Heritage Science

RESEARCH Open Access

A detection method for the ridge beast based on improved YOLOv3 algorithm

Miaole Hou[1,2], Wuchen Hao[1,2], Youqiang Dong[1,2*] and Yuhang Ji[1,2]

Abstract

The ridge beast is a beast placed on the ridge of the roof of ancient Chinese buildings, not only has a decorative function, and has a strict hierarchical meaning, the number and form of the ridge beast placed on different levels of buildings are strictly limited. The detection technology of ridge beast decorative parts has important application value in the fields of fine 3D reconstruction of ancient buildings, historical dating and cultural and tourism services. Aiming at the problem of poor detection performance of traditional detection algorithms due to high texture similarity and poor discrimination of ridge beast, this paper proposed an improved YOLOv3 based detection algorithm for ridge beast decorative pieces. In terms of basic network improvement, local features are aggregated to the deep separable convolution internal embedding summation layer, and point convolution is used to connect the channel information of original features and aggregated features, so as to expand the receptive field and learn more diverse features. The residual structure of the feature extraction network was constructed by using the convolution, and the extraction effect of the model on the fine-grained features of the ridge beast was optimized, so that the detection accuracy was improved. In the prediction head improvement of the model, the original linear structure was reconstructed, and the extrusion and excitation modules were introduced to model the channel relationship of multi-scale feature map, which suppressed the response of interference signals and made the feature more directivity. The parallel 1×1 and 3×3 convolution are used to construct a multi-size convolution structure, which enhances the semantic information extraction ability of the model and further improves the detection effect. Experiments were conducted on the constructed ridge-beast dataset, and the results showed that the mAP of the improved algorithm can reach 86.48%, which is 3.05% higher than YOLO-v3, and the model parameters are reduced by 70%, which has a better detection performance and can provide a reference for the automated detection of ancient building components.

Keywords Ancient buildings, Ridge beast decorative parts, Deep learning, Object detection, YOLOv3

Introduction

Ancient architecture is a carrier of national wisdom and has rich artistic, cultural, scientific and emotional value. Ridge beasts are decorative elements placed on the ridge of ancient Chinese buildings, which have carried thousands of years of Chinese civilization. The number and types of ridge beasts vary among different historical periods or different levels of ancient buildings, and have important artistic and cultural values [1]. Ridge beast detection technology can automatically identify and locate the types and positions of ridge beasts in images, which can support the fine 3D reconstruction of ancient building roofs [2], Assist experts to identify the age of buildings and improve efficiency, as well as play a role in knowledge dissemination and cultural communication in cultural tourism service scenarios. Therefore, it is important to carry out research on automated detection

*Correspondence:
Youqiang Dong
dongyouqiang@bucea.edu.cn
[1] School of Geomatics and Urban Spatial Informatics, Beijing University of Civil Engineering and Architecture, Beijing 100044, China
[2] Key Laboratory for Architectural Heritage Fine Reconstruction & Health Monitoring, Beijing 100044, China

technology for ridge beast in the context of cultural heritage.

In recent years, object detection has been a challenging research direction in computer vision, which aims to determine whether there is a target instance of a known category in the image, and if it exists, to output the spatial location of the target instance in the original image and the corresponding category name, i.e., to locate the detected target instance with a rectangular border and to classify the detected target instance. Currently, target detection technology is often used in daily safety life, robot navigation, traffic scene detection, aerospace and other fields.

Traditional target detection algorithms generate a large number of candidate regions by means of sliding windows or image segmentation, followed by image feature extraction for each candidate region, such as (HOG, SIFT [3], Haar, etc.), and pass the extracted image features to classifiers (such as SWM, Adaboost, and Random Forest [4], etc.) to determine the class of candidate regions. However, there are limitations in the features extracted by traditional methods, and the method of generating candidate regions requires a large amount of computational overhead, and the detection accuracy and detection speed do not meet the requirements of practical applications, making the research on traditional target detection techniques a bottleneck. With the performance saturation of detection algorithms caused by manual feature extraction, target detection stagnated for some time after this until the emergence of AlexNet [5] network in 2012, making deep learning back into the mainstream, which far outperformed other algorithms in the ImageNet image classification task. Due to the excellent learning ability of deep convolutional networks to extract higher-level semantic features in images, researchers applied this learning ability to the task of target detection. Since then, deep learning-based target detection began to move forward at an unprecedented speed, and with the successive emergence of excellent networks such as VGGNet, ResNet, GoogleNet, and so on, the dominance of deep learning was gradually demonstrated in the fields of classification, object detection, and image segmentation, which greatly exceeded the level of traditional algorithms.

However, there is less research on applying deep learning target detection methods to the detection of ancient Chinese architectural components, which is currently in its infancy, and the ridge beasts are more similar in color, texture, and morphology, and traditional deep convolution has not been able to take into account the learning of similar target difference features, resulting in more false detections.

In the actual ridge beast detection scenario, external factors such as lighting, photography distance, photography perspective, and target size weaken the texture features of different ridge beasts in the image, which brings certain challenges to the detection of ridge beast decorative parts. The YOLOv3 algorithm has higher detection performance compared with other detection algorithms, so the YOLOv3 model is selected as the base structure, and the model is improved to propose a network model that is more suitable for detecting ridge beast decorative parts.

The main contributions of this paper are as follows:

1. For the ridge beast detection task, a ridge beast dataset is constructed.
2. In terms of backbone network, in order to optimize the extraction effect of fine-grained features of images while streamlining the number of model parameters, this paper optimizes the learning method of internal features by embedding a local summation function into the depth-separable convolution and obtains an improved depth-separable convolution. At the same time, the feature extraction network is constructed by combining this convolutional kernel to enhance the utilization of contextual information by the model and improve the detection accuracy.
3. In the feature fusion stage, the squeeze and excitation block (SE block) and the multi-size convolutional structure are fused as the prediction head to improve the extraction of semantic information.

The article is organized as follows: "Related works" Section provides an overview of previous and recent work on deep learning-based image target detection. In "Materials and methods" Section, the main data sources and methods of the work in this paper are described. In "Results and discussion" Section, we present our experimental setting as well as the experimental results, respectively. Finally, in "Conclusions and future works" Section we draw conclusions.

Related works

Deep learning is a neural network model with multiple layers of perception, featuring nonlinear fitting and adaptive feature extraction. In recent years, target detection methods based on deep learning have been widely used. The research directions of deep learning target detection mainly include two-stage detection methods and single-stage detection methods: two-stage detection (region generating convolutional neural network), by extracting feature information in the proposed region for classification and localization. In 2014 Girshick proposed R-CNN [6], the proposed algorithm provides a new approach for target detection and becomes the two the basis of two-stage detection algorithm. In 2015, He et al. proposed SPP- Net to solve the problem of repeated convolution in R-CNN and improve the detection speed.

In the field of cultural heritage, literature [7] used Faster-RCNN to detect dripping components and damaged hook-head components, and calculated the spacing of adjacent components to infer the number of missing components using the linear distribution characteristics of dripping. The literature [8] trained real-time detectors based on SSD-Mobilenet to achieve the detection of bird's-eye ornamentation. The literature built a Great Wall protection system consisting of three parts: data collection end, database and computing terminal based on mobile dense sensing (MCS) technology and deep learning model to achieve rapid detection of masonry damage.

The single-stage detection, represented by the YOLO series [9] algorithm, has the core idea of gridding the whole image, and the grid where the center of the target is located is responsible for predicting the target and locating the target position by means of regression. The literature [10] introduced the idea of spatial pyramid pooling to strengthen the network, improved the loss function based on the ratio of the center distance and diagonal distance between the two prediction frames and labeled frames, and solved the problem of invalid IOU loss caused by the non-overlap of the two by calculating the DIOU loss. The literature draws on the DenseNet idea to build inter-layer dense connectivity structures in the underlying network to varying degrees, enhancing feature extraction capabilities and information reuse. In the literature [11], an improved YOLOv3 detection algorithm is proposed, which uses a dual-path network as a feature extraction network and establishes four feature layers of different scales for multi-scale prediction to avoid the problems of information loss during network transmission and the lack of rich semantic feature extraction for small targets. The literature proposed SSD detection algorithm mainly balances the shortcomings of R-CNN series algorithm and YOLO series algorithm, SSD uses FPN (Feature Pyramid Networks, multi-scale feature map), adopts different size and number of pre-defined boxes for different size of convolutional layers, outputs the predicted true edges and category confidence respectively, and finally uses NMS (Non Maximum Suppression) to output the results. Slightly improving the detection of small targets in the graph, a series of methods such as DSSD, RSSD [12], and ASSD [13] have gradually emerged in order to continue optimizing the detection quality.

Materials and methods

Ridge beast dataset

According to the location and appearance of the ridge beast decorative parts, they can be divided into 14 categories: Immortals, Dragons, Phoenixes, Lions, Seahorses, Pegasus, Axolotls, Suan ni, Xie zhi, Dou niu, Xing shi, Chui shou, Wen shou, and Tao shou. Due to the vast distribution of ancient Chinese architecture, it is difficult to obtain images of the official, civil and religious styles and a full range of ridge beast decorations through aerial photography or field photography. However, some open source image libraries and search engines have accumulated rich image resources in the process of operating for a long time, so this paper uses Internet crowd-sourced data with the advantages of low acquisition cost to build a ridge beast dataset by crawling crowd-sourced images. Firstly, the images of ancient buildings were crawled from open source libraries and search engines by combining different keywords; secondly, the undamaged images in JPEG format were screened, and the images of historical buildings containing the decorative parts of ridge beast were selected by manual screening, see Fig. 1.

The ridge beast dataset contains a total of 13760 objects, which are randomly divided into a test set and a training set in the ratio of approximately 1:4, with 11049 images as the training set and 2711 images as the test set, as a way to perform parameter validation and training of the deep network model to avoid overfitting of the training model. Finally, the types and locations of the ridge beast targets were precisely marked on the images by using the annotation tool according to their distribution and appearance. Table 1 shows the distribution statistics of 14 types of ridge beast. Since ridge beast are mostly small-sized targets in the images, they are densely

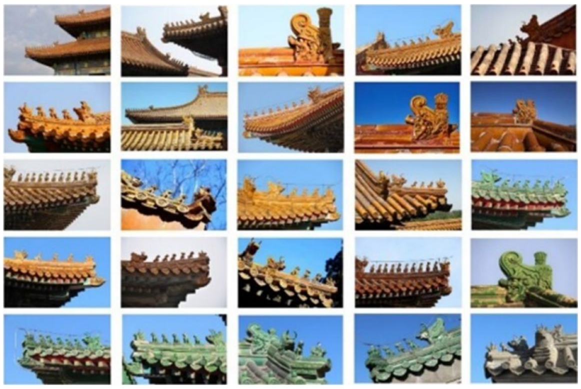
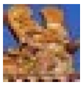


Fig. 1 Images of some unlabeled ancient buildings

Table 1 Statistical table of the distribution of ridge beasts

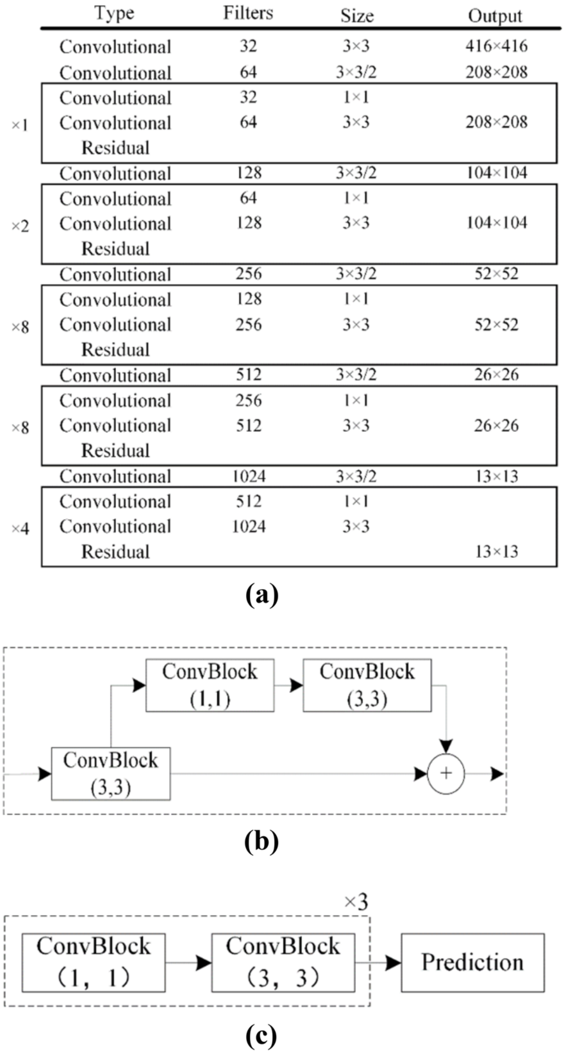
Name	Immortals	Dragons	Phoenixes	Lions	Seahorses	Pegasus	Axolotls
Test	272	252	235	234	192	187	115
Train	1070	964	934	907	776	720	505
Image							
Name	**Suan ni**	**Xie zhi**	**Dou niu**	**Xing shi**	**Chui shou**	**Wen shou**	**Tao shou**
Test	118	80	82	60	325	330	229
Train	475	363	369	292	1406	1328	940
Image							

arranged. The color of the ridge beast is close to the roof, and the geometric form and surface texture are similar and the Internet images vary greatly in resolution and clarity, making the detailed features of the ridge beast a key factor affecting the detection results.

Deep neural network model

The core idea of YOLOv3 is a regression-based cell prediction grid. The overall structure of the model consists of three parts: the DarkNet-53 base network, the upsampling layer and the prediction head (YOLO-Head) with three branching structures [14]. In this paper, we present YOLOv3 as well as the loss function of the model and the improved YOLOv3 proposed in this paper, and experimentally validate it.

DarkNet53 is used to extract image depth features and is a feature extraction network consisting of 53 convolutional layers, each consisting of a 2D convolutional operator, a Batch Normalization (BN) layer and a Leaky Relu excitation function. As shown in Fig. 2a, the underlying network uses a total of five convolutional layers with a step size of 2 to downsample the feature maps, and the feature map resolution is reduced to 1/2 for each downsampling while the number of channels is expanded by a factor of two. To solve the problems of traditional convolutional neural networks, such as network degradation with increasing depth, the network connects several residual modules after each downsampled convolutional layer to maintain a linear mapping by learning the residuals of potential mappings between layers. The structure of the residual module is shown in Fig. 2b, including two series-connected 1×1 and 3×3 convolutional layers, which form a hopping layer connection structure by summing the input tensor with the output tensor of the convolutional layer. The number of residual modules



	Type	Filters	Size	Output
	Convolutional	32	3×3	416×416
	Convolutional	64	3×3/2	208×208
×1	Convolutional	32	1×1	
	Convolutional	64	3×3	208×208
	Residual			
	Convolutional	128	3×3/2	104×104
×2	Convolutional	64	1×1	
	Convolutional	128	3×3	104×104
	Residual			
	Convolutional	256	3×3/2	52×52
×8	Convolutional	128	1×1	
	Convolutional	256	3×3	52×52
	Residual			
	Convolutional	512	3×3/2	26×26
×8	Convolutional	256	1×1	
	Convolutional	512	3×3	26×26
	Residual			
	Convolutional	1024	3×3/2	13×13
×4	Convolutional	512	1×1	
	Convolutional	1024	3×3	
	Residual			13×13

Fig. 2 YOLO-v3 structure chart. (**a** DarkNet53 network structure **b** The structure of the residual module, **c** YOLO-Head structure)

used before each downsampled convolutional layer is 1, 2, 8, 8, and 4, in that order.

The upsampling layer is used for stitching feature maps of different sizes to realize the construction of feature pyramids. The interpolation upsampling operation is mainly used to amplify the resolution of low-size feature maps, so that they can be channel-connected with large-size feature maps to achieve improved detection of multi-scale targets.

YOLO-Head is used to fuse multi-scale features for the extraction of semantic information such as location and category. YOLOv3 uses a 3-head structure for prediction of target information, and the YOLO-Head structure is shown in Fig. 2c, which uses three sets of alternating 1×1 and 3×3 convolutions to fuse and extract semantic information from the multi-scale feature maps. Finally, each prediction end outputs 13×13×3×(4+1+C), 26×26×3×(4+1+C) and 52×52×3×(4+1+C) 3 prediction tensor, which is processed by position back-calculation and non-maximal value suppression algorithm to finally obtain the optimal detection results.

The loss function is the algebraic sum of coordinate loss ($Loss_{coord}$), confidence loss ($Loss_{IOU}$)) and classification loss ($Loss_{class}$); coordinate loss, i.e., localization error, including central coordinate error and dimensional error; confidence loss, i.e., existence error of the target; and classification loss, i.e., class error. The loss function is as follows:

$$Loss_{coord} = \sum_{i=0}^{S\times S}\sum_{j=0}^{3} I_{ij}^{obj}(2 - w_i \times h_i)\times\left\{\left[\widehat{x}_i log(x_i) + \left(1-\widehat{x}_i\right)log\left(1-\widehat{x}\right)\right] + \left[\widehat{y}_i log\left(y_i\right) + \left(1-\widehat{y}_i\right)log\left(1-\widehat{y}\right)\right]\right\}$$

$$+0.5 \times \sum_{i=0}^{S\times S}\sum_{j=0}^{3} I_{ij}^{obj}(2 - w_i \times h_i) \times \left[\left(w_i - \hat{w}_i\right)^2 + \left(h_i - \hat{h}_i\right)^2\right] \ (3-1)$$

$$Loss_{IOU} = -\left\{\sum_{i=0}^{S\times S}\sum_{j=0}^{3} I_{ij}^{obj}\left[\hat{C}_i log(C_i) + \left(1-\hat{C}_i\right)log(1-C_i)\right] + \sum_{i=0}^{S\times S}\sum_{j=0}^{3} I_{ij}^{noobj}\left[\hat{C}_i log(C_i) + \left(1-\hat{C}_i\right)log(1-C_i)\right]\right\} \ (3-2)$$

$$Loss_{class} = -\sum_{i=0}^{S\times S} I_i^{obj} \sum_{C\in classes}\left[\widehat{p_C} log(p_C) + \left(1-\widehat{p_C}\right)log(1-p_C)\right] \ (3-3)$$

$$Loss = Loss_{coord} + Loss_{IOU} + Loss_{class} \ (3-4)$$

where $Loss_{class}$ is category loss, $Loss_{coord}$ is coordinate loss, $Loss_{IOU}$ is confidence loss. I_{ij}^{obj} means whether the jth anchor frame in the ith grid is responsible for the predicted target, and I_{ij}^{obj} means whether the target exists in the ith grid. $\widehat{x}_i$, $\widehat{y}_i$, $\widehat{w}_i$, $\widehat{h}_i$ are the ridge beast The center coordinates and dimensional truth values of the potential region, $\widehat{C}$ is the confidence truth value, and x_i, y_i, w_i, h_i, C_i are the predicted values. To prevent the positioning error from accounting for too large a proportion, a correction term $(2 - w_i \times h_i)$ is used to constrain and a weight of 0.5 is assigned to the dimensional error to limit the positioning loss.

Improved methods

The detection algorithm of ancient architectural ridge beast decorative parts needs to achieve accurate recognition of ridge beast in a complex environment and reduce the size of the model by optimizing the YOLOv3 backbone. The proposed target detection model of ridge beast decorative parts is shown in Fig. 3, and the overall structure inherits the main framework of YOLOv3. This study aims to improve the base network (DPC-DarkNet), including the base network, and the multi-size convolutional prediction head (SE-MYOLO-Head) incorporating the attention mechanism, so as to increase the accuracy of target detection, improve the detection speed, and reduce the network parameters.

DPC-DarkNet

In real ridge beast detection scenarios, factors such as long photographic distance, similar shape of ridge beast targets, and small size weaken the texture features of different ridge beasts in the images. The traditional YOLOv3 algorithm uses DarkNet53 feature extraction network, which uses residual network to change the feature map dimension by standard

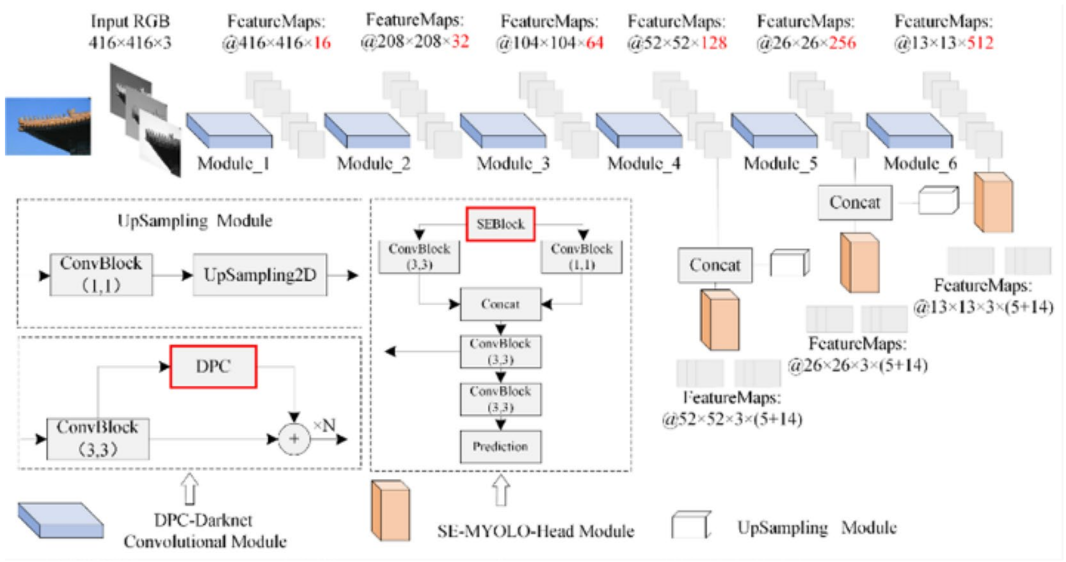


Fig. 3 Improved network structure

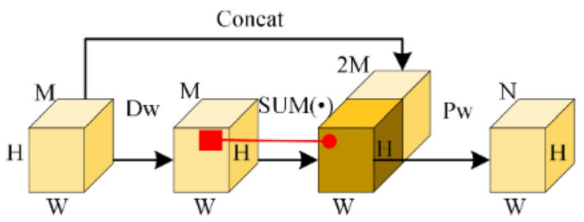


Fig. 4 Schematic diagram of deep aggregation convolution

convolution kernel to form a bottleneck network [15]. However, this approach not only has a huge number of parameters, but also easily causes information loss, which leads to certain limitations of the neural network in expressing fine-grained features of ridge beast, and then affects the detection accuracy of ridge beast. In order to reduce the training parameters and improve the ability of the underlying network to extract fine-grained texture features, this paper enhances the ability of the network to learn differential features based on the idea of feature aggregation, and proposes the Depthwise Polymetric Convolution (DPC) by improving the depth-separable convolution with local summation functions, and uses this convolution to build the DPC-DarkNet feature extraction network.

The DPC structure is shown in Fig. 4, and for depth-separable convolution, the internal features are aggregated using a local summation function to expand the convolution field of perception and enhance feature discrepancy. The input features are then stitched with the aggregated features, and the original single-pixel features are fused with their neighborhood features using point convolution to learn the image depth aggregated features. The specific computational steps are as follows.

For an input tensor T_{input} of type $H \times W \times M$, apply a 3×3 depth aggregation convolution of sliding step size 1 and number N to it.

① Compute 2D convolution on each of the M channels of T_{input} independently using the filter set D_w to obtain T_d.
② Using a 3×3 sliding window, the SUM() summation function is applied to each center pixel on M channels of T_d to calculate the algebraic sum of the elements in the range to obtain T_{sum}. Finally, the input tensor T_{input} is stitched with T_{sum} to obtain T_{concat}.
③ Compute T_{concat} point convolution depth aggregation features along the depth direction using filter P_w.
④ Repeat step ③ N times to obtain the output tensor T_{output} of type $H \times W \times N$.

SUM() in step ② is the summation function, if the matrix$\boldsymbol{A} = (a_{ij})_{m \times n}$, then $SUM(\boldsymbol{A}) = \sum_{j=1}^{n}\sum_{i=1}^{m} a_{ij}$. As shown in Fig. 4, DPC uses the summation function to aggregate D_w convolutional features without introducing additional parameters, but is able to combine features from a larger range of regions, thus increasing the mobility of information across channels and the variability of depth features. Deep aggregation convolution enables the neural network to learn deep aggregated

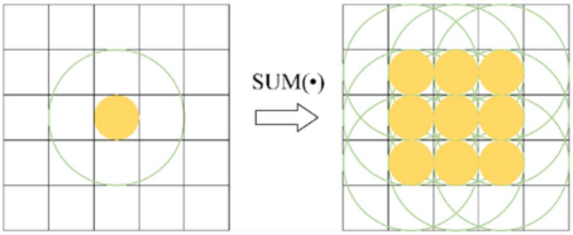


Fig. 5 Feel the process of wild expansion

features, with the in-kernel point convolution weight coefficients determining what local signals are combined, which can enhance the extraction of fine-grained features to some extent.

In contrast to the traditional convolution operation in which each output channel is convolved with each channel of the input, the number of parameters needed is the product of the number of input channels and the number of output channels. In contrast, in deep Convergence convolution, deep convolution is first performed, i.e., a separate convolution kernel is applied to each input channel so that an intermediate feature map with the same number of input channels can be obtained. Then, in the aggregated convolution, a smaller convolution kernel is used to convolve the intermediate feature maps to generate the final output feature maps, thus greatly simplifying the number of training parameters. Secondly, this network structure does not require a bottleneck structure, which can avoid feature loss, as shown in Fig. 5. The sensory field is expanded, which improves the utilization of contextual information; the local features are aggregated, which makes the features more differentiated; and the original features of the input can be retained, which suppresses the aggregation of useless information.

In fact, there are two extreme cases of using DPC to construct residual structures. The first one is that the point convolution discards features from the depth convolution part, when the traditional residual network structure becomes a special form of DPC residual. The other is that the point convolution only extracts the features from the depth convolution part, and the network can still learn the depth aggregation features. In general, the point convolution in DPC will fuse both single-pixel information and contextual information. Therefore, DPC-DarkNet can optimize the extraction of fine-grained features from images.

SE-MYOLO-head

In the feature fusion stage, the YOLOv3 algorithm fuses multi-scale features by constructing a feature pyramid. Information such as location and category of the target is encoded in the channel in the form of anchor frames, so that the deep-level features near the output have more semantic information. Shallow-level features are directly derived from the underlying network and possess more location, color and texture features, and the feature fusion method is crucial. The original YOLO-Head is a linear structure, and when fusing multi-scale features, the 1×1 and 3×3 convolutional groups in series are directly used without considering the feature correlation in different region ranges, which corrupts the semantic information.

Based on this, this paper reconstructs the linear prediction head of YOLOv3 with the structure shown as SE-MYOLO-Head in Fig. 2. Firstly, the multi-scale feature map channel relations are modeled using the SE module. the SE module (Fig. 6) is a kind of channel domain attention, which first performs pooling extrusion on channel global features to generate channel descriptors. The channel descriptors are then computed and passed through a fully connected neural network to model the interrelationships between channels, which are motivated as channel weight coefficients that can express relative importance. Finally, feature reassignment is achieved by matrix number multiplication. The SE module finally uses a sigmoid excitation function to map each weight coefficient to a scaling factor in the (0, 1) interval, thus enabling suppression of the response of interfering signals and making the features more directional.

To better fuse information from different region ranges, the prediction head proposed in this paper, after the SE module, uses a multi-size convolution structure, i.e., standard convolution of 1×1 and 3×3 in parallel to fuse multi-scale features. Then the tensor obtained by splicing the two operations is used in series to further extract semantic information using two standard convolutions with filter numbers c and 2c, respectively. The computed features of the former are fed to the up-sampling convolution module (see UpSampling Module in Fig. 2), while the computed features of the latter are passed through the 2D convolution layer with bias Convolution layer with bias to predict the output tensor.

Results and discussion

Experimental environment

In this paper, the Pascal VOC [16] criterion was used to calculate the mean Average Precision (mAP) to

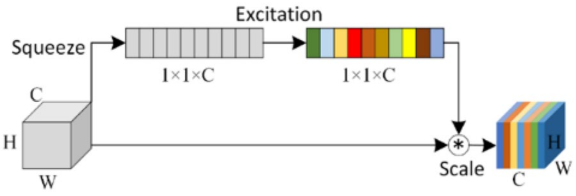


Fig. 6 SE Structure

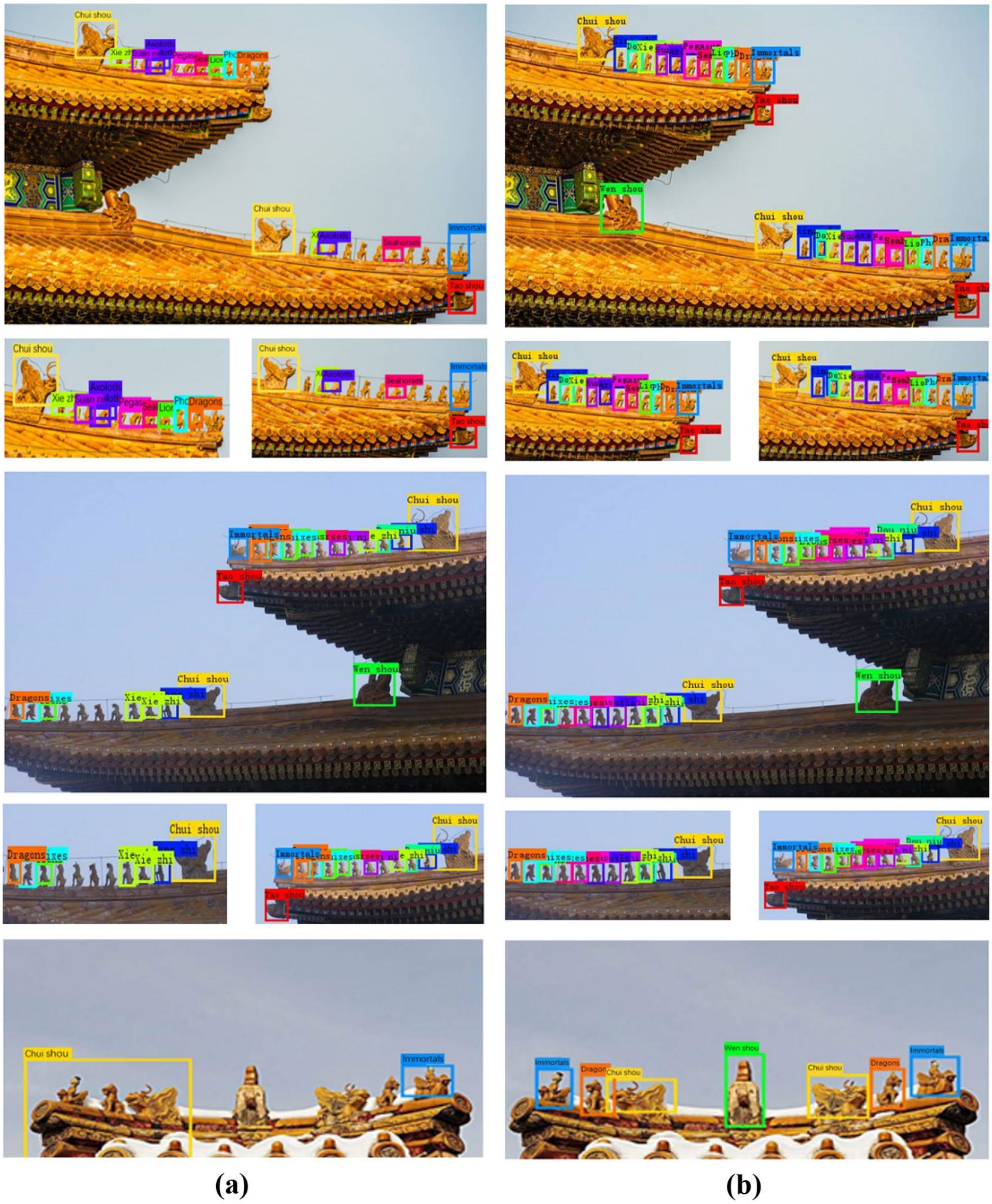


Fig. 7 Comparison of the detection effect of YOLO-v3 and improved YOLO-v3 (**a** YOLOv3 detection effect **b** improved YOLOv3 detection effect)

evaluate the detection effectiveness of the algorithm on the dataset.

The experiments were conducted in NVIDIA GeForce RTX 2080Ti, 11 GB; Python3.7; Tensorflow2.3 environment. 9 anchor frames with dimensions generated by K-Means clustering: (14, 17), (21, 24), (28, 32), (35, 42), (46, 52), (64, 64), (86, 88), (139, 152), (327, 354). Optimization was performed using Adam optimizer with initial learning rate set to 0.0001, beta_1 to 0.9, beta_2 to 0.999, learning rate decay rate (Decay Rate) to 0.5, and training 100 rounds (Epoch). The input image size parameter is 640×640, and the scaling method is used to unify the

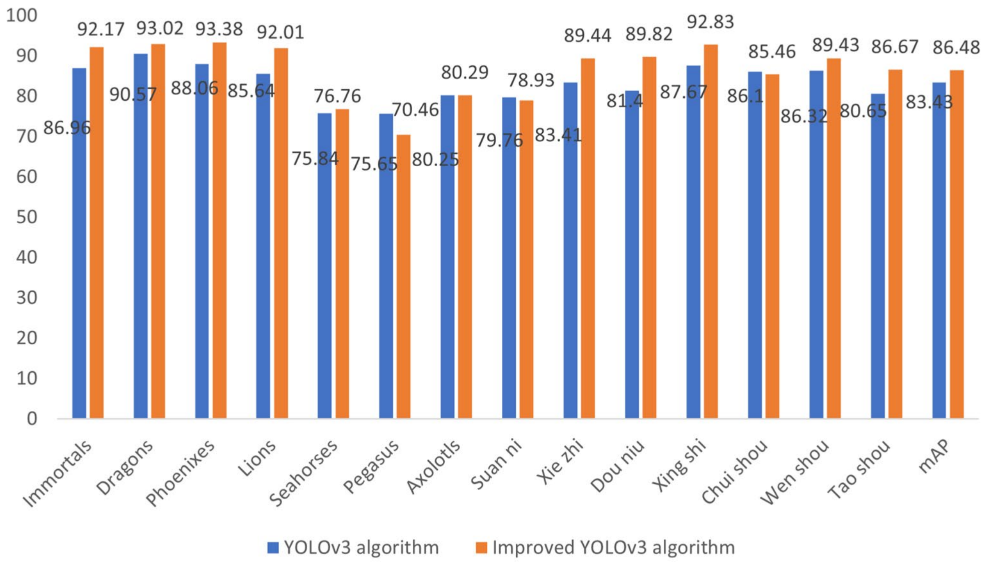


Fig. 8 AP values for testing of 14 kinds of ridge beast decorative parts

image with irregularities. The specific process is to calculate the scaling factor with 640 using the longest edge of the image as the reference edge for image scaling, and then use RGB (128, 128, 128) to fill the image into a regular square.

Detecting results

In this paper, the detection results of the algorithm are tested on the ridge beast test set. Figure 7 compares the detection results of YOLOv3 with improved YOLOv3 on typical images of ridge beast decorations, Fig. 8 shows the detection accuracy of 14 types of ridge beast decorations using improved YOLOv3, Fig. 9 compares the detection performance of various algorithms in terms of accuracy, time consumption, and number of model parameters, respectively, and Fig. 10 shows the final results of detection using improved YOLOv3.

Comparing Fig. 7a, b it can be seen that when in the longer distance photography condition, the ridge beast parts in the image are mostly presented in small and dense form, and at this time there are more missed and wrong detections in YOLOv3, and the bounding box appears too wide, too narrow and multiple objects in the same box. The improved YOLOv3 can better detect the various types of decorative parts of the beasts in the figure, and the location of the corner points of the detection frame is more accurate. The red circle marked in the figure is at the side view angle of the kissing animal, which has never appeared in the training set, but the improved YOLOv3 in this paper identifies it correctly. It shows that the algorithm in this paper can learn more differentiated features using deep aggregated convolution, and is more effective in extracting semantic information of images.

The comparison in Fig. 8 shows that, relative to the YOLOv3 algorithm, the detection accuracy of the 14 decorative components of the ridge beast test set using the improved YOLOv3 is improved for 12 of the categories, and the mAP can reach 86.48%, surpassing the YOLOv3 algorithm by 3.05%.

As can be seen from Fig. 9, in terms of detection accuracy, the improved YOLOv3 still has the best detection accuracy, followed by the two-stage detection algorithm Faster-RCNN; in terms of detection speed, Tiny_YOLO takes the absolute advantage of 17.43 ms per frame, but loses more accuracy, Faster-RCNN takes the longest time of 247.68 ms per frame, while The improved YOLOv3 is not significantly different from the original YOLOv3 at 80.67 ms per frame, which is due to the redundant floating point computation when constructing the summation function in this paper, and the speed should be better than YOLOv3 in theory; in terms of the number of trainable parameters of the model, the improved YOLOv3 is only 33% of YOLOv3. Figure 10 shows the final effect of detection using the improved YOLOv3 algorithm.

Discussion

The improvement strategies in this paper include the DPC-Darknet base network improved based on deep aggregated convolution, and the SE-MYOLO-Head prediction head incorporating the attention mechanism, and model decomposition experiments were conducted to test the accuracy gains produced by each improvement

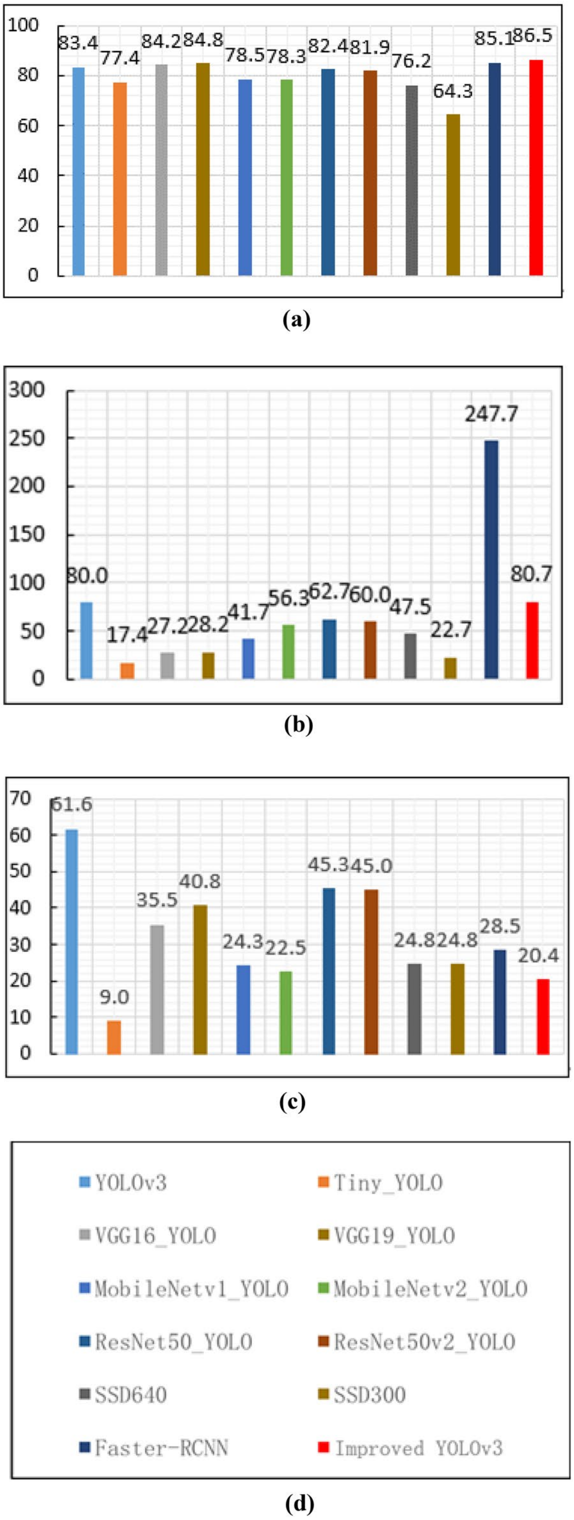


Fig. 9 Comparison of detection performance of different algorithms (**a** mAP/%; **b** elapsed time/ms; **c** number of parameters/M; **d** legend)

strategy. As shown in Table 2, the following four models were compared under the same conditions: Model A is the traditional YOLOv3 model using Darkent-53 base network with YOLO-Head prediction head; Model B uses Darknet-53 as the base network and SE-MYOLO-Head as the prediction head; Model C uses DPC-Darknet as the base network, YOLO-Head as the predictor head; Model D is the modified YOLOv3 in this paper, using DPC-Darknet as the base network and SE-MYOLO-Head as the predictor head.

As can be seen from Table 2, Model D, as the improved model proposed in this paper, has a significant improvement in accuracy compared with the other models. In order to be able to verify the effects of the DPC-Darknet base network and the SE-MYOLO-Head prediction head proposed in this paper on the model, the effectiveness of the module improvement is reflected by comparing the detection accuracy of different models. By comparing Model A with Model B and Model C with Model D, the accuracy is improved by 2.27% and 0.74%, respectively, which shows that the SE-MYOLO-Head prediction head proposed in this paper can better fuse the semantic information and thus improve the detection accuracy. Similarly, by comparing Model A with Model C and Model B with Model D, the accuracy is improved by 2.31% and 0.78%, respectively. It can be seen that the Darkent-53 base network proposed in this paper can not only avoid the loss of information, but also improve the utilization of context, and further improve the detection accuracy. In summary, the improved YOLOv3 model in this paper has superior detection performance.

Conclusions and future works

In order to solve the problem of accurate recognition of ancient architectural ridge beast decorative parts, this paper improves the YOLO-v3 model and produces a ridge beast dataset for testing. A local summation function is introduced to improve the internal construction of the traditional depth-separable convolution, and the convolutional field is increased to enable the network to learn deep aggregated features. The DPC-DarkNet feature extraction network constructed based on this convolutional kernel has better extraction results for fine-grained features of ridge beast. By fusing the existing SE modules and reconfiguring the linear YOLO-Head into a multi-size convolutional structure, the neural network is able to extract rich semantic features. The mAP of this paper's algorithm is 86.48%, which can identify the small-sized and dense ridge beast decorative parts of the image more accurately and has real-time performance,

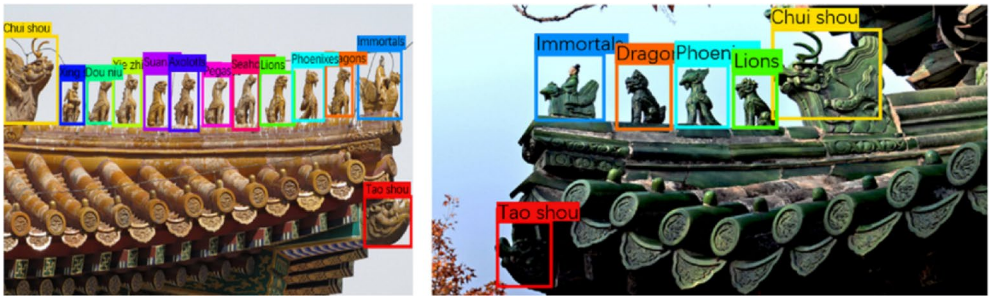


Fig. 10 Improved detection effect of YOLOv3

Table 2 The comparison on different methods

	Model A	Model B	Model C	Model D
Darknet-53	√	√	×	×
DPC-Darknet	×	×	√	√
YOLO-Head	√	×	√	×
SE-MYOLO-Head	×	√	×	√
mAP/%	83.43	85.70	85.74	86.48

which can provide a reference meaning for the automatic interpretation of ancient architecture images.

The quality of the training samples also affects the generalization ability of the model. Usually, images with higher resolution and clarity are used for training, which is beneficial to improve the detection effect of the model in real scenes. Therefore, in future research work, multi-sensor data sources such as cell phones, cameras, and drones will be combined to improve the quality of the data set for the detection application of real scenes; in addition, ridge beast are decorative parts with certain rules, and their spatial arrangement order is related to the category and have obvious form characteristics. The next step will be to integrate the knowledge of ancient building construction to study the detection model of ridge beasts with better generalization ability.

Acknowledgements
Thanks to the support of the Key Laboratory of Fine Reconstruction and Health Monitoring of Architectural Heritage.

Author contributions
MH conceived the presented idea and put forward experimental suggestions. WH conducted and refined the analysis process and wrote the manuscript. All authors approved the final manuscript.

Funding
This research was supported by the National Key R&D Program Project. (2019YFC1520800) 、High-level teaching in Beijing municipal universities Division Team Building Support Program Great Wall Scholar Training Program (CIT&TCD20180322) 、Scientific research project of Beijing Municipal Education Commission. (KM202110016005) 、Natural Science Foundation of Beijing Municipality Purpose—Joint Fund Project of the Municipal Education Commission Purpose (KZ202110016021), Beijing University of Civil Engineering and Architecture graduate innovation Project (PG2020079).

Availability of data and materials
All data generated or analyzed during this study are included in this published article.

Declarations

Competing interests
The authors declare that they have no competing interests.

Received: 27 February 2023 Accepted: 9 July 2023
Published online: 08 August 2023

References

1. Wu S. On chi wei beast roof ridge decorations of the Bohai state. Asian Archaeol. 2022;5(1–2):1–9.
2. Huo PP, Hou ML, Dong YQ, et al. A method for 3D reconstruction of the Ming and Qing official-style roof using a decorative components template library[J]. ISPRS Int J Geo Inf. 2020;9(10):570–90.
3. Lowe DG. Distinctive image features from scale-invariant keypoints. Int J Comput Vision. 2004;60:91–110.
4. Liaw A, Wiener M. Classification and regression by randomForest. R news. 2002;2(3):18–22.
5. Krizhevsky A, Sutskever I, Hinton GE. Imagenet classification with deep convolutional neural networks. Commun ACM. 2017;60(6):84–90.
6. Ren S, He K, Girshick R, Sun J. Faster r-cnn: towards real-time object detection with region proposal networks. Adv Neural Inf Process Syst. 2015;2015:28.
7. He K, Zhang X, Ren S, Sun J. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Trans Pattern Anal Mach Intell. 2015;37(9):1904–16.
8. Zou Z, Zhao X, Zhao P, Qi F, Wang N. CNN-based statistics and location estimation of missing components in routine inspection of historic buildings. J Cult Herit. 2019;38:221–30.
9. Huang YQ, Zheng JC, Sun SD, Yang CF, Liu J. Optimized YOLOv3 algorithm and its application in traffic flow detections. Appl Sci. 2020;10(9):3079.
10. Tian Y, Yang G, Wang Z, Wang H, Li E, Liang Z. Apple detection during different growth stages in orchards using the improved YOLO-V3 model. Comput Electron Agric. 2019;157:417–26.
11. Chen J, Wang Z, Wu J, Hu Q, Zhao C, Tan C, Luo T. An improved Yolov3 based on dual path network for cherry tomatoes detection. J Food Process Eng. 2017;44(10):e13803.
12. Jeong J, Park H, Kwak N. Enhancement of SSD by concatenating feature maps for object detection. arXiv Preprint. 2017. https://doi.org/10.4855/arXiv.1701.06659.
13. Yi J, Wu P, Metaxas DN. ASSD: attentive single shot multibox detector. Comput Vis Image Underst. 2019;189:102827.

14. Deng L, Li H, Liu H, Gu J. A lightweight YOLOv3 algorithm used for safety helmet detection. Sci Rep. 2022;12(1):10981.
15. Hurtik P, Molek V, Hula J, Vajgl M, Vlasanek P, Nejezchleba T. Poly-YOLO: higher speed, more precise detection and instance segmentation for YOLOv3. Neural Comput Appl. 2022;34(10):8275–90.
16. Everingham M, Eslami SA, Van Gool L, Williams CK, Winn J, Zisserman A. The pascal visual object classes challenge: a retrospective. Int J Comput Vision. 2015;111:98–136.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.