prediction of public space art design indicators, there are still problems in the research of target detection algorithms in practical applications. Based on the DCNN, this paper studies the accurate detection algorithm and implementation of urban streets in complex scenes. This paper uses the characteristics of DCNN coding to collect and compress data at the same time, studies the prediction module of urban street saliency detection algorithm, and combines saliency map to determine the saliency of urban street art design indicators in the measurement domain. The experimental method can greatly shorten the index prediction scan time and solve the problems of high window calibration redundancy and long positioning time in index prediction. The experimental results show that the proposed method combining urban street mask and public space feature information can reduce other interference information, the average accuracy of target detection is increased by 0.398, and the error is reduced to 3.12%, which significantly promotes urban streets and improves recognition accuracy.

1. Introduction

With the advancement of DCNNs (Deep Learning), the accuracy of computers in urban street recognition has surpassed the human brain [1]. As the name suggests, urban street detection and recognition in complex scenes mainly include two parts: scene detection and urban street recognition. Scene detection is urban street frame selection. The main methods are sliding window method and selective search method, while the mainstream method of urban street recognition is CNN, including the method of combining object localization and DCNN, such as Fast R–CNN and SPP-net. However, although the recognition accuracy of CNN network model is high, training the large amount of data, long training time, and expensive hardware cost limit the application scope of CNN [2–5]. In addition, the scene detection algorithm, which is the basis of urban street recognition, also has problems such as high complexity, long time for urban street positioning, and inaccuracy. The DCNN algorithm combines Faster R–CNN and YOLO, which are two detection frameworks based on DCNNs. The target positioning and classification are completed at one time, which can ensure the speed of fast detection of target city streets: in addition, since the DCNN algorithm uses the different feature layers of the convolutional neural network to predict the target object with multiscale features, we have modified the scale of the prior frame in the DCNN, mainly to increase the scale of the prior frame and improve the aspect ratio of the prior frame. In order to verify the algorithm, we used the DCNN and YOLO to conduct experiments and comparative analysis on the street data set constructed by ourselves. Through the analysis, we found that the modified DCNN algorithm is compared with the original DCNN algorithm. The neural network algorithm improves the detection performance of small target city streets, but the detection speed becomes slower [6–8].

The understanding of urban streets in natural scenes usually includes two processes: the detection of artistic indicators of natural scenes and the recognition of artistic indicators of natural scenes, and the accuracy of the detection process plays a decisive role in the successful recognition of characters. In practical engineering, the detection effect of artistic indicators in natural scenes is often unsatisfactory, so that the entire art indicator recognizer cannot correctly identify the art indicators in the street space, and the result is that the understanding of the semantics of urban streets cannot be accurately expressed [9–12]. Stoian [13] et al. proposed a natural scene art indicator detection and recognition algorithm based on a sliding window mechanism using random forest classifiers and Hog features. Based on the framework of pictorial structures, the algorithm divides and fuses the candidate frame area (including the art index area) that is determined as a positive sample into words and then uses an existing dictionary to verify and correct the combined words. Chew [14] et al. proposed the use of cascaded Convolutions Neural Networks (CNNs) for the detection and recognition of art indicators, respectively. In the first stage (detection stage), CNN classifies according to a series of art index candidate regions located by the sliding window. During the classification process, the candidate regions that do not contain art indicators are eliminated. In the second stage (recognition stage), CNN performs art index segmentation based on the art index region obtained in the first stage and then identifies a single art index. Similarly, Qiu [15] uses CNN as a classifier to determine whether the sliding window contains characters. This method uses the output candidate area of CNN as the saliency value, and the sliding step size of the sliding window on the city street is 1 pixel each time, so it can generate the saliency city street with the same size as the original image. Then, the extreme value regions are generated based on these saliency maps, and candidate character regions are extracted from these extreme value regions. Then, the city street threshold method is applied to binarize the candidate art index line area, and corresponding rules (such as the interval between characters) are defined to segment individual characters in the candidate art index line area. There will be many characters in repeated regions, and most of them can be eliminated through the nonmaximum suppression operation, so that the remaining characters are basically the characters that need to be detected. Li [16] et al. proposed a Stroke Width Transform (SWT) algorithm to generate a stroke width map by extracting the parallel edges of urban streets on the edge (specifying the edge angle needs to meet the established rules). In the SWT graph, each pixel represents the possible stroke width for that pixel. Then, through the connected region extraction algorithm, the pixels that are close to the SWT image and have similar stroke widths (ratio less than 3) are fused together to form a connected region, and the connected region is used as a character candidate region. Corbane [17] believes that some connected regions that do not contain characters are filtered out through some artificially defined correlations (such as geometric rules such as the aspect ratio of the connected region, the ratio of the height and width of the stroke width in the connected region.

In this paper, an urban street recognition algorithm based on Fast Feature Fusion (FFF) is proposed, and a fast feature fusion network framework is given, which mainly includes two modules: feature fusion and local selection. Among them, in the feature fusion module, we designed a shallow CNN network, then derived the fitness function and art index coding rules that can be applied to the genetic algorithm, and combined the DCNN features with the traditional urban street features through the genetic algorithm Fusion. The local selection module can reduce the fusion features, reduce the radius within the class, and further improve the recognition rate. The algorithm can solve the problems of high training time and hardware cost of DCNN model in urban street recognition. Finally, we use the algorithm proposed in this paper to implement a DCNN application software [18–20] for the detection of specific vehicle models based on road traffic bayonet camera data. The B/S network architecture is used to complete the application software design of DCNN under the big data of urban streets [21–24].

2. Prediction of Design Metrics for DCNNs

2.1. Neural Network Index Measurement. All image sets obtained after the threshold processing of the neural network indicators are called the maximum area set. During the process of increasing the threshold, some connected areas will not change significantly with the threshold change within a certain threshold interval. Combining traditional detection algorithms with DCNN technology for research, the powerful feature expression capabilities of convolutional neural networks have made breakthroughs in target detection-related research. Driven by the DCNN technology, the performance of the current target detection algorithm is far superior to the traditional target detection algorithm, and it has become the mainstream of the current target detection algorithm. There are two main types of detection algorithms based on DCNNs.

$$\text{Fitness}\left(f\left(x_1\right)\right) - \frac{f\left(x_1\right)}{f\left(x_1\right) + f\left(x_2\right) + f\left(x_3\right) + \ldots + f\left(x_n\right)} = 0. \quad (1)$$

The DCNN breaks through the idea of the traditional target detection algorithm and lays the foundation for the successful combination of the subsequent DCNN and target detection. The main flow of the algorithm includes the following. First, an appropriate amount of candidate regions is extracted from the original street space using the Selective Search algorithm. Assuming that the three-dimensional map is used as a topography and elevation map, when water injection is used, it will slowly submerge from bottom to top. The maximum stable extreme value area focuses on the connected area that can maintain a relatively stable state within a certain range. Compared with the watershed algorithm, the essence is the same, but the difference is that the watershed will pay more attention to the threshold value that separates various connected regions, and the edge that divides these connected regions is the watershed of the connected regions. Then, the scale of the candidate region is normalized, and the feature vector of the candidate region is extracted through the trained convolutional neural network.

$$\text{Priorbox}\{f\text{unction}, f(x)\} = \begin{cases} \sqrt{f(x_1) - 1} + \sqrt{f(x_2) - 1} + \sqrt{f(x_3) - 1} + \ldots + \sqrt{f(x_n) - 1}, & x > 0 \\ \dfrac{f(x)}{\sqrt{f(x) - 1}}, & x \leq 0 \end{cases}. \tag{2}$$

Then, the extracted feature vector is input into the SVM classifier for classification. Finally, the regressor is used to correct the position of the candidate region. Enter the street space into the pretrained classifier, and directly perform regression by extracting the feature vector on the fully connected layer, which can directly return the bounding box information and category confidence of the target city street. YOLO directly uses a neural network for detection, which is an end-to-end one. Stage detection framework: its advantages are that the detection speed is fast, and the model is highly robust; in addition, since YOLO performs convolution operations on the entire input street space, the detected target has a larger field of view, and the background false detection rate is low. However, there are also shortcomings such as inaccurate positioning of target city streets and poor detection of small target city streets.

$$G(s, r, t) \times P(r - 1, s - 1, t - 1)\mathrm{d}r\mathrm{d}s\mathrm{d}t = \begin{cases} \min p[w(i, j)|i + j \subset R(\cos i, \sin i)] \\ \max p[w(i, j)|i, j \cup T = \varnothing] \end{cases}. \tag{3}$$

First, through the art index detection algorithm, the art index candidate area is obtained, and the art index is separated from the street space background; then, the urban street blocks in the separated art index area are preprocessed. Common urban street preprocessing includes the use of filtering for the algorithm denoising the city streets, enhancing the city streets, and uniformly scaling the city streets to the size that the algorithm can handle. Features, such as edge features, stroke features, and structural features of art indicators features, are classified, and postprocessing is performed using information such as high-level semantics of art indicators to obtain the final art indicator recognition results. Art index identification can be classified as a pattern matching problem, and the traditional art index identification method is generally realized by a classifier.

2.2. Urban Street Design Matching Algorithm. A representative and challenging dataset is established through the collection and processing of urban streets, and an algorithm based on neural network is proposed to detect and identify the collected urban streets. At the same time, compared with the current popular algorithms, this method shows that the research further improves the current popular urban street detection algorithm DCNN, making it suitable for detecting art index objects, using a deep neural network CRNN that combines CNN and RNN to identify art indicators, and at the same time, the research proposes a new idea. Before performing MSER detection on the image, grayscale transformation of the image is performed. MSER is a detection algorithm based on grayscale images. After MSER detection, a series of maximum stable extreme value regions will be obtained, among which there are many nested maximum stable extreme values. Value region: take these maximum stable extreme value regions as candidate regions of the text region and screen them, and artificially define some heuristic geometric constraints according to experience to remove negative samples from the candidate regions (Table 1).

In the convolutional neural network, the input street space is used for convolution operation with filters. The study found that the statistical characteristics of different regions are the same, so different regions can use the same weight operation; that is, using the same convolution kernel traversing the learned features in different regions greatly reduces the complexity. This operation enables one convolution kernel to obtain one feature in feature learning and generally uses multiple convolution kernels to obtain more urban street features such as color and outline. For the distance measurement of urban streets, two methods have been tried successively, namely, curve fitting based on least squares method and longitudinal distance measurement of urban streets based on camera focal length measurement.

2.3. Depth Convolutional Data Metrics. For the public space urban streets with the same depth convolution data and high similarity, we cannot distinguish them by attribute identification. If the public space similarity is calculated directly by the distance metric pair, but the appropriate similarity threshold cannot be established, resulting in the effect of public space recognition being poor. This method is based on a multitask learning framework, which judges whether similar public space urban streets are the same as a whole, mainly by regressing the difference values of the feature vectors of public space urban streets. At the same time, the overall judgment result is interpreted according to whether there are subtle differences between similar urban streets. The specific implementation method extracts the high-dimensional convolutional layer features of urban streets in similar public spaces, respectively, and extracts the features of Figure 1. After the geometric constraints are screened, the remaining maximum stable extreme value area can be determined as a text area, but a large number of nested areas or

Table 1: Description of art index detection.

Features		Street space 1	Street space 2	Street space 3
Convolutional neural network accuracy	Weight operation 1	62.10	10.82	83.39
	Weight operation 2	9.74	22.79	57.67
	Weight operation 3	95.16	27.41	56.92
	Weight operation 4	28.68	18.31	99.76
	Weight operation 5	93.16	32.28	66.68
	Weight operation 6	6.20	24.78	6.20
	Weight operation 7	42.90	9.64	42.90

overlapping areas will inevitably affect the detection results, so after screening, these nested and overlapping areas are merged or eliminated to output the detection result.

This paper combines the target city street detection and distance measurement of the improved YOLOv3 algorithm and modifies the function of outputting the detected city street information in the program. The program outputs 6 categories of target object classification and positioning information and 4 categories of target city streets and the longitudinal distance of the camera, and the video detection speed reaches 29.8 frames per second, which meets the real-time requirements. This method can provide a reference for car assisted driving in natural road traffic scenarios. Then, a deconvolution network is introduced, and the combined features are used to regress the saliency map with the same scale as the street city to identify public space city streets.

$$\int_{i=1}\int_{j=1} f_{ij}d_{ij}\mathrm{d}i\mathrm{d}j - \int_{i=1}\int_{j=1} f_{ij}g_{ij}\mathrm{d}i\mathrm{d}j - \int_{i=1}\int_{j=1} w_{ij}g_{ij}\mathrm{d}i\mathrm{d}j = 0. \tag{4}$$

When designing the loss function, we take the logarithmic function for the output of the logistic regression in the overall judgment task; at the same time, we introduce the mean square error function to calculate the predicted saliency map. In addition, we add a penalty factor to the first coefficient of the original log function to reduce the number of false positive samples. In order to take the task model, we have conducted a large number of experimental comparative analysis, and the results show that the model has methods such as attribute classification and distance measurement.

2.4. Neural Network Target Detection. The DCNN algorithm is mainly designed by combining the two detection frameworks Faster R–CNN and YOLO. Compared with YOLO, it can still maintain a fast detection speed; based on Faster R. The anchor points are proposed in CNN, and the DCNN algorithm uses a similar prior box mechanism. The difference is that Faster R–CNN extracts features on the last convolutional feature layer to predict the target, while the DCNN adopts the feature pyramid method to predict the target using the features of different receptive fields in multiple feature layers of the convolutional neural network. Therefore, the task of this section is to use the DCNN algorithm to realize the street detection in the high-speed road scene and realize an end-to-end, efficient and robust detection framework. Faster R–CNN improves the way of generating candidate regions and introduces boxes to replace the selective search algorithm, which greatly improves the detection speed.

The original YOLO algorithm directly takes feature map of the entire city street in Figure 2 and uses convolutional layers to predict bounding boxes in different regions, which greatly improves the detection speed. Another YOLO-based approach is a DCNN, which is used to predict the bounding boxes of city streets. The regression problem is simplified by achieving translation and scale invariance of the regression by using default boxes of different scales. At present, in most art index recognition algorithms, art index detection and art index recognition are two completely independent modules. The city streets are input into the art index detection system. First, through the art index positioning, roughly locate the art index candidate area, and then pass verification, and further distinguish the art indicator area from the nonart indicator area. The art index box obtained by the art index detection is input into the text recognition system, and the art index segmentation step divides the art index into characters, obtains the most accurate character outline and provides it to the subsequent character recognition step, and finally characterizes the segmented urban street blocks, getting the final identified art index sequence.

$$\begin{bmatrix} f_{ij} & 1 \\ -1 & -f_{ij} \end{bmatrix} \times \begin{bmatrix} g_{ij} & \min(i-1, j-1) \\ -\min(i, j) & -g_{ij} \end{bmatrix} - \begin{bmatrix} g_{ij} & 1 \\ -1 & -g_{ij} \end{bmatrix} \times \begin{bmatrix} g_{ij} & \max(i, j) \\ \max(i-1, j-1) & -g_{ij} \end{bmatrix} = 0. \tag{5}$$

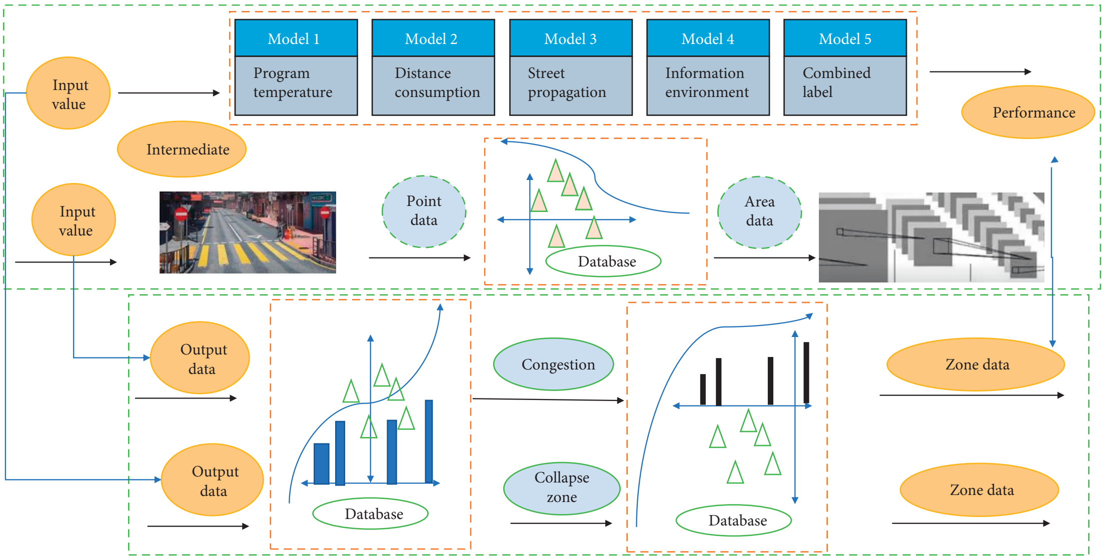

Figure 1: Depth convolution data measurement framework.

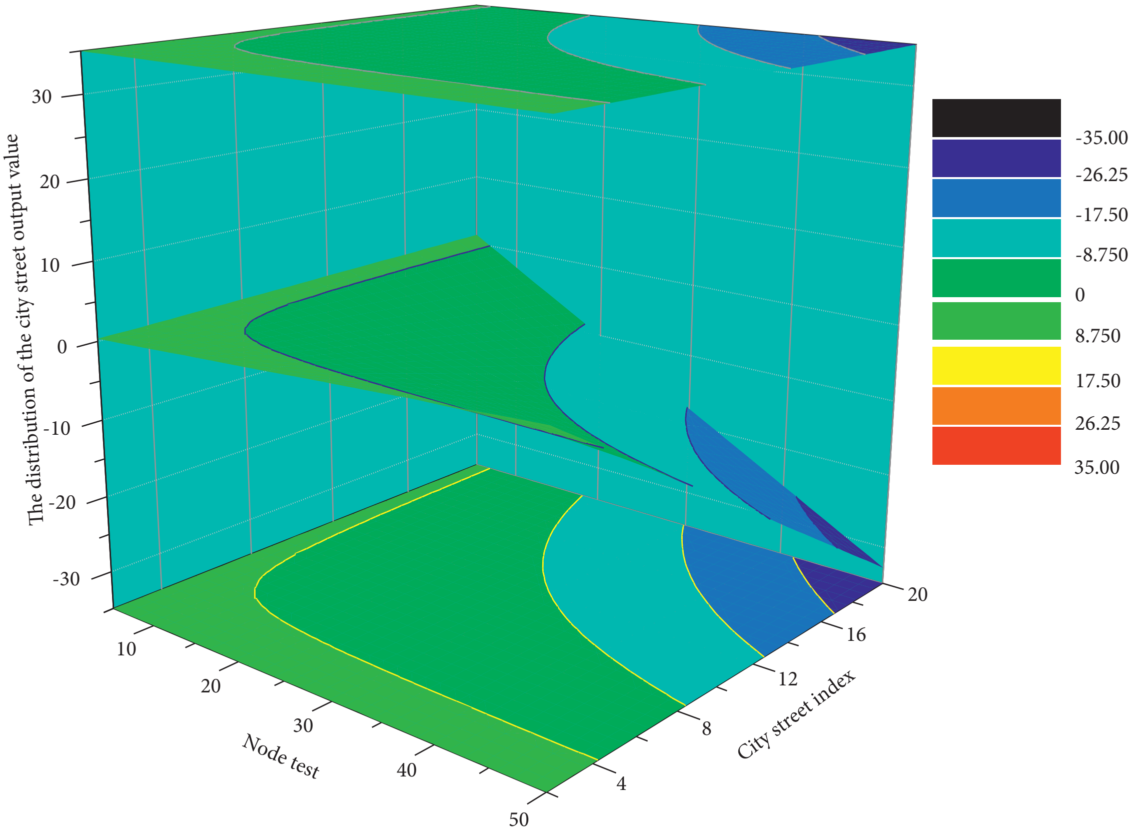

Figure 2: Search distribution of urban street index training network.

The DCNN is essentially a feature extraction process, that is, let patchl and patch be inputs to the two branches of the network, respectively, and then make a similarity loss function on the feature vector extracted in the last layer of the network to train the network. The feature extraction process of deep convolutional networks for the two inputs is independent of each other. However, what is different from the explicit feature extraction of deep convolutional networks is that, in the dual-channel neural network, the two input grayscale images are combined together and regarded as a dual-channel city street, and the convolution kernel and the input two-channel matrix is weighted, combined, and mapped to a high-dimensional space, and the error loss function for evaluating the similarity is directly learned by using the output value of the fully connected layer, ignoring the process of feature extraction.

3. Construction of a Prediction Model for Urban Street Public Space Art Design Indicators Based on DCNNs

3.1. Multitask Deep Convolutional Art Metric Encoding. In the multitask depth convolution process, the traditional SIFT feature has a process of selecting key points, while the dense SIFT feature collection ignores the process of selecting key points but densely selects points in a certain area like a sliding window and calculates the SIFT description algorithm. We densely compute SIFT features with a sliding window of size 24×24 with a sliding step of l, while repeating the process on different scale spaces. Finally, 128-dimensional SIFT features can be obtained. Due to the large amount of data, such as the subsequent art index coding learning, principal component analysis (PCA) is used to reduce the dimension of 128-dimensional features to 64-dimensional.

The direction of LSTM is fixed, and it can only use the context of the past, but for the sequence based on city streets, the context of the two directions before and after it affects each other, and the two forward and backward LSTMs are combined, and a bidirectional LSTM can be generated. In addition, by stacking multiple bidirectional LSTMs, a deep bidirectional LSTM network can be obtained, and the deep structure can more effectively perform high-level abstraction.

$$\text{if}\begin{cases} y_i - x > 0 \\ y_i - x < 1 \end{cases}, \begin{cases} \sum \sqrt{a^2 - b^2} - \sum \sqrt{a^2 + b^2} + 1 = 0 \\ \sum \sqrt{a^2 - b^2} + \sum \sqrt{a^2 - b^2} + 1 = 0 \end{cases}. \tag{6}$$

The DCNN algorithm uses feature maps of different scales to detect objects. Its advantage is that it can use large-scale feature maps to detect N4, target city streets, and use d, f10 feature maps to predict large-scale target city streets in the original image. The 8×8 priori frame set by each feature unit is relatively small, which can be responsible for detecting small-scale target streets far from the monitoring equipment. In the YOLO detection framework, the last layer of the network is a 4096-dimensional feature vector. By regressing the feature vector, the category confidence of the target city street and the location coordinates of the box are predicted. The core of the DCNN algorithm is to directly use the convolution kernel on the feature map to predict the prior boxes and categories of a series of target city streets. For the feature maps of different convolutional layers in Figure 3, 3×3 convolution kernels are used for prediction.

The street space input to the neural network uses a single pixel that constitutes the street space as a feature node. For a color street space with a resolution of 64×64, the input feature of the fully connected network is $64 \times 64 \times 3 = 12288$ nodes. The input layer of the neural network in it contains 12288 feature nodes. In order to take the model description, the next layer of the network also has 12288 nodes. Each neuron node in the two layers is connected to each other, and then the weight matrix is calculated, which is equal to $12288 \times 12288 \approx 150$ million parameters; since this $64 \times 64 \times 3$ street space is very small, there is no problem with training so many parameters. But if this is a 1280×720 color street space, the fully connected network weight matrix will become very large. R–CNN improves the way of generating candidate regions and replaces the selective search algorithm, which greatly improves the detection speed.

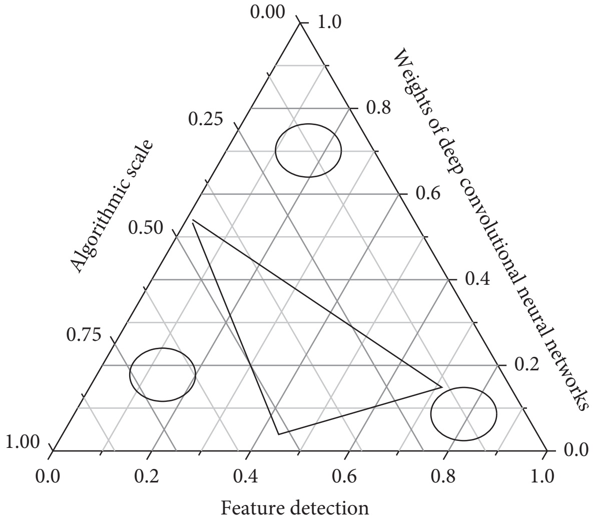

Figure 3: Scale feature detection comparison of DCNN algorithms.

$$\begin{cases} \min i = sig\,\text{mod}\left[\sum_{n=1}^{i} \sum_{n=1}^{i} f_i \mathrm{d}_j, \max(i, j)\right] \\ \min j = \log(i, j) \times \log(i-1, j-1) \\ \min k = \int_j^v \left[w(x, y)_{ij} + f(x, y)\right] \mathrm{d}x\mathrm{d}y \end{cases}. \tag{7}$$

The original YOLO algorithm directly improves the detection speed. Another YOLO-based approach is a DCNN, which uses default boxes to predict the bounding boxes of city streets. The regression problem is simplified by achieving translation and scale invariance of the regression by using default boxes of different location. The DCNN maintains the same high detection speed as YOLO while improving the original YOLO detection performance. After detection by MSER, a large number of text candidate connected regions are obtained, including a large number of text regions and nontext regions, and even a lot of nested connected regions. In order to deal with the large number of redundant connected areas brought by the detection algorithm, Matas artificially prescribes some heuristic rules based on experience to filter these redundant areas, which mainly include three aspects: area size, area ratio, and height and widths.

3.2. Identification of Public Space in Urban Streets. In order to identify more comprehensive features of urban streets, a convolutional layer is generally composed of multiple convolution kernels. The result obtained by convolution of the convolution kernel is called a feature map. After a convolutional layer, multiple feature maps are generated for each city street. This also completes a high-dimensional feature extraction of the data. A network structure composed of multiple convolution kernels is called a convolution layer. The number of convolution kernels in a convolution layer means as many convolution channels as there are, and how many new feature maps are derived from an input image. Tanh represents the excitation function. After the city street is convoluted, the abstract feature of the city street at a certain level is obtained. In the convolutional neural network, people can understand the physical meaning of the feature maps of the first few convolutional layers, but with the deepening of the convolutional layers, the physical meaning of the feature maps has become more and more complex and abstract.

$$\frac{1}{1000}\times\frac{\sum Y\left[i|i=0,1,2,3\ldots j-1,j\right]_j}{\sum_{i=1}^{N}\left[x(i),y(i)\right]_j}-\frac{a}{1000}\times\frac{\sum_{i=1}^{N}X_i\cdot W_j}{w(i,j)}=0. \tag{8}$$

In essence, each layer of the network can be regarded as a filter, and data is transmitted forward through each filter. The neural network simulates the human brain through this transmission process. At the same time, the network parameters are corrected by backpropagation with the help of gradient descent and chain derivation. However, in traditional urban street processing, feature vector extraction is performed in the pixel domain. The more the pixels, the larger the network parameter scale. Therefore, if the scale of the network parameters is too large, the huge number of parameters to be trained means huge training costs, so if you want to apply the neural network to urban street processing, you must reduce the parameters to speed up the training speed.

The convolutional layer in the neural network was born based on this idea. The convolution operation of city streets can be understood as a filtering process. In the S4 layer, the 16 images of size 10 ∗ 10 from the C3 layer are continuously convolved through a 2 ∗ 2 convolution kernel to obtain 16 5 ∗ 5 downsampled images. Then, we connect the C5 convolution layer to form 120 1 ∗ 1 convolution results, connect the 120 convolution values to the fully connected layer F6, use the black and white art index code (-1 for white, 1 for black) to encode the bitmap art index, and finally connect the fully connected one and take the number of connection nodes to 10; you can represent 0 to 9 in decimal.

When the number of iterations in Figure 4 reaches 3000, the accuracy of the validation set reaches 0.97 and maintains a dynamic balance. After completing the network model training, we tested the test dataset proposed in the previous section with the model saved at 8000 iterations. After completing the test on the test data set, we use the matrix to show the verification results of the vehicle color and model on the test data set. For the convolutional layer, a pooling layer (-1, +1) is usually added afterwards. The residual error of the convolution kernel in the pooling layer corresponds to convolutional layer. Therefore, the pooling layer needs to be it. The residual item of the pooling layer is upsampling operation, for example, to ensure that the size of the pooling layer and the convolutional layer are the same, and finally multiplied by the partial derivative of the excitation function and the weight coefficient, respectively, and the residual of the convolutional layer can be obtained.

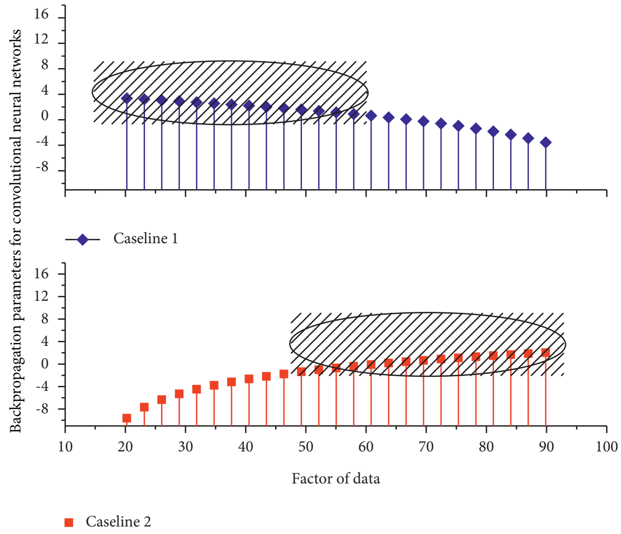

Figure 4: Parameter distribution of backpropagation in DCNN.

3.3. Prediction of Art and Design Indicators. In view of the large difference between the bounding boxes of urban street art design indicators, based on the YOLOv3 algorithm, an improved YOLOv3 deep residual convolutional neural network architecture model with 5 feature detection maps and 155 layers is designed, and the improved architecture increases by 7. The output detection maps of ×7 and 104 × 104 are responsible for detecting larger and smaller target city streets in the road traffic field of view, respectively. The street space for network training comes from the BDD100 K dataset. The results reveal that, compared with the YOLOv3 architecture, the improved multidetection map YOLOv3 network achieves an average accuracy of 54.48% on the validation dataset, an increase of 5.11 percentage points.

The common disadvantage of both sigmoid and tanh activation functions is that when the feature combination value z is particularly large or small, the slope of the function will become extremely small, resulting in a very slow gradient descent of the loss function, and even disappearance of the gradient. Problem: the derivative of the ReLu function is 1, which does not cause the gradient to become smaller. Using the ReLu activation function can train a deeper network, two linear functions can be implemented in a single judgment statement in the program, and the sigmoid function needs to perform time-consuming four arithmetic operations on floating-point numbers. In the process of network weight training, the ReLu activation function is used in neural networks. The error loss will drop faster than using sigmoid or tanh activation functions.

$$\overbrace{\frac{f_1f_2}{v1}\sin(i,j,k),\frac{f_1f_2}{v2}\sin(i,j,k),\ldots,\frac{f_1f_2}{v3}\sin(i,j,k)=1}^{\lim(i,j,k)}. \tag{9}$$

The pooling layer, also known as the downsampling layer, is to sample data from a neighborhood of a city street. Dimensionality reduction not only speeds up model training and prevents overfitting, but also enhances robustness. From the actual effect, the convolutional neural network has strong robustness to the spatial scale transformation of urban streets, and the pooling layer plays an extremely important role. The essence of the Softmax layer is a nonlinear classifier. Logistics regression can be seen as a special case of Softmax regression when $k = 2$. Softmax function is the logistic function of k classification. In the DCNN, Softmax is usually used in conjunction with the BP network, combining the hidden layer mapping mechanism of the BP network with the Softmax multiclassification mechanism. In Softmax regression, if x is input, the probability distribution for the label y is as follows.

The main function of the output layer in Table 2 is classification. The possible probability of each class is obtained by calculation as the prediction result. In the output layer, a loss function (10ss action) needs to be established, also called the objective function, that is, to measure the prediction of different result, which is the goal that the network training process needs to be optimized. Common loss functions include SX Loss, Euclideanloss, and contrastiveloss. The most commonly used loss function in practical classification tasks is the SomnaxLoss function. The convolutional neural network can obtain the prediction ability. The convolutional neural network can establish the optimal mathematical relationship between the input data and the label by training the network parameters. The convolutional neural network adopts the classic error backpropagation algorithm (BP), and the algorithm with more parameters generally uses the stochastic network parameters to realize the real result.

4. Application and Analysis of the Prediction Model of Urban Street Public Space Art Design Index

4.1. Data Preprocessing of DCNNs. X1 and X2 are the two inputs to the network, and W is a shared parameter of the network. The purpose of training such a neural network is to minimize its loss function value when X1 and X2 belong to the same category; maximize its loss function value. Among them, the most commonly used loss function was originally derived from Yann LeCun's proposal for dimensionality reduction through invariant mapping. The test set is tested by the network model, and the test results of the test set of vehicles, pedestrians, and cyclists are obtained. AP is the average accuracy rate, and the recall rate and accuracy rate can only show the limitation of the single point value.

And AP is an indicator that can reflect the global performance, the area below the P-R curve drawn in the figure, and the value of AP. mAP is the average of different kinds of AP values. From the P-R curves of the test set of vehicles, pedestrians, and cyclists, it can be seen that the detection effect of vehicles is relatively good, and its detection effect is much better than that of pedestrians and cyclists, while the detection effect of pedestrians and cyclists is average. The analysis found that because the number of vehicles in the training set is large, there are many categories, and the target size of the vehicle is large.

$$\left\langle \begin{array}{l} \log\left[\dfrac{x(i)}{x(j)}\right] = tx(i,j) \times \dfrac{f_1 f_2}{d(i,j)} \sin g(i,j) \\ \\ \log\left[\dfrac{y(i)}{y(j)}\right] = tx(i,j) \times \dfrac{f_1 f_2}{d(i,j)} \cos g(i,j) \end{array} \right. . \tag{10}$$

Among them, X indicates the confidence that the target contained in the prior box belongs to category P, Q represents the weight of the two, and the default value is equal to 1. N represents the number of matched prior boxes. To speed up training, the Alex model uses two GPU parallel structures. It can be found that the network is divided into parts, and the two parts have the same structure. Different parts are allocated to different GPUs, so that multichannel parallel processing of data can be realized. Data parallelism divides the training data into two parts to obtain two model parameters and then combines the two parameters to obtain the final network model for model parameters.

$$u_{obj}^{coord} \prod\prod \lambda(i,j) * g(x) - \prod\left[u^2(x_1), u^2(x_2), u^2(x_3), \ldots, u^2(x_{n-1}), u^2(x_n)\right] = 0. \tag{11}$$

The parameters of the entire network are divided into two parts, and the two parameters are trained on different GPUs with the same data to perform full city-street-to-city-street saliency prediction. The collected targets are extensive, which makes the network model's generalization ability for vehicle detection relatively strong, while for pedestrians and cyclists, due to the changeable shapes of people and changing clothes, the collection of data sets is based on vehicles on the road. In the video detection of the driving recorder, the probability of pedestrians and cyclists appearing on the road is relatively small, so the collected training sets of pedestrians and cyclists are relatively small, the number of samples is too small, and the parameters in the training process are relatively small. The selected fit in Figure 5 is good.

The training process of the network in the experiment is divided into three stages. First, all test data sets are pretrained on the SynthText data set. The SynthText data set contains 800,000 synthetic art indicators of city streets, and the city streets are rendered by mixing art indicators. Since the positions and transformations of the artistic indicators are to continue training the pretrained model obtained in the first step using the data set collected and produced by oneself, the Lasso method performs two-step variable screening, so that not only the truly influential independent

Table 2: Classification of neural network output functions.

	Regression 1	Regression 2	Regression 3	Regression 4	Regression 5
Input value	2.86	3	3.14	3.28	3.42
Minimum value	2.17	2.27	2.37	2.47	2.57
Maximum value	1.48	1.54	1.6	1.66	1.72
Main value	0.79	0.81	0.83	0.85	0.87
Average value	0.1	0.08	0.06	0.04	0.02
Output value	0.59	0.65	0.71	0.77	0.83

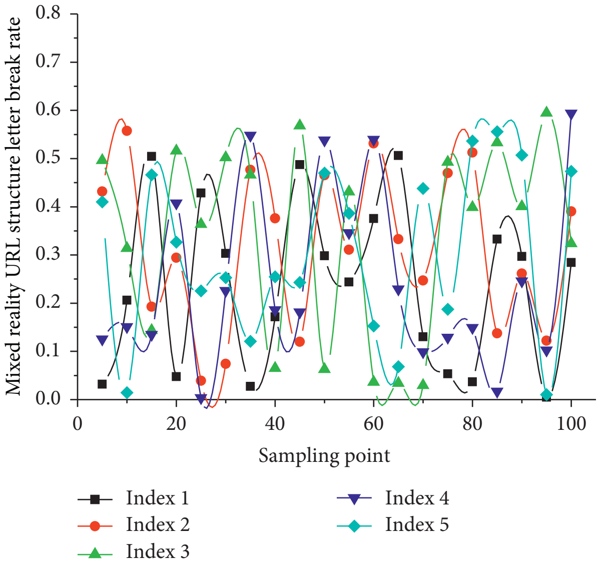

Figure 5: Distribution of truncation rate of hybrid convolutional neural network.

variables can be selected, but also with the increase of samples or dimensions, the amount of computation added by the method in this paper is far less than the above data mining methods; secondly, in the first process of selecting nonparametric independent variables, the B-spline basis expansion is performed on each variable, which fully considers the nonlinear relationship between the independent variable and the dependent variable and has strong adaptability and flexibility; but the fitting effect is obviously better than that of the stepwise regression and Lasso methods. Finally, we continue to train the model with samples to obtain better model parameters.

4.2. Prediction and Simulation of Public Space Art Design Indicators. Stepwise regression is to gradually introduce independent variables into the regression model, and each introduced independent variable must be tested. The general regression equation test method is the F test. At the same time, each introduced variable is tested one by one. If the variable does not pass the test, the currently introduced independent variable should be eliminated. If the currently introduced independent variable passes the test, the variable will be retained and implemented step by step until the end. Full connection is actually matrix multiplication, which is equivalent to feature space transformation, changing high-dimensional to low-dimensional.

$$\max\{(1-x)(1-y)\} - \max\{m\arg in - m\arg out\} = 0. \tag{12}$$

Therefore, after the convolutional layer operation, the pooling layer is generally connected to reduce the dimension of the feature map, and the feature can also be generalized. At present, the most commonly used pooling methods are as follows: (1) maximum pooling; (2) average pooling; (3) random pooling. The pooling operation is similar to the convolution operation. It is also through the sliding window according to the convolution operation. In the window, according to the pooling method, the corresponding value is selected, then the corresponding weight is multiplied, and the bias term is added. The results are output as eigenvalues.

$$\left\langle \begin{array}{l} D\dfrac{[1-q(t), box(t)]}{D(q,t)} \longrightarrow IOT[(q,t),(q,t-1)] \\ \\ \dfrac{D(q,t)}{D(q-1,t)} \longrightarrow 1 - IOT(box, box-1) \end{array} \right. . \tag{13}$$

The advantages of the method based on FisherVector art index coding are as follows: firstly, mapping features to high-dimensional space can improve the expressive ability of urban street features; secondly, different numbers of feature art index codes can be spliced into art index coding vectors of the same length. The two city streets to be verified are separately obtained by means of Fisher Vector art index encoding to obtain the art index codes, and then the distance of the vectors of the two city streets is calculated. In addition, linear SVM or other classifiers can also be used for measurement, that is, to obtain the squared difference of two feature vectors of two urban streets and input them into the SVM classifier, that is, to verify the two urban streets through the idea of classification. After the VGG16 network, a new network structure appeared, which is called a deep residual network. ResNet was introduced in detail in the previous chapters and will not be described in detail here. This section uses deep residual networks ResNet-50 and ResNet-101 as shared convolutional layer networks for training and testing on the BelagLogos dataset, respectively.

In the DCNN algorithm in Figure 6, two convolution kernels of size 3×3 are used to perform convolution operations on the layers respectively. The number of prior boxes in the graph is 4, 6, 6, 6, 4, and 4 in sequence, so the number of prior boxes obtained by the DCNN network is 8732, and the calculation formula is as follows. The two 3×3 mentioned above convolution kernel: one is used to predict the confidence of classification. If there are x prior boxes and y categories of classification, then $x*y$ convolution kernels are required; the other is used to predict the position of regression, if the number of prior boxes is x, then $4*x$ convolution kernels are required. Finally, in the second step, a nonparametric additive model is established, and while further variable screening is performed, the functional relationship between variables is well grasped, so the fitting effect is also extremely good. According to the analysis results of the example with 99 samples and 53 variables, it can be seen that the number of variables selected by the method in this paper is 5 and 12 less than that of the stepwise regression and Lasso methods, respectively.

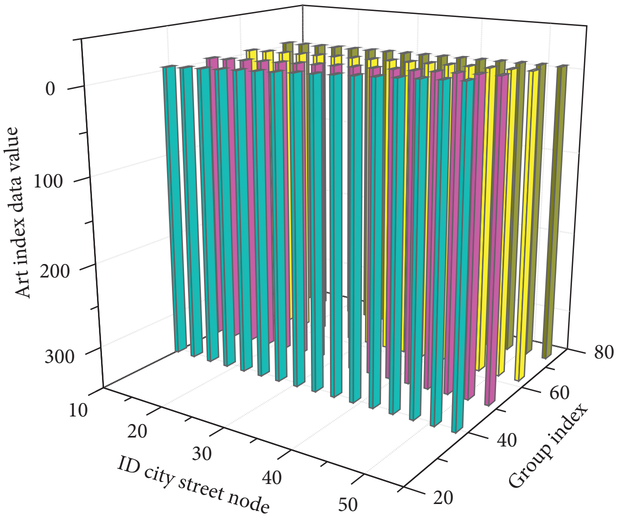

Figure 6: Coding distribution of urban street art indicators.

$$\oiint W[q(t,x)]\mathrm{d}x\mathrm{d}t \times \oiint W[q(t,x-1)]\mathrm{d}x\mathrm{d}t = \oiiint W(q-1,t-1,x-1)\mathrm{d}q\mathrm{d}t\mathrm{d}x. \tag{14}$$

In the training phase, we first utilize VGG 16 in ILSVRC CLS. Perform and train on the LOC data set, and then use the model to extract feature of different scales of the training samples on multiple convolutional layers, and match the a priori boxes of different feature layers with the samples. According to the above matching principle, if the a priori box matches the ground truth successfully, it means that the a priori box contains the target. However, since there are some differences in position and scale between the a priori box and the ground truth, the purpose of training is to regress the prior with the same scale and position as the ground truth as much as possible under the premise that the category score of the a priori box is high. As shown in the text, the training sample contains the ground truth of two streets of different colors. For the ground truth of the red street, the purple dashed box in the text matches it.

4.3. Example Application and Analysis. Since the number of ground truths in the training samples is far less than the number of a priori boxes, if only matching is performed according to the principle of maximum intersection ratio, most a priori boxes are negative samples, which will cause extremely unbalanced positive and negative samples. The traditional linear regression model may miss important information, and the obtained results cannot explain the dependent variable well, so this paper is based on the cubic B-spline (Basis-Spline) expansion, the specific form is shown in the following formula, and it can be seen that the cubic B-spline first includes the primary, quadratic, and cubic terms of the independent variable, and then the piecewise function. Since the nodes of the piecewise function are not fixed for different samples, this method is called nonparametric in this paper. Return: therefore, the DCNN algorithm proposes a second a priori box matching principle: for the a priori box that is not successfully matched under the first matching principle, if the intersection ratio is with the positive sample,

$$\max(i,j)+\max(i-1,j-1)+\max(i-2,j-2)+\ldots+\max(0,0)-f(i,j)=0. \tag{15}$$

Among them, n is the number of feature maps. In the experiments in this section, $n=6$, and Sk represents the ratio of the size of the prior frame to the feature map. S1 and Smin are two hyperparameters that represent the scale, respectively. In the experiments in this section, we take 0.8 and 0.05, respectively; through the above formula, it can be calculated that Sk = [0.05, 0.20, 0.35, 0.50, 0.65, 0.80]. In the experiment of this section, the aspect ratio $a = \{1, 2, 4, 6, 1/2, 1/3, 1/4\}$ of the prior frame is set, and the purpose is to select a priori with a richer aspect ratio box to improve the detection effect of small target city streets. Overall, the results of the whole experiment are basically greater than 0.9. However, the average accuracy of ResNet-101 is more than 0.2 lower than that of ResNet-50, and more than 0.1 lower than that of VGG16, indicating the logo street space in the BelagLogos dataset, ResNet. The depth of -101 is too deep, making gradient disappearance worse. The above experimental results show that the ResNet-50 model works best as a logo classification network. We will use the ResNet-50 of Figure 7 as the shared convolutional layer of the logo classification network.

First, we select the current detection framework with high accuracy and fast running speed, add the urban street detection algorithm branch on the basis of the indoor road segmentation that has been implemented in the previous chapter, and realize parallel operations of classification, detection, and segmentation and use it at the end. The sum of the three loss functions is used to adjust and update the network parameters to achieve optimality. We select a deeper basic shared framework to extract features, expand data sets, and improve optimization algorithms. Finally, the training and verification are carried out on the MS COCO public data set and the collected and produced indoor experimental scene data set, which compress the results of a single task to show the advantages of the combined algorithm.

$$\lim_{i,j\longrightarrow\infty} \sqrt{\sum Z(\alpha,T)\times(x-i,x-j)} = \sqrt{\frac{\cos\left[Z_i(i,j)-Z_j(i,j)\right]}{\cos\left[Z_i(i,j)-Z(i,j)\right]}} + \sqrt{\frac{\sin\left[Z_i(i,j)+Z_j(i,j)\right]}{\sin\left[Z_i(i,j)+Z(i,j)\right]}}. \tag{16}$$

The random cropping objects are the same as art indicators in urban streets in natural scenes. The random clipping strategy studied was to randomly set the minimum fixed-size city street, which is fed into the network. Each layer of AlexNet contains only one convolutional layer, and the size of the convolution kernel is 7 × 7, while each layer of VGGNet contains multiple convolutional layers, the size of the convolution kernel is 3 × 3, and the size of the convolution kernel is 3 × 3 in each layer. Max Pooling is used between products. VGGNet does not use local response normalization (LRN), because local response normalization cannot improve the performance of the ImageNet large-scale visual recognition challenge data set, and it will only consume more memory and longer computing time, so VGGNet uses ReLu behind all convolutional layers. Due to the limited number of datasets collected in the study, it is easy to overfit the trained model. The study performed data augmentation operations on the data in the datasets in Table 3, including changing the size of urban streets, fuzzy operations on urban streets, and adding noise to city streets and more.

In addition, by default, each feature map will have a prior box that is set with two square prior boxes with an aspect ratio of 1 but different scales. The entire network only uses the operations learned from this feature from the beginning, which greatly improves the running speed. Generally speaking, the better the classification result of a deep convolutional neural network on the ImageNet dataset, the stronger the generalization ability of its deep features. Since VGGNet has been trained in the ImageNet large-scale visual recognition challenge set with obtained excellent classification results, this paper retains 5 groups of 13 convolutional layers of the original VGGNet network structure, which will be used for extraction. 2 fully connected convolutional layers for features and 1 fully connected layer for classifying features are removed, but the input image of size 256 × 256 will change the feature image obtained after five pooling layers of the convolutional neural network. As 1/32 of the input image, that is, 8 × 8, such a resolution obviously does not meet the needs of the output result. Classic series of models of R-CNN and Faster R-CNN, it is necessary to extract the proposed frame first, determine the fixed size of the feature map through ROI pooling, and then iteratively correct the position of the proposed frame through training.

$$\text{for}\begin{cases} i \in [-1,0] \\ j \in [0,1] \end{cases}, \ \max\sqrt{[d(i)+d(k,i)]} = \sqrt{\max\left[f(x)-f(x_i)\right]+\min\left[f(y)-f(y_i)\right]}. \tag{17}$$

The overall prediction accuracy was tested by using the network models saved at 10,000, 20,000, and 50,000 network iterations, respectively. When the number of iterations was 10,000, the prediction accuracy was 92.0%, and when the number of iterations was 20,000, the prediction accuracy was 93.6%. When the number of times is 50,000, the prediction accuracy is 94.4%. Instead of using a fixed 3 × 3 kernel size, we use different sized kernels to smooth out predictions. The

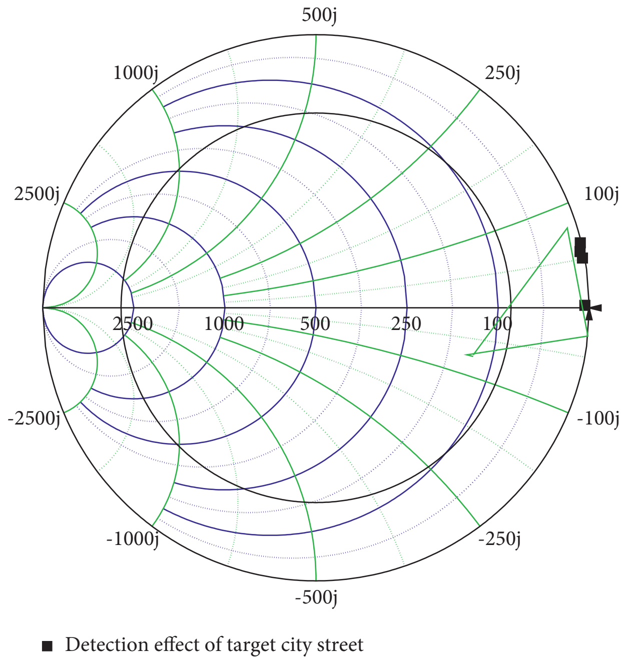

Figure 7: Comparison of detection effects of target city streets.

Table 3: Data augmentation algorithm for urban street art indicators.

Number	Data augmentation algorithm	Code text
1	The study performed test	Import java.awt. ∗ ;
2	With a threshold of dxdt	Import java.awt.event. ∗ ;
3	The random $q(a,b)$	Import.avax.swing. ∗ ; / ∗ ∗ jframe ∗ /
4	Each cropped x^2+t^2 region	Public class appgraphinout {
5	Add noise to $t, x-1$	Public static void main(string args[]){
6	On urban streets min . (t)	New appframe();
7	Data augmentation operation	Static void print(string prefix, int n)
8	Overfit the trained $q > x$ model	Import java.awt ∗ ;
9	Datasets collected in $\sqrt{x^2+t^2}$	Import javax.swing. ∗ ;
10	Compares the $F(q-1,t)$ results	Class appframe extends jframe
11	A prior box with $1+q(t)$	System.out.print(prefix+" "+s+"\n");
12	Extract the proposed $W(q(t,x))$	Jtextfieldin = new jtextfield(10);
13	On the basis of $\delta(q(a,b))$	Jbuttonbtn = new jbutton("");
14	Each feature map min . $(t-1,x)$	Jlabel out = new jlabel("");
15	The entire network map	Public appframe()
16	The operations learned from it	While(s.length()<4)s = "0"+s;
17	The prior $g(x)\cup g(t)$ box	Setlayout(new flowlayout());
18	Sk represents the $\sqrt{a^2-b^2}$	Getcontentpane().add(in);
19	Actual scale of $a+b$	G.drawoval(x0-r,y0-r,r ∗ 2,r ∗ 2);

new network structure does not need to be retrained; it only needs to use different scale network models to make predictions and finally fuse all the available data. The test Faster R-CNN achieves 70% accuracy on VOC2007. In Faster R–CNN, the I-Rush N (Region Proposal Nerdors) network layer is introduced to replace the previous proposal box selection operation, and the softmax loss is used to regress the position of the proposal box, enabling testing Faster R. The accuracy rate of CNN on VOC2007 reaches 73%, which greatly improves the accuracy.

5. Conclusion

This paper deeply studies the core composition of convolutional network and the principles of each operation, including basic feature extraction network, downsampling, upsampling, transposed convolution, and other operations. In view of the low accuracy of the original segmentation network algorithm in the indoor complex road environment, an improved fully convolutional neural network art index segmentation algorithm is designed. The whole framework is divided into two parts. The shallow features are combined with the deep abstract features, and multiscale art index information is added, which makes the reference information for the final prediction and segmentation of the network more comprehensive. By completing the verification on the experimental scene data set, the convergence speed is improved. In order to better realize indoor scene understanding, a multitask segmentation algorithm based on fully convolutional neural network is proposed, which realizes the segmentation of indoor complex environment. By adding the target detection branch to the improved urban street segmentation algorithm, the task of combining urban street classification, detection, and segmentation is innovatively realized, and the network is further optimized. By improving the network loss function, using deeper feature extraction network and model optimization algorithm to improve the algorithm, and using the transfer learning idea to adjust the training, the results prove good robustness in complex indoor segmentation scenes and, at the same time, improve the average accuracy.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by Henan College of Transportation.

References

[1] S. Law, C. I. Seresinhe, Y. Shen, and M. Gutierrez-Roig, "Street-Frontage-Net: urban image classification using deep convolutional neural networks," *International Journal of Geographical Information Science*, vol. 34, no. 4, pp. 681–707, 2020.

[2] J. Mast, C. Wei, and M. Wurm, "Mapping urban villages using fully convolutional neural networks," *Remote Sensing Letters*, vol. 11, no. 7, pp. 630–639, 2020.

[3] D. Verma, A. Jana, and K. Ramamritham, "Classification and mapping of sound sources in local urban streets through AudioSet data and Bayesian optimized Neural Networks," *Noise Mapping*, vol. 6, no. 1, pp. 52–71, 2019.

[4] R. Li, S. Wang, F. Zhu, and J. Huang, "Adaptive graph convolutional neural networks," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[5] S. Srivastava, J. E. Vargas-Muñoz, D. Swinkels, and D. Tuia, "Multilabel building functions classification from ground pictures using convolutional neural networks," in *Proceedings of the 2nd ACM SIGSPATIAL international workshop on AI for geographic knowledge discovery*, vol. 21, pp. 43–46, Seattle, WA, November 2018.

[6] S. Pouyanfar, Y. Tao, A. Mohan et al., "Dynamic sampling in convolutional neural networks for imbalanced data classification," in *Proceedings of the 2018 IEEE conference on multimedia information processing and retrieval (MIPR)*, vol. 3, pp. 112–117, IEEE, Miami, FL, USA, April, 2018.

[7] K. Gkolias and E. I. Vlahogianni, "Convolutional neural networks for on-street parking space detection in urban networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 12, pp. 4318–4327, 2018.

[8] S. M. Azimi, P. Fischer, M. Körner, and P. Reinartz, "Aerial LaneNet: lane-marking semantic segmentation in aerial imagery using wavelet-enhanced cost-sensitive symmetric fully convolutional neural networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 5, pp. 2920–2938, 2018.

[9] F. Alhasoun and González, "Urban street contexts classification using convolutional neural networks and streets imagery," in *Proceedings of the 2019 18th IEEE International Conference On Machine Learning And Applications (ICMLA)*, vol. 21, pp. 1198–1204, IEEE, Boca Raton, FL, USA, December 2019.

[10] N. Ahmed, M. N. Islam, A. S. Tuba, M. R. C. Mahdy, and M. Sujauddin, "Solving visual pollution with deep learning: a new nexus in environmental management," *Journal of Environmental Management*, vol. 248, Article ID 109253, 2019.

[11] K. Jaiswal and D. K. Patel, "Sound classification using convolutional neural networks," in *Proceedings of the Cloud Computing in Emerging Markets (CCEM)*, vol. 13, pp. 81–84, IEEE, Bangalore, India, November 2018.

[12] M. Li, J. Qin, D. Li, R. Chen, X. Liao, and B. Guo, "VNLSTM-PoseNet: a novel deep ConvNet for real-time 6-DOF camera relocalization in urban streets," *Geo-Spatial Information Science*, vol. 24, no. 3, pp. 422–437, 2021.

[13] A. Stoian, V. Poulain, J. Inglada, V. Poughon, and D. Derksen, "Land cover maps production with high resolution satellite image time series and convolutional neural networks: adaptations and limits for operational systems," *Remote Sensing*, vol. 11, no. 17, p. 1986, 2019.

[14] R. F. Chew, S. Amer, K. Jones et al., "Residential scene classification for gridded population sampling in developing countries using DCNNs on satellite imagery," *International Journal of Health Geographics*, vol. 17, no. 1, pp. 15–17, 2018.

[15] C. Qiu, M. Schmitt, C. Geiß, T.-H. K. Chen, and X. X. Zhu, "A framework for large-scale mapping of human settlement extent from Sentinel-2 images via fully convolutional neural networks," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 163, pp. 152–170, 2020.

[16] Y. Li, X. Zhang, and D. Chen, "Csrnet: dilated convolutional neural networks for understanding the highly congested scenes," *Proceedings of the computer vision and pattern recognition*, vol. 21, pp. 1091–1100, 2018.

[17] C. Corbane, V. Syrris, F. Sabo et al., "Convolutional neural networks for global human settlements mapping from Sentinel-2 satellite imagery," *Neural Computing & Applications*, vol. 33, no. 12, pp. 6697–6720, 2021.

[18] Q. He, Z. Li, W. Gao et al., "Predictive models for daylight performance of general floorplans based on CNN and GAN: a proof-of-concept study," *Building and Environment*, vol. 206, p. 108346, 2021.

[19] D. Griffiths and J. Boehm, "Improving public data for building segmentation from Convolutional Neural Networks (CNNs) for fused airborne lidar and image data using active contours," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 154, pp. 70–83, 2019.

[20] M. Helbich, Y. Yao, Y. Liu, J. Zhang, P. Liu, and R. Wang, "Using deep learning to examine street view green and blue

spaces and their associations with geriatric depression in Beijing, China," *Environment International*, vol. 126, pp. 107–117, 2019.

[21] S. Wirges, T. Fischer, C. Stiller, and J. B. Frias, "Object detection and classification in occupancy grid maps using deep convolutional networks," *IEEE* in *Proceedings of the Intelligent Transportation Systems (ITSC)*, vol. 7, pp. 3530–3535, Maui, HI, USA, November 2018.

[22] N. Esquivel, O. Nicolis, B. Peralta, and J. Mateu, "Spatio-temporal prediction of Baltimore crime events using CLSTM neural networks," *IEEE Access*, vol. 8, pp. 209101–209112, 2020.

[23] F. Alidoost, H. Arefi, and F. Tombari, "2D image-to-3D model: knowledge-based 3D building reconstruction (3DBR) using single aerial images and convolutional neural networks (CNNs)," *Remote Sensing*, vol. 11, no. 19, p. 2219, 2019.

[24] Y. Yi, Z. Zhang, W. Zhang, C. Zhang, W. Li, and T. Zhao, "Semantic segmentation of urban buildings from VHR remote sensing imagery using a deep convolutional neural network," *Remote Sensing*, vol. 11, no. 15, p. 1774, 2019.