International Journal of Geo-Information

MDPI

Article

Recognition and Classification of Typical Building Shapes Based on YOLO Object Detection Models

Xiao Wang *, Haizhong Qian, Limin Xie, Xu Wang and Bohao Li

Institute of Geospatial Information, Information Engineering University, Zhengzhou 450001, China; qianhaizhong2005@163.com (H.Q.); gis_xlm@163.com (L.X.); w15822390883@outlook.com (X.W.); lbh030508@outlook.com (B.L.)

* Correspondence: xiaowang0209@outlook.com

Abstract: The recognition and classification of building shapes are the prerequisites and foundation for building simplification, matching, and change detection, which have always been important research problems in the field of cartographic generalization. Due to the ambiguity and uncertainty of building shape outlines, it is difficult to describe them using unified rules, which has always limited the quality and automation level of building shape recognition. In response to the above issues, by introducing object detection technology in computer vision, this article proposes a building shape recognition and classification method based on the YOLO object detection model. Firstly, for different types of buildings, four levels of building training data samples are constructed, and YOLOv5, YOLOv8, YOLOv9, and YOLOv9 integrating attention modules are selected for training. The trained models are used to test the shape judgment of buildings in the dataset and verify the learning effectiveness of the models. The experimental results show that the YOLO model can accurately classify and locate the shape of buildings, and its recognition and detection effect have the ability to simulate advanced human visual cognition, which provides a new solution for the fuzzy shape recognition of buildings with complex outlines and local deformation.

Keywords: cartographic generalization; building shape recognition; building classification; object detection; YOLO

check for updates

Citation: Wang, X.; Qian, H.; Xie, L.; Wang, X.; Li, B. Recognition and Classification of Typical Building Shapes Based on YOLO Object Detection Models. *ISPRS Int. J. Geo-Inf.* **2024**, *13*, 433. https://doi.org/10.3390/ijgi13120433

Academic Editors: Wolfgang Kainz and Florian Hruby

Received: 14 September 2024
Revised: 11 November 2024
Accepted: 12 November 2024
Published: 2 December 2024

Copyright: © 2024 by the authors. Published by MDPI on behalf of the International Society for Photogrammetry and Remote Sensing. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Buildings are one of the most important types of geographic entities in large-scale maps and occupy a significant amount of map load [1]. As the scale becomes smaller, buildings require generalization operations to adapt to the changes in the map range. The commonly used building generalization operators include selection, merging, simplification, typification, and displacement, among which the simplification operator is the most frequently used in building generalization [2–5]. One of the basic principles is that the shape characteristics of the building should be preserved before and after simplification; therefore, the accurate recognition and classification of building shapes is a prerequisite and an important basis for building generalization. Building shape also plays a key role in studies of urban building functional and semantic classification [6–10], building similarity and complexity measurement [11–14], and building pattern detection [15]. Meanwhile, accurately identifying different types of building shapes also has significant importance and roles in spatial data matching, change detection, and other aspects [16].

The recognition and classification of building shapes can be divided into two strategies: traditional geometric methods and artificial intelligence methods. Within the traditional geometric methods, there are mainly two approaches: One is the shape description method, which involves designing morphological indices to describe the geometric feature structure of buildings. For example, using Fourier transforms to describe shape characteristics [17,18] and employing NW and SW algorithms from DNA sequence alignment to measure building

shapes [19]. Another one is the template matching method: The main idea here is to summarize common, typical building shapes into a series of shape templates. By measuring the similarity between buildings and shape templates, the specific shape type of the building can be determined [20–23].

Since building shape recognition belongs to the realm of cognitive fields, it inherently possesses characteristics of fuzziness, as well as strong subjectivity and uncertainty [24]. Therefore, building shapes are difficult to describe completely in a unified form. When using traditional geometric methods for building shape recognition, both the shape description methods and template matching methods largely rely on shape geometry and similarity calculation, which can deal with buildings with common and regular shapes. For buildings with complex shapes and irregular outlines, which brings more interference for the recognition, these methods cannot recognize the building shape with the overall structure, resulting in the wrong classification.

With the development of artificial intelligence technology in recent years, related intelligent models and algorithms have been introduced into the research of building shape recognition and classification. The main approach is to use relevant models from machine learning and deep learning by collecting training datasets for building shapes and training the models to eventually have the ability to detect building shapes. For example, Ma et al. used self-supervised machine learning methods to measure building shapes [25]. Liu et al. achieved high-precision recognition of building shapes by using a Fourier shape descriptor that integrates multiple shape feature parameters as a neural network recognizer [26]. Jiao et al. used the AlexNet convolutional neural network from deep learning to classify buildings [27]. Yan et al. introduced an autoencoder learning method from deep learning to describe building shape features [28]. Yu et al. extracted multiple features from the contours of areal residential features to obtain corresponding graphic representations and used graph convolutional neural networks for multi-round extraction and aggregation to recognize residential feature types [29]. Yan et al. proposed a building shape adaptive simplification method using a graph convolutional autoencoder network [30]. Liu et al. adopted a deep point convolutional network to recognize building shapes, which can directly execute on the building nodes without constructing the graphs [31]. They also proposed a relation network method addressing building shape recognition with few labeled samples [32].

The above-mentioned methods have significantly promoted the intelligence and automation level of building shape recognition. However, there are still some issues, such as the traditional machine learning methods primarily relying on shallow learning, and shallow-structure models may not meet the requirements for feature extraction, as it requires complex feature engineering settings and high demands for experience and domain knowledge. Convolutional neural network methods in deep learning are mainly aimed at the classification of entire images. The recognition process requires eliminating interference from surrounding buildings, making it difficult to identify buildings with corresponding shapes from complex backgrounds, which increases the complexity of practical applications. The core of the graph convolutional neural network method is to represent building shape features using a graph structure, which essentially involves feature extraction of building shapes. The buildings with complex contour shapes and shape deformations may still face the same issues as traditional geometric description methods. Therefore, further exploration and attempts to find new intelligent methods for building shape recognition are still necessary and beneficial.

Object detection is a classic research topic in computer vision and image processing, the purpose of which is to identify all the objects of interest from an image and determine their categories and locations based on the image [33]. With the development of artificial intelligence and deep learning technologies, the performance of object detection techniques has significantly improved and has been widely applied in various fields such as autonomous driving [34], video surveillance [35], remote sensing image interpretation [36], medical detection [37], and product quality inspection [38]. The essence of building shape recognition is also to identify buildings with specific shape types from building data and

determine their types. Therefore, object detection and building shape recognition share the same mechanism, both essentially involving the discovery of specific targets. In summary, although object detection has been widely applied in multiple disciplines and research fields, it has not been fully utilized in the field of cartographic generalization, specifically the recognition and classification of building shapes in this study.

The YOLO (You Only Look Once) model is the representative of theone-stage detector in object detection. It is a preferred object detection model due to its small architecture, fast speed, and strong generalization ability, which can directly output the position and category of the bounding box through the neural network [39]. This paper introduces object detection methods for building shape recognition and classification, proposing a method for building shape recognition and classification based on the YOLO model, which can not only identify the specific type of building shapes but also accurately detect the locations.

2. Typical Building Shape

In this paper, the types of building shapes to be recognized are all typical shapes. Through the inspection and analysis of a large amount of building data, the recognition of the building types shown in Figure 1 is considered, which mainly includes the letter-shaped buildings such as E-like, F-like, H-like, L-like, T-like, Y-like, Z-like, and cross-like. Other shape types of buildings, such as basic graphic elements and special types, are not involved in this paper. Basic graphic element buildings mainly include rectangles, squares, parallelograms, trapezoids, circles, etc. These types of building shapes are relatively regular and can be easily recognized by geometric methods. Moreover, the shapes of such buildings are relatively simple, so in the cartographic generalization, selection and merging are the main operators, with less involvement in the simplification operator. Therefore, shape recognition research generally does not involve the recognition of basic graphic element buildings. For other buildings with special or more complex types, such as numerical shapes, they are relatively rare in the dataset. It is difficult to form a certain number of training datasets. If there is a need to recognize a specific type of special-shaped building, it will only be necessary to increase the corresponding type of building in the training process.

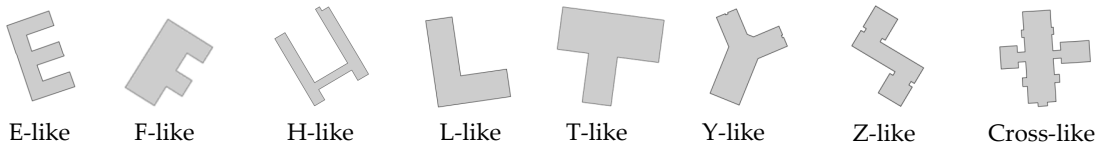


Figure 1. Typical types of building shapes.

3. YOLOv9 Integrating Attention Mechanism

YOLO is a typical representative of the one-stage object detection model [40] and was first proposed in 2016 [41]. It refers to the ability to identify the category and location of objects in an image with just a single pass. YOLO is known as a region-free (non-proposal) method. Compared to region-based methods in two-stage detection, YOLO does not need to pre-identify regions where objects might exist. From 2016 to 2024, YOLO has evolved through eleven versions [42–46]. This paper utilizes YOLOv9, one of the latest YOLO models, to recognize and classify building shapes and compares its detection performance with two significant updated versions of the series, YOLOv5 and YOLOv8.

3.1. YOLOv9 Network Structure

YOLOv9 is a significant iteration of the YOLO series by Chien-Yao Wang et al. [46]. As part of the Yolo family in object detection, Yolov9 builds upon its predecessors by incorporating novel architectural enhancements and optimization techniques, thereby improving both detection accuracy and computational efficiency. YOLOv9 achieves this balance by leveraging a deep convolutional neural network (CNN) architecture that enables the simultaneous prediction of bounding boxes and class probabilities in a single forward

pass, thus streamlining the detection process. The network architecture of YOLOv9 is shown in Figure 2.

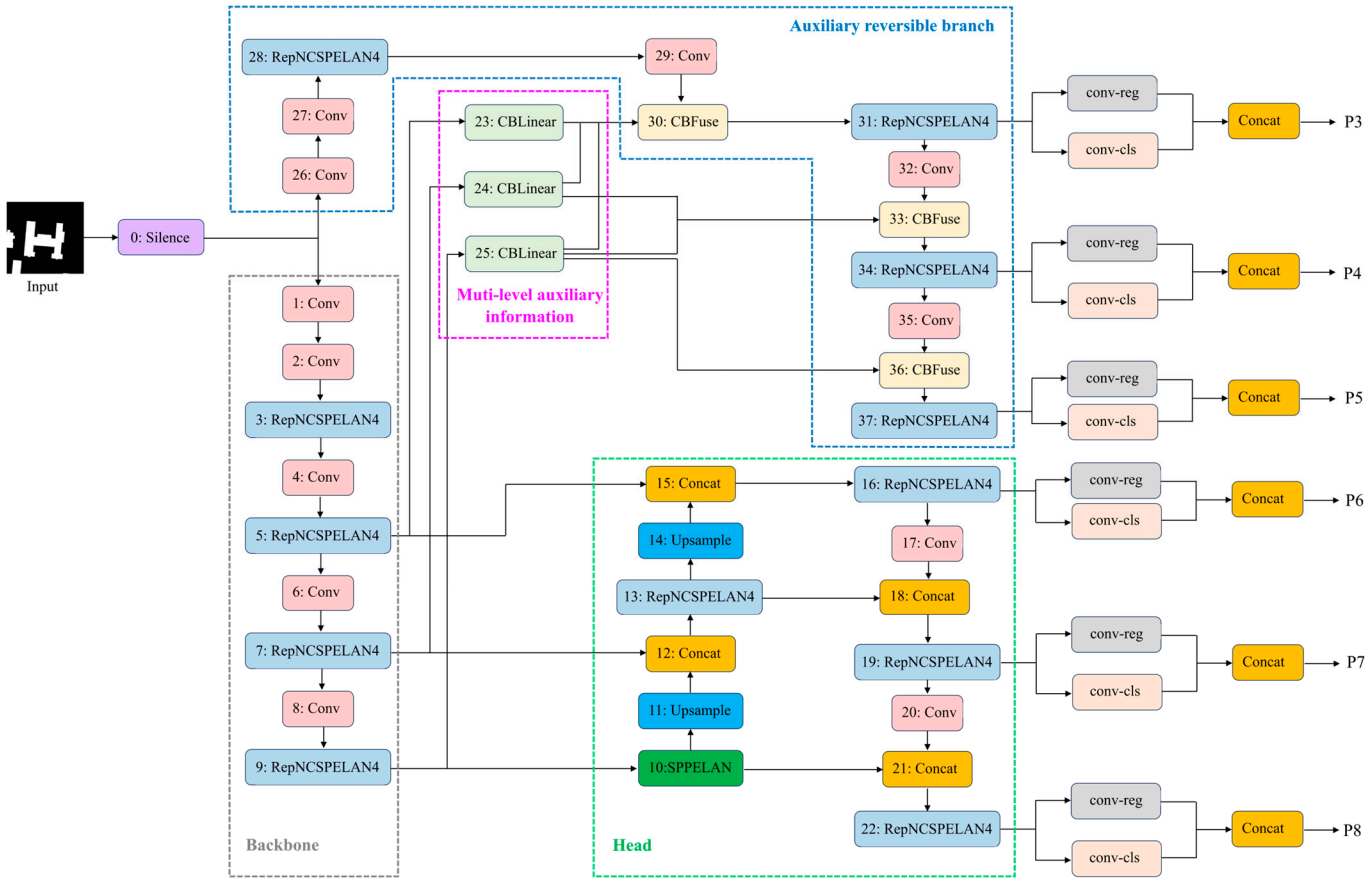


Figure 2. Network architecture of YOLOv9.

The YOLOv9 model introduces a novel concept called Programmable Gradient Information (PGI), which aims to solve the issue of information loss in deep networks during the forward propagation process in object detection models. PGI ensures that deep neural networks can maintain complete input information during the learning process, thereby obtaining reliable gradient information to update weights and improving the accuracy of weight updates [46]. As shown in Figure 2, PGI consists of three main components: the main branch (backbone), the auxiliary reversible branch, and multi-level auxiliary information. PGI generates stable gradients through the auxiliary reversible branch to ensure that deep network features maintain key attributes when performing specific tasks, avoiding semantic information loss that may occur when fusing multi-path features in traditional deep supervision. Additionally, YOLOv9 adopts a new lightweight network architecture called Generalized Efficient Layer Aggregation Network (GELAN, as shown in Figure 3). The GELAN module is a further extension of the ELAN module from YOLOv7, with the design considering the parameter amount, computational complexity, and inference speed. GELAN optimizes the network structure through gradient path planning and achieves parameter utilization efficiency surpassing current state-of-the-art methods. This design not only improves the model's performance but also ensures its efficiency, allowing YOLOv9 to maintain a lightweight profile while achieving accuracy and speed.

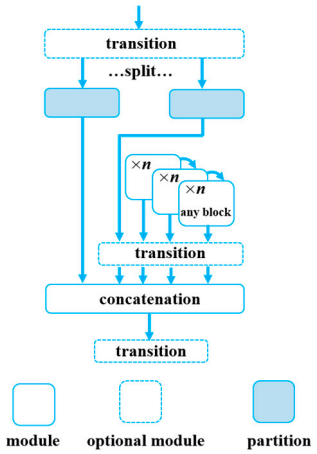


Figure 3. The GELAN module of YOLOv9 [46].

3.2. Attention Mechanism

The attention mechanism is a technique used in deep learning models that allows the model to selectively focus on specific areas of the input data when making predictions [47]. This mechanism, inspired by the human mental process of selective focus, has emerged as a pillar in a variety of applications, accelerating developments in natural language processing, computer vision, and beyond [48]. The introduction of an attention mechanism in object detection can make the model focus on the key information of the image, filter irrelevant information, and save computing resources. At the same time, the detection effect of the model on small targets can be improved. In this paper, SE, CBAM, NAM, and SimAM attention modules are, respectively, introduced into the red box position, as shown in Figure 4, and their performance improvements are compared.

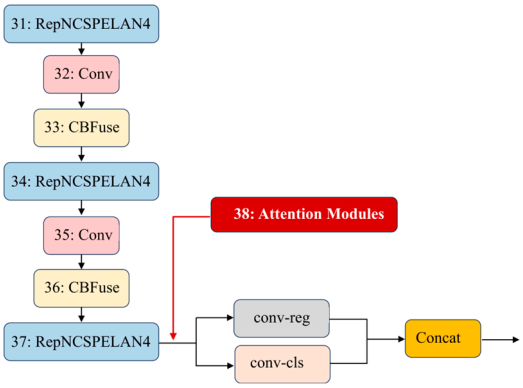


Figure 4. The introduction position of attention modules.

3.2.1. SE

The SE attention mechanism mainly includes Squeeze and Excitation operations. At the same time, attention is paid to the important difference of channel information in the feature map so that the model adaptively adjusts the attention weight of each channel. The basic structure of the SE module is shown in Figure 5 [49].

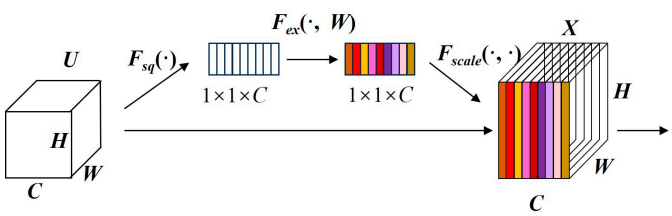


Figure 5. The introduction position of attention modules [49].

3.2.2. CBAM

CBAM (convolutional block attention module) mainly includes two sequential submodules: the channel attention module (CAM) and the spatial attention module (SAM) [50]. Based on the input feature map, CBAM successively integrates the attention weights along the channel and spatial modules and multiplies the input features, respectively, to obtain new features in order to extract important information from the feature map. Its basic structure is shown in Figure 6.

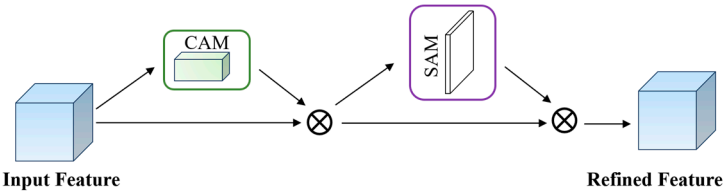


Figure 6. The introduction position of attention modules [50].

3.2.3. NAM

NAM (normalization-based attention module) uses the contribution factor of weight to improve the performance of the attention mechanism [51]. NAM uses batch-normalized scale factors to represent the importance of weights, which can effectively avoid the full connection layer and convolutional layer used by SE and CBAM modules. That is, NAM uses CBAM's module integration, redesigns the channel and spatial attention submodules, and then empowers the NAM module at the end of each network block. For the channel attention submodule, the scale factor in batch normalization is used, and the channel variance is calculated by the scale factor as the basis to measure the importance of weight. The structures of CAM and SAM are shown in Figure 7.

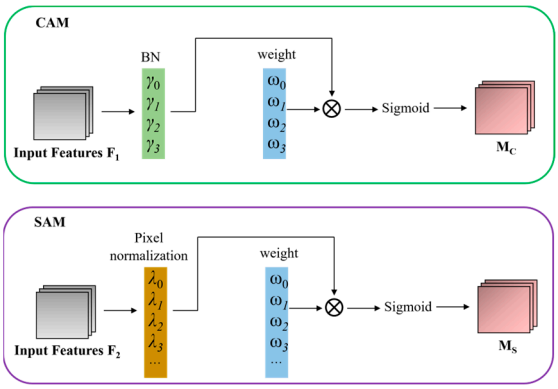


Figure 7. The structures of CAM and SAM in the NAM module [51].

3.2.4. SimAM

SimAM (simple attention module) is a simple and parameter-free attention module for convolutional neural networks [52]. The design of SimAM was inspired by neuroscience theories in the mammalian brain and, in particular, the design of an energy function based on the established theory of spatial inhibition to implement this theory. SimAM implements this function by deriving a simple solution that takes the function as the attention importance of each neuron in the feature graph. The implementation of the attention module is guided by this energy function, avoiding excessive heuristics. As shown in Figure 8, SimAM improves performance on a variety of visual tasks by inferring 3D attention weights for feature graphs and optimizing the energy function to find the importance of each neuron.

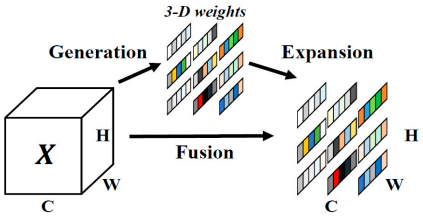


Figure 8. The structure of the SimAM attention module [52].

4. Building Shape Recognition Based on YOLO Models

The overall procedure of the proposed method is described as follows: First, the buildings with typical shape types are filtered out from the building data, and the training datasets are constructed accordingly. To compare the performance of different YOLO models, another two significant update versions of the YOLO series, namely YOLOv5 and YOLOv8, are also selected and trained to compare with YOLOv9 and attention mechanism models, ultimately resulting in the corresponding building shape recognition models. An initial building dataset is used to construct a test dataset, on which the trained building shape recognition models are predicted and validated. The detection and recognition effects of different models are compared. The overall process is shown in Figure 9.

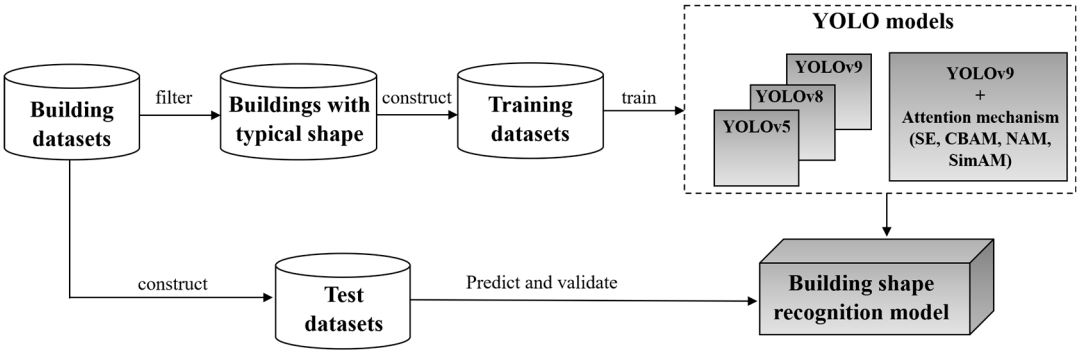


Figure 9. Recognition procedure of building shape based on YOLO object detection model.

4.1. Training Datasets Creation

The building data used for shape recognition were obtained from the GeoFabrik of OSM (OpenStreetMap), which mainly focuses on European countries (such as Germany, France, the United Kingdom, etc.); then, the typical types of building shapes were differentiated through manual selection. To ensure the balanced effects in training for different types of building shapes, 50 buildings of each shape type were selected as training data. When selecting the training sample data for buildings, four principles should be followed:

- **Principle 1**: The selected buildings should have the standard shape of a certain type. This ensures that the samples can provide the model with a clear basic style for different shape types;
- **Principle 2**: The selected buildings generally have a standard shape but with complex convex and concave parts in the building contours. This principle aims to improve the model's ability to distinguish different shape types in complex conditions;
- **Principle 3**: The selected buildings are approximate to a certain standard shape but with some partial or global shape deformations such as stretching or distortion. This is intended to further enhance the capabilities of the model to recognize building shapes under various conditions;
- **Principle 4**: The selected buildings have both the complex contours of Principle 2 and the shape deformations of Principle 3, which aim at further increasing the complexity of the training data.

The four principles range from simple to complex, fully considering the human visual cognitive characteristics and aligning with the progressive learning character of "from

easy to difficult" in human learning and judgment. During the training process, it can increase the diversity of the training data, providing data support for enhancing the model's generalization capability. Since the YOLO series models have already incorporated data augmentation methods, such as mosaic, adaptive anchor box calculation, and adaptive target scaling, which effectively expand the training dataset, there is no need to perform separate data augmentation on the training data samples. Table 1 shows some examples of training data, where the number of training samples for each type of building shape is generally consistent under each principle.

Table 1. Examples of different types of buildings for training.

Shape Type	Principle 1 Standard Shape	Principle 2 Complex Shape	Principle 3 Shape Deformation	Principle 4 with Prin.2 and Prin.3
E-like				
F-like				
H-like				
L-like				
T-like				
Y-like				
Z-like				
Cross-like				

For the annotation of the training dataset, the online annotation tool MakeSense was used to label the samples by manually judging the type of building and annotating the enclosing rectangle box for the building. Ultimately, a TXT label file is obtained that contains information such as the type of building, center point coordinates, and the width and height of the annotation box.

4.2. Training of YOLO Models

The training of the YOLO series models was conducted on a Windows 11.0 64-bit operating system. The computer hardware environment includes an Inter(R) Core(TM) i9-14900HX CPU @ 2.20 GHz, 32 GB RAM, and a GPU (NVIDIA GeForce RTX 4090 Laptop GPU). Python programming language is used, with Python 3.9 version, and Pytorch 2.01 is utilized as the deep learning framework. The CUDA 12.4 version is also employed.

The initial weights for training were provided by the pre-trained weights of YOLOv5, YOLOv8, and YOLOv9. These pre-trained weights were obtained through training on the COCO (Common Objects in Context) dataset and represent a set of parameters with optimal recognition effects. The specific size of the weight files is shown in Table 2. For example, YOLOv5 offers four different model sizes (s, m, l, x models) and their corresponding pre-trained weights (in .pt files). The larger the weight file, the larger the number of parameters, and the better the recognition effect. However, the corresponding training time will also be longer. The hyperparameters for training are set as follows: the learning rate is set to 0.01;

the batch size is set to −1, which means automatically finding the optimal batch size by the model; the image size is 640 × 640; and the epoch is 400.

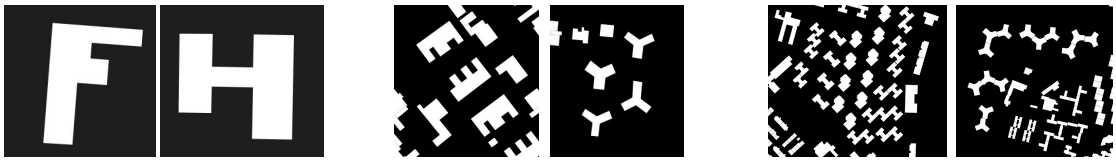
Table 2. Pre-trained weights of YOLO series models.

Model	Weights	Size
yolov5s	yolov5s.pt	14.1 MB
yolov5m	yolov5m.pt	41.1 MB
yolov5l	yolov5l.pt	90.1 MB
yolov5x	yolov5x.pt	166.0 MB
yolov8s	yolov8s.pt	21.5 MB
yolov8m	yolov8m.pt	49.7 MB
yolov8l	yolov8l.pt	83.7 MB
yolov8x	yolov8x.pt	130 MB
yolov9c	yolov9c.pt	98.3 MB
yolov9e	yolov9e.pt	133 MB

4.3. Prediction and Evaluation

4.3.1. Test Dataset Description and Model Prediction

After training, the model is applied to a new area's test dataset for prediction and validation. Compared to the training dataset, where each image contains only one type of building shape, the test dataset further increases the complexity and richness of the detection task. Three-level test datasets with different complexity were constructed to verify the training effectiveness and recognition capability of the model (Figure 10).



(**a**) (**b**) (**c**)

Figure 10. Examples of test datasets: (**a**) single building; (**b**) complex scene; (**c**) large-area scene.

Test Dataset 1: The Single-Building Test Dataset consists of 160 images, with 20 images for each of the eight typical shape types selected in this study. This dataset is targeted at different shape types. It is primarily used to test the detection effect of the trained models on the eight typical shape types of buildings selected in this paper. Among them, the test data for the seven letter-like types, excluding the cross-like type, are randomly selected from the public building shape datasets provided by reference [53].

Test Dataset 2: The Complex Scene Test Dataset consists of 50 scenes. Unlike Test Dataset 1, which contains only single buildings, this dataset includes more complex scenes with multiple types and numbers of buildings to detect, and the target buildings are surrounded by other buildings.

Test Dataset 3: The Large-area Scene Test Dataset consists of 10 scenes. This test dataset simulates the scenario of map browsing, with a larger area involved and multiple shape types of buildings within each scene. There are more target buildings of different shapes to be detected, and they are also surrounded by numerous other buildings, posing additional challenges. This dataset is used to evaluate the model's recognition ability and generalization in larger areas for different shape types of buildings.

By utilizing these three levels of complexity in the test datasets, the model's performance can be thoroughly assessed under various conditions, ensuring that the trained model is robust and effective in real-world scenarios.

During the model prediction and validation, it is necessary to set two parameters: confidence (conf) and Intersection over Union (IoU). These two parameters play a crucial role in determining the quality of the detected objects and the model performance. The

confidence (conf) parameter represents the probability that the model believes there is an object of a certain type within a detection box. It ranges from 0 to 1, with higher values indicating a higher probability that the detection box contains the target object. In this study, the confidence threshold is set to 0.5, which means that only detection boxes with a probability greater than 0.5 will be considered positive detections. IoU measures the overlap between the predicted bounding box and the ground truth bounding box. It is calculated as the area of the intersection divided by the area of the union between the predicted and the ground truth bounding boxes. A higher IoU value indicates a better match between the predicted and the truth-bounding boxes. The IoU threshold determines how similar a predicted bounding box must be to the ground truth bounding box to be considered a true positive. In this study, the IoU threshold is set to 0.25, which means that a predicted bounding box must have an IoU of at least 0.25 with the ground truth bounding box to be considered a true positive.

By setting these thresholds, the model can filter out low-confidence detections and only consider those that have a significant overlap with the ground truth, thus improving the quality of the detected objects. The recognition results for YOLOv5, YOLOv8, YOLOv9, and YOLOv9 with attention modules on the three test datasets are shown in Tables 3–5 (only shows results within the largest pre-trained weights of each model).

Table 3. Examples of recognition results on Test Dataset 1 (single building).

Shape Type	YOLOv5x	YOLOv8x	YOLOv9e	YOLOv9e + SE	YOLOv9e + CBAM	YOLOv9e + NAM	YOLOv9e + SimAM
E-like							
F-like							
H-like							
L-like							
T-like							
Y-like							
Z-like							
Cross-like							

Table 4. Examples of recognition results on Test Dataset 2 (complex scene).

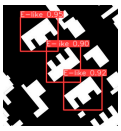
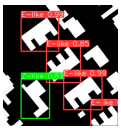
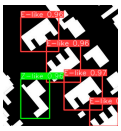
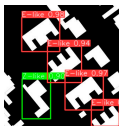
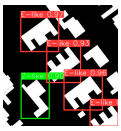
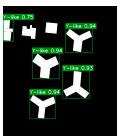
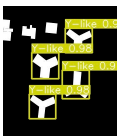
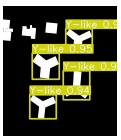
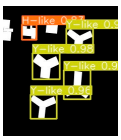
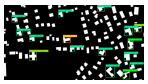
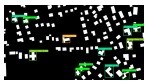
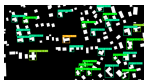
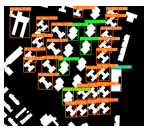
Shape Type	YOLOv5x	YOLOv8x	YOLOv9e	YOLOv9e + SE	YOLOv9e + CBAM	YOLOv9e + NAM	YOLOv9e + SimAM
E-like							
F-like							
H-like							
L-like							
T-like							
Y-like							
Z-like							
Cross-like							

Table 5. Examples of recognition results on Test Dataset 3 (large-area scene).

Scene	YOLOv5x	YOLOv8x	YOLOv9e	YOLOv9e + SE	YOLOv9e + CBAM	YOLOv9e + NAM	YOLOv9e + SimAM
1							
2							
3							

Table 5. *Cont.*

Scene	YOLOv5x	YOLOv8x	YOLOv9e	YOLOv9e + SE	YOLOv9e + CBAM	YOLOv9e + NAM	YOLOv9e + SimAM
4							
5							

4.3.2. Detecting Results Evaluation

The evaluation metrics of object detection models mainly include detection accuracy and detection speed. The speed metric primarily evaluates the model's processing performance on video data. Since this paper focuses solely on image detection, the speed evaluation metric is not calculated. When evaluating detection accuracy, the detection targets are typically categorized into positive samples and negative samples. A positive sample is correctly identified as a positive sample, and it is called a true positive (*TP*); a positive sample that is incorrectly identified as a negative sample is called a false negative (*FN*); a negative sample that is correctly identified as a negative sample is called a true negative (*TN*); and a negative sample that is incorrectly identified as a positive sample is called a false positive (*FP*). The precision, recall, and *F1* score are commonly used metrics to evaluate the performance of object detection models.

Precision (*P*): It measures the proportion of correctly detected objects out of all objects that were detected by the model. High precision indicates that the model is very good at avoiding false positives. Its calculation formula is as follows:

$$P = \frac{TP}{TP + FP} \tag{1}$$

Recall (*R*): It measures the proportion of correctly detected objects out of all actual objects in the dataset. High recall indicates that the model is very good at detecting all actual objects. Its calculation formula is as follows:

$$R = \frac{TP}{TP + FN} \tag{2}$$

F1 Score: It is the harmonic mean of precision and recall. It provides a single metric that balances the trade-off between precision and recall. A high *F1* score indicates that the model is performing well in terms of both precision and recall.

$$F1score = \frac{2 \cdot P \cdot R}{P + R} \tag{3}$$

These metrics are used to assess the model's performance in correctly identifying objects of interest and to understand where the model might be making mistakes. Table 6 provides the statistics of precision, recall, and F1 score for the YOLOv5, YOLOv8, YOLOv9, and YOLOv9 with different attention modules with different parameter sizes across the three test datasets.

Table 6. Statistic of different YOLO models on three test datasets.

Yolo Models	P (%)			R (%)			F1 Score		
	TestData1	TestData2	TestData3	TestData1	TestData2	TestData3	TestData1	TestData2	TestData3
YOLOv5s	95.1	74.8	35.4	88.2	89.9	19.7	0.915	0.817	0.253
YOLOv5m	90.1	77.5	49.0	94.1	91.1	38.6	0.921	0.838	0.432
YOLOv5l	87.0	76.2	31.2	90.1	82.2	33.9	0.885	0.791	0.325
YOLOv5x	87.3	76.4	45.0	87.3	83.2	33.1	0.873	0.797	0.381
YOLOv8s	96.6	87.2	46.2	92.9	79.8	7.3	0.947	0.833	0.126
YOLOv8m	93.9	87.2	37.9	91.4	79.8	6.8	0.926	0.833	0.115
YOLOv8l	91.9	82.4	40.0	79.2	81.5	8.1	0.851	0.819	0.135
YOLOv8x	96.5	83.3	28.6	89.6	68.4	8.6	0.929	0.751	0.132
YOLOv9c	90.2	77.8	86.6	95.2	92.6	87.0	0.926	0.846	0.868
YOLOv9e	93.3	84.3	90.7	92.1	93.9	92.8	0.927	0.888	0.917
YOLOv9e + SE	96.1	83.6	90.9	95.0	95.9	89.0	0.955	0.893	0.899
YOLOv9e + CBAM	97.3	90.4	95.2	93.1	93.5	90.3	0.952	0.919	0.927
YOLOv9e + NAM	95.1	88.8	91.8	88.8	90.0	90.2	0.918	0.894	0.910
YOLOv9e + SimAM	98.6	88.8	83.9	91.9	93.8	82.9	0.951	0.894	0.834

4.3.3. Results Analysis

(1) Analysis of basic YOLO models

By analyzing the results in Tables 3–5, the performance of different basic YOLO models can be compared across different test datasets and understand how the model's performance changes with different parameter sizes.

In Test Dataset 1, all images only contain one building type. The prediction results of YOLOv5, YOLOv8, and YOLOv9 models show minimal differences, with only slight variations in the detection probabilities of certain shapes. Except for a few models, the precision and recall rates are above 90%, with only a small number of type detection errors and missed detections. This indicates that the YOLO models possess the ability to recognize building shapes through training.

In Test Dataset 2, there is interference from other buildings surrounding the target buildings. The detection results show that all three models can identify the location and type of the target buildings, with a recognition accuracy of over 80% for most models, and the recognition errors are relatively concentrated in a few scenes. This suggests that the YOLO models are capable of identifying building shape types from complex backgrounds and can effectively avoid interference from surrounding noise.

In Test Dataset 3, as the detection area increases, significant performance differences among the YOLO models become apparent. The YOLOv9 model performs best, with precision and recall rates both above 85%. However, YOLOv5 and YOLOv8 models have more errors and missed detections in specific scenes. This is because in large-area scenes, due to scaling reasons, buildings become smaller, reducing their proportion in the image. When the target size is too small, the receptive field of the traditional convolutional neural networks used in the backbone networks of YOLOv5 and YOLOv8 becomes too large, leading to overly coarse features that cannot effectively distinguish between targets and backgrounds, resulting in more errors and missed detections. In contrast, YOLOv9 has several improved strategies to address small target detection issues, such as using the CSP Darknet53 network as the backbone, designing the PANet feature fusion module, and adopting the SIoU (Scale-Invariant Overlap Union) loss function. Therefore, YOLOv9 has the best performance to detect small buildings in large-area scenes.

(2) Analysis of YOLOv9 with attention modules

From Table 6, compared with the YOLOv9e model, it shows that by integrating attention modules, the performance of YOLOv9 has increased in general. The CBAM attention module especially enhances the precision rate in the three test datasets for 4%, 6.1%, and 4.5%, respectively. This indicates that by integrating the attention modules, the YOLO models can perform better than the original model. Among the four related attention modules, the CBAM module has the best improvements to the original models.

4.3.4. Test on Different Zoom Levels

To address the performance discrepancies among different YOLO models on Test Dataset 3, a test was conducted to verify the detection capabilities of the models on small targets at different zoom levels within the same scene. By analyzing the map browsing experience, it was determined that zoom levels of 21 and 20 are relatively reasonable and comfortable for human vision. Zoom levels higher than 21 result in buildings that are too large and may exceed the map canvas, making it difficult to observe map elements. Zoom levels lower than 20 lead to buildings that are too small to be distinguished by the human eye. To test the models' ability to detect small target buildings, two additional zoom levels (19 and 18) were added below the optimal browsing levels (21 and 20). A total of 200 test data images (50 for each zoom level) were created to validate the detection performance of YOLO models on targets of different sizes. Given that the size of the buildings themselves can affect the visual impact at different zoom levels, the test focused on smaller buildings. The test results, with confidence (conf) thresholds set at 0.5 and 0.01, which aim to assess the models' maximum detection capabilities, are presented in Table 7.

Table 7. Prediction results of YOLOv9e in different zoom levels.

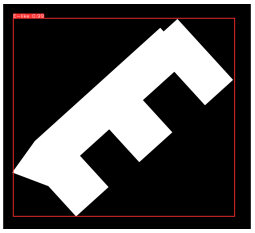
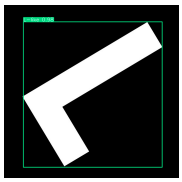
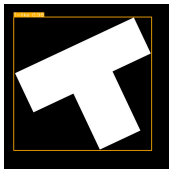
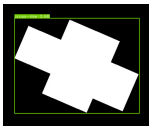
Shape Type	Level 21	Level 20	Level 19	Level 18
E-like	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5/0.01
F-like	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5  Conf = 0.01	 Conf = 0.5/0.01
F-like	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5  Conf = 0.01	 Conf = 0.5/0.01
L-like	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5  Conf = 0.01
T-like	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5  Conf = 0.01

Table 7. *Cont.*

Shape Type	Level 21	Level 20	Level 19	Level 18
Y-like	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5 Conf = 0.01	 Conf = 0.5/0.01
Z-like	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5/0.01
Cross-like	 Conf = 0.5/0.01	 Conf = 0.5/0.01	 Conf = 0.5/0.01	  Conf = 0.5 Conf = 0.01

The evaluation metrics in Table 8 show that when the zoom level is 18 (extremely small buildings), the detection performance of all YOLO models is not ideal. At zoom level 19 (normal small targets), the YOLOv9 model achieves relatively good detection accuracy, while the YOLOv5 and YOLOv8 models perform poorly. This test aligns with the results of the large-scale scene detection in Test Dataset 3 for the three basic YOLO models, further confirming the differences in their capabilities for detecting small targets. By adding the attention modules, the recall rate in the low zoom levels (18 and 19) has been largely increased in both of the two confidence thresholds, which demonstrates that the attention modules can enhance the ability of the YOLO model to recognize the small targets.

Table 8. Statistic of different YOLO models on different zoom levels.

Yolo Models	P (%) Conf = 0.5/0.01				R (%) Conf = 0.5/0.01			
	L18	L19	L20	L21	L18	L19	L20	L21
YOLOv5s	0.0/0.0	11.8/15.6	65.1/66.0	78.7/80.0	0.0/0.0	5.7/21.7	80.0/91.2	92.5/100.0
YOLOv5m	0.0/0.0	18.2/25.7	69.0/68.8	82.0/82.0	0.0/0.0	12.5/37.5	78.4/94.3	100.0/100.0
YOLOv5l	0.0/0.0	40.0/34.3	78.0/75.6	85.7/86.0	0.0/0.0	28.6/44.4	78.0/87.2	97.7/100.0
YOLOv5x	0.0/0.0	29.4/24.1	76.3/66.7	87.0/85.7	0.0/0.0	13.2/25.0	70.7/85.7	90.9/97.7
YOLOv8s	0.0/0.0	50.0/18.4	68.6/63.8	88.1/81.6	0.0/0.0	6.4/36.8	61.5/90.9	82.2/97.6
YOLOv8m	0.0/0.0	12.5/17.1	82.1/68.8	91.5/92	0.0/0.0	2.3/28.6	74.4/94.3	93.5/100.0
YOLOv8l	0.0/14.3	26.7/18.9	78.9/72.3	95.5/93.8	0.0/2.3	10.3/35.0	71.4/91.9	87.5/95.7
YOLOv8x	0.0/0.0	33.3/23.5	80.0/70.8	89.1/88.0	0.0/0.0	9.5/33.3	65.1/94.4	91.1/100.0
YOLOv9c	100.0/100.0	90.3/90.3	93.2/93.2	98.0/96.0	2.0/2.0	59.6/59.6	87.2/87.2	100.0/100.0
YOLOv9e	100.0/66.7	96.4/81.4	93.2/96.0	97.9/98.0	8.0/43.9	55.1/83.3	87.2/100.0	93.9/100.0
YOLOv9e + SE	100.0/74.4	97.2/93.75	97.9/94.0	89.8/88.0	24.0/78.0	72.0/96.0	94.0/100.0	98.0/100.0
YOLOv9e + CBAM	100.0/81.1	100.0/97.8	100.0/98.0	100.0/100	18.0/74.0	72.0/92.0	94.0/98.0	94.0/98.0
YOLOv9e + NAM	94.4/76.9	94.3/85.7	91.8/90.0	95.7/89.8	36.0/78.0	70.0/98.0	98.0/100.0	94.0/98.0
YOLOv9e + SimAM	80.0/60.5	87.0/83.7	95.9/92.0	95.8/88.0	30.0/86.0	92.0/98.0	98.0/100.0	96.0/100.0

5. Discussion

5.1. Analysis of Different Building Shape Types

The detection results of different YOLO models on Test Dataset 1 are statistically analyzed for precision and recall rate by shape type (as shown in Table 9). Among the selected eight typical building shape types, the L-like, H-like, T-like, Y-like, and Z-like buildings have better recognition accuracy. This is because the features of these building shape types are relatively prominent, making it easier for the model to extract their features. The errors mainly occur in two aspects: one is the misidentification between E-like and F-like buildings. These two shape types are similar and easy to confuse, especially since the small convex and concave building contours can interfere with the model's judgment

(Figure 11a). Another one is the misidentifying of the cross-like buildings as Z-like. This is particularly evident in "chubby" cross-like buildings, which are relatively similar to Z-like buildings and can easily lead to misidentification (Figure 11b). In terms of recall, the L-like buildings have a relatively low recall rate. This is because, compared to other types, the shape of L-like buildings is simpler, and the model can extract relatively less feature information, making it more prone to missed detections.

Table 9. Statistics of different YOLO models on different building shapes.

YOLO Models	E-Like (%)		F-Like (%)		H-Like (%)		L-Like (%)		T-Like (%)		Y-Like (%)		Z-Like (%)		Cross-Like (%)	
	P	R	P	R	P	R	P	R	P	R	P	R	P	R	P	R
YOLOv5s	78.9	93.8	100	90	100	95	100	60	100	75	100	100	89.5	94.4	95	100
YOLOv5m	94.7	94.7	52.9	75	100	100	100	65	87.5	77.8	100	100	94.1	84.2	90	100
YOLOv5l	95	100	60	64.3	100	75	100	95	94.1	84.2	100	100	90	100	100	100
YOLOv5x	95	100	47.1	72.7	100	95	81.8	50	100	80	100	100	95	100	65	100
YOLOv8s	100	100	84.2	94.1	100	85	100	85	100	80	100	100	100	100	85	100
YOLOv8m	100	95	66.7	66.7	100	95	100	80	100	90	100	100	100	100	80	100
YOLOv8l	94.7	94.7	50	81.8	100	85	100	20	100	80	100	100	100	80	94.7	94.7
YOLOv8x	95	100	84.2	94.1	100	85	100	55	100	95	100	100	100	95	89.5	94.4
YOLOv9c	85	100	52.9	75	100	100	100	80	90	100	100	100	100	100	90	100
YOLOv9e	95	100	85	100	100	90	100	65	88.9	80	100	100	100	95	90	100
YOLOv9e + SE	84.2	95	95	100	100	100	100	85	100	95	100	100	100	95	94.4	90
YOLOv9e + CBAM	100	100	73.3	75	100	100	100	80	100	95	100	100	100	95	100	100
YOLOv9e + NAM	100	100	62.5	80	100	100	100	60	94.4	90	95	100	100	80	100	100
YOLOv9e + SimAM	94.4	90	100	80	100	95	100	85	85	100	100	100	100	85	100	100
Average	93.33	97.82	68.3	81.37	100	90.5	98.18	65.5	96.05	84.2	100	100	96.86	94.86	87.92	98.91

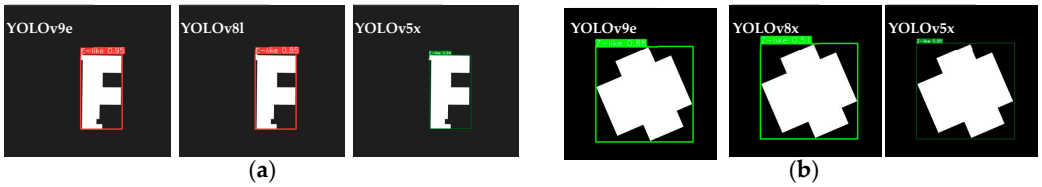


Figure 11. Misidentification examples of building shape: (**a**) F-like; (**b**) cross-like.

5.2. Analysis of Different YOLO Models

From the results of the three test datasets in Test Dataset 1, for the judgment of single building shape types, the performance of the three YOLO models is relatively similar, and the precision and recall rates are relatively high. This indicates that through training, the YOLO models can have the ability to recognize building shapes, and the models can judge building shapes without being affected by the contour morphology and deformation of buildings, identifying the overall shape from the perspective of standard shape, complex contour, local deformation, and complex contour combined with local deformation (Figure 12). This demonstrates that through multiple epochs of learning and training on the sample data, the YOLO model can simulate human visual cognitive characteristics, avoiding the impact of local complex morphology and deformation on shape recognition and possessing a high level of generalization and cognitive ability similar to human vision for building shapes.

However, with the increase in scene complexity and zoom level, the detection performance of YOLOv8 and YOLOv5 has somewhat decreased, mainly due to the network structure's lack of consideration for small target detection. The YOLOv9 model, on the other hand, can still maintain good detection performance. Additionally, the threshold for confidence (conf) also affects the detection results, especially in terms of recall rate. The threshold used in this experiment (conf = 0.5) is relatively strict. When this threshold is reduced, more targets could be detected.

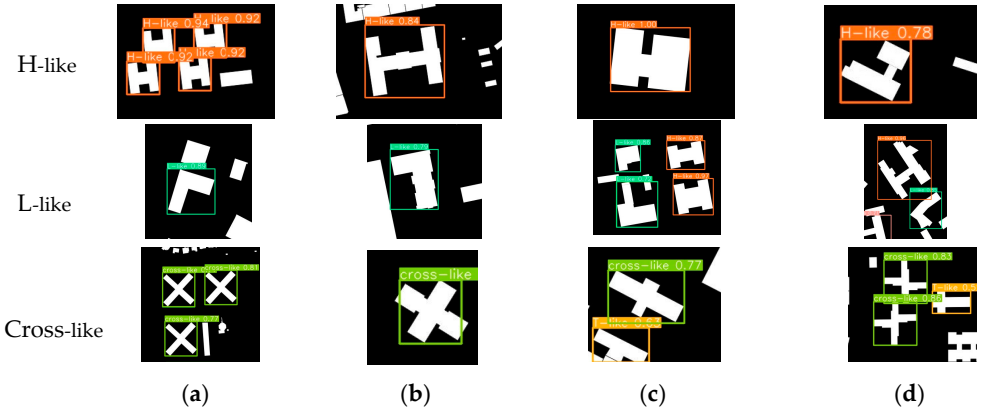


Figure 12. Detection results of YOLOv9e on Test Dataset 2: (**a**) standard shape; (**b**) complex contour; (**c**) local deformation; (**d**) complex contour combined with local deformation.

5.3. Analysis of Simulating Human Visual Cognition

Abstract summarizing and fuzzy judgment abilities are advanced cognitive abilities of human vision that determine the ability to accurately extract the main features of objects from interference and noise. These abilities are also challenging for computer algorithms to simulate. Based on the detection results in various complex scenarios, the overall evaluation of the YOLO model in this paper is as follows: it possesses the ability to simulate advanced human visual cognition with a high level of intelligence. This is mainly reflected in the following six abilities:

Abstract summarizing ability: The YOLO model can abstract and summarize the main shape of a building from complex details in the building contour, achieving accurate recognition (Figure 13a).

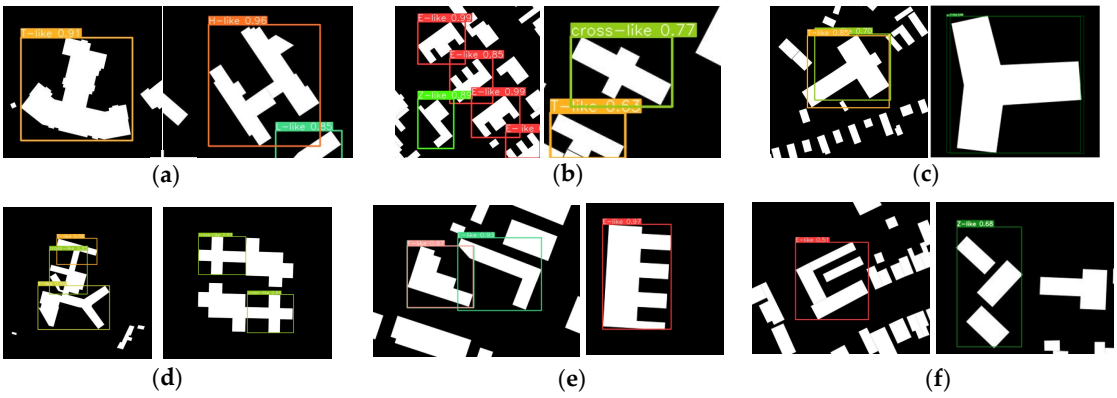


Figure 13. Simulation of advanced human visual cognition by YOLO models: (**a**) abstract summarizing; (**b**) edge detection; (**c**) fuzzy judgment; (**d**) local recognition; (**e**) analogical reasoning; (**f**) visual extension.

Edge detection ability: It can identify shapes at the edge positions of the image. For example, in Figure 13b, there are incomplete E-like and T-like buildings at the edge positions. The YOLO model can determine the type from incomplete local information.

Fuzzy judgment ability: For buildings with ambiguous shape types, the YOLO model can identify multiple detection results. As shown in Figure 13c, the building can be classified as T-like or cross-like, Y-like or T-like, and the YOLO model gives two potential prediction results, indicating its ability to simulate human fuzzy judgment.

Local recognition ability: As shown in Figure 13d, for some local shapes within larger irregular buildings, the YOLO model can identify specific shapes from the overall information, demonstrating the ability to distinguish local features from the overall context.

Analogical reasoning ability: As shown in Figure 13e, from a human cognitive perspective, two buildings in the image are considered similar to F-shaped and E-shaped buildings. Although the YOLO model's judgment is a false detection, the process simulates the analogical reasoning effect of human cognition.

Visual extension ability: Human vision has the ability to extend judgment, allowing the perception of specific shapes composed of multiple discrete buildings, as shown in Figure 13f. Although the YOLO model's judgment is also a false detection, this process demonstrates its ability to simulate visual extension characteristics, which can support other detection tasks.

These six abilities represent advanced judgment abilities in human visual recognition, which are unique to human vision for the comprehensive identification and simulation of imagination of complex objects. The YOLO model can simulate these abilities, indicating a high level of intelligence.

5.4. Comparison with Other Methods

Compared to methods using convolutional neural networks for image classification (such as the AlexNet convolutional neural network in the literature [27]), the shape classification method based on the YOLO object detection model does not require the removal of noise interference around the target building. It not only identifies the type of building shape but also provides localization, which can support subsequent simplification operations.

In contrast to the method using graph neural networks to recognize building shapes in the literature [29], the target detection method does not require extensive feature extraction and constructs the graph structure during the pre-training of the model. It can simulate human cognitive characteristics and detect blurry shapes without the need for separate training data for special-shaped buildings. To compare the proposed method with the GNN method, the GNN method has been implemented in the above three test datasets. The results of the three test datasets show that the proposed method has a comparable performance with the GNN method in common shape recognition.

To make a further comparison between these two methods, another new test dataset was created. The selected buildings have more complex contours, which increases the recognition difficulty. In the new dataset, the buildings have a more complex shape; there are various convex and concave in the contours, as well as deformations. Figure 14 displays the recognition results; the wrong recognition of the GNN method is marked by the red boxes. From the statistics of Table 10, the YOLOv9 model with the attention modules performs better than the GNN method.



(**a**) (**b**)

Figure 14. Comparison between (**a**) GNN method and (**b**) YOLOv9e + CBAM with complex shapes.

Table 10. Comparison between GNN and YOLOv9 with different attention modules.

Models	P (%)	R (%)
GNN	62.7	100
YOLOv9e + SE	76.1	100
YOLOv9e + CBAM	78.2	100
YOLOv9e + NAM	77.5	97.9
YOLOv9e + SimAM	69.7	100

5.5. Further Improvements

This study has several limitations. The model still has errors in detection and missed detections. Missed detections are mainly influenced by the confidence (conf) threshold setting. In our study, the conf threshold is set to 0.5, and when the threshold is reduced, more building shapes can be detected. For the false detections, two approaches can be taken: by further increasing the number and diversity of samples in the training dataset and by further adjusting the model's network structure to enhance the model's detection capabilities. The proposed method is raster-based, and the transfer of the recognized shape type information to the original vector data needs to be considered. This is also the future study of this paper.

6. Conclusions

This paper introduces the YOLO object detection models to identify typical building shape types. By constructing four levels of training datasets for the model, the model is trained to fully learn the features of building shapes. The trained YOLO object detection model can not only recognize the type of building shapes but also identify their specific locations. It also possesses the ability to simulate advanced human visual cognitive characteristics and is capable of performing building shape detection tasks in complex scenarios. The method presented in this paper only uses the attention mechanism to improve the basic YOLO models. How to further improve and adjust the YOLO model specifically for building shape recognition to further enhance the detection quality effectively is worth in-depth research.

Author Contributions: Conceptualization, Xiao Wang; building data collection, Xiao Wang; training data collection and annotation, Xiao Wang, Xu Wang and Bohao Li; methodology, Xiao Wang and Haizhong Qian; YOLO model training and validating, Xiao Wang; experimental results analyzing, Xiao Wang and Limin Xie; supervision, Haizhong Qian; original manuscript writing, Xiao Wang; manuscript ed-iting, Limin Xie and Haizhong Qian. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by The National Natural Science Foundation of China, grant number 42101453, 42271463.

Data Availability Statement: The building data was downloaded from Openstreetmap. The selected data of different building shape type for training are available by contacting the first author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Zhu, G. *Cartography*; Wuhan University Press: Wuhan, China, 2004.
2. Robinson, A.H.; Morrison, J.L.; Muehrcke, P.C.; Kimerling, A.J.; Guptill, S.C. *Elements of Cartography*, 6th ed.; Wiley: New York, NY, USA, 1995.
3. Shea, K.S.; McMaster, R.B. Cartographic generalization in a digital environment: When and how to generalize. *Scanning Electron Microsc. Meet* **1989**, 56–67.
4. Roth, R.E.; Brewer, C.A.; Stryker, M.S. A typology of operators for maintaining legible map designs at multiple scales. *Cartogr. Perspect.* **2011**, *68*, 29–64. [CrossRef]
5. Foerster, T.; Stoter, J.E.; Köbben, B. Towards a formal classification of generalization operators. In Proceedings of the 23rd International Cartographic Conference, Moscow, Russia, 4–10 August 2007.

6. Du, S.; Zhang, F.; Zhang, X. Semantic classification of urban buildings combining VHR image and GIS data: An improved random forest approach. *ISPRS J. Photogramm. Remote Sens.* **2015**, *105*, 107–119. [CrossRef]
7. Steiniger, S.; Lange, T.; Burghardt, D.; Weibel, R. An approach for the classification of urban buildings structures based on discriminant analysis techniques. *Trans. GIS* **2008**, *12*, 31–59. [CrossRef]
8. Xu, Y.; He, Z.; Xie, X.; Xie, Z.; Luo, J.; Xie, H. Building function classification in Nanjing, China, using deep learning. *Trans. GIS* **2022**, *26*, 2145–2165. [CrossRef]
9. Bandam, A.; Busari, E.; Syranidou, C.; Linssen, J.; Stolten, D. Classification of Building Types in Germany: A Data-Driven ModelingApproach. *Data* **2022**, *7*, 45. [CrossRef]
10. Wurm, M.; Schmitt, A.; Taubenböck, H. Building Types' Classification Using Shape-Based Features and Linear Discriminant Functions. *IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens.* **2016**, *9*, 1901–1912. [CrossRef]
11. Fan, H.; Zhao, Z.; Li, W. Towards Measuring Shape Similarity of Polygons Based on Multiscale Features and Grid Context Descriptors. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 279. [CrossRef]
12. Fu, J.; Fan, L.; Yu, Z.; Zhou, K. A Moment-Based Shape Similarity Measurement for Areal Entities in Geographical Vector Data. *ISPRS Int. J. Geo-Inf.* **2018**, *7*, 208. [CrossRef]
13. Basaraner, M.; Cetinkaya, S. Performance of shape indices and classification schemes for characterising perceptual shape complexity of building footprints in GIS. *Int. J. Geogr. Inf. Sci.* **2017**, *31*, 1952–1977. [CrossRef]
14. Li, W.; Goodchild, M.F.; Church, R. An efficient measure of compactness for two-dimensional shapes and its application in regionalization problems. *Int. J. Geogr. Inf. Sci.* **2013**, *27*, 1227–1250. [CrossRef]
15. Yan, X.; Ai, T.; Yang, M.; Yin, H. A graph convolutional neural network for classification of building patterns using spatial vector data. *ISPRS J. Photogramm. Remote Sens.* **2019**, *150*, 259–273. [CrossRef]
16. Zhang, Q.; Li, D.; Gong, J. Areal feature matching among urban geographic databases. *J. Remote Sens.* **2004**, *8*, 107–112.
17. Ai, T.; Shuai, Y.; Li, J. A spatial query based on shape similarity cognition. *Acta Geod. Cartogr. Sin.* **2009**, *38*, 356–362.
18. Ai, T.; Cheng, X.; Liu, P.; Yang, M. A shape analysis and template matching of building features by the Fourier transform method. *Comput. Environ. Urban Syst.* **2013**, *41*, 219–233. [CrossRef]
19. Wei, Z.; Guo, Q.; Cheng, L.; Liu, Y.; Tong, Y. Shape similarity measurement based on DNA alignment for buildings with multiple orthogonal features. *Acta Geod. Cartogr. Sin.* **2021**, *50*, 1683–1693.
20. Rainsford, D.; Mackaness, W. Template Matching in Support of Generalisation of Rural Buildings. In *Advances in Spatial Data Handling*; Richardson, D.E., van Oosterom, P., Eds.; Springer: Berlin/Heidelberg, Germany, 2002.
21. Yan, X.; Ai, T.; Zhang, X. Template Matching and Simplification Method for Building Features Based on Shape Cognition. *ISPRS Int. J. Geo-Inf.* **2017**, *6*, 250. [CrossRef]
22. Yan, X.; Ai, T.; Yang, M. A simplification of residential feature by the shape cognition and template matching method. *Acta Geod. Cartogr. Sin.* **2016**, *45*, 874–882.
23. Liu, P.; Ai, T.; Hu, J.; Cheng, X. Building-polygon simplification based on shape matching of prototype template. *Geomat. Inf. Sci. Wuhan Univ.* **2010**, *35*, 1369–1372.
24. Liu, P. Application of shape recognition in map generalization. *Acta Geod. Cartogr. Sin.* **2012**, *41*, 316.
25. Ma, L.; Yan, H.; Wang, Z.; Liu, B.; Lv, W. Geometry shape measurement of building surface elements based on self-supervised machine learning. *Sci. Surv. Mapp.* **2017**, *42*, 171–177.
26. Liu, P.; Huang, X.; Ma, H.; Yang, M. Fourier descriptor-based neural network method for high-precision shape recognition of building polygon. *Acta Geod. Cartogr. Sin.* **2022**, *51*, 1969–1976.
27. Jiao, Y.; Liu, P.; Liu, A.; Liu, S. Map building shape classification method based on AlexNet. *J. Geo-Inf. Sci.* **2022**, *24*, 2333–2341.
28. Yan, X.; Ai, T.; Yang, M.; Zheng, J. Shape cognition in map space using deep auto-encoder learning. *Acta Geod. Cartogr. Sin.* **2021**, *50*, 757–765.
29. Yu, Y.; He, K.; Wu, F.; Xu, J. Graph convolution neural network method for shape classification of areal settlements. *Acta Geod. Cartogr. Sin.* **2022**, *51*, 2390–2402.
30. Yan, X.; Yuan, T.; Yang, M.; Kong, B.; Liu, P. An adaptive building simplification approach based on shape analysis and representation. *Acta Geod. Cartogr. Sin.* **2022**, *51*, 269–278.
31. Liu, C.; Hu, Y.; Li, Z.; Xu, J.; Han, Z.; Guo, J. TriangleConv: A Deep Point Convolutional Network for Recognizing Building Shapes in the Map Space. *ISPRS Int. J. Geo-Inf.* **2021**, *10*, 687. [CrossRef]
32. Hu, Y.; Liu, C.; Li, Z.; Xu, J.; Han, Z.; Guo, J. Few-Shot Building Footprint Shape Classification with Relation Network. *ISPRS Int. J. Geo-Inf.* **2022**, *11*, 311. [CrossRef]
33. Guo, Q.; Liu, N.; Wang, Z.; Sun, Y. Review of deep learning based object detection algorithms. *J. Detect. Control* **2023**, *45*, 10–20+26.
34. Gupta, A.; Anpalagan, A.; Guan, L.; Khwaja, A.S. Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. *Array* **2021**, *10*, 100057. [CrossRef]
35. Jha, S.; Seo, C.; Yang, E.; Joshi, G.P. Real time object detection and trackingsystem for video surveillance system. *Multimed Tools Appl.* **2021**, *80*, 3981–3996. [CrossRef]
36. Li, K.; Wan, G.; Cheng, G.; Meng, L.; Han, J. Object detection in optical remote sensing images: A survey and a new benchmark. *ISPRS J. Photogramm. Remote Sens.* **2020**, *159*, 296–307. [CrossRef]
37. Baumgartner, M.; Jaeger, P.F.; Isensee, F.; Maier-Hein, K.H. nnDetection: A Self-configuring Method for Medical Object Detection. In Proceedings of the MICCAI 2021, Strasbourg, France, 27 September–1 October 2021; pp. 530–539.

38. Ma, Y.; Yin, J.; Huang, F.; Li, Q. Surface defect inspection of industrial products with object detection deep networks: A systematic review. *Artif. Intell. Rev.* **2024**, *57*, 333. [CrossRef]
39. Jiang, P.; Ergu, D.; Liu, F.; Cai, Y.; Ma, B. A Review of Yolo Algorithm Developments. *Procedia Comput. Sci.* **2022**, *199*, 1066–1073. [CrossRef]
40. Terven, J.; Córdova-Esparza, D.-M.; Romero-González, J.-A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS. *Mach. Learn. Knowl. Extr.* **2023**, *5*, 1680–1716. [CrossRef]
41. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
42. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
43. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
44. Bochkovskiy, A.; Wang, C.Y.; Liao, H.M. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv* **2020**, arXiv:2004.10934.
45. Wang, C.Y.; Bochkovskiy, A.; Liao, H.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Vancouver, BC, Canada, 18–22 June 2023; pp. 7464–7475.
46. Wang, C.Y.; Yeh, I.H.; Liao, H.M. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. *arXiv* **2024**, arXiv:2402.13616.
47. Mnih, V.; Heess, N.; Graves, A.; Kavukcuoglu, K. Recurrent models of visual attention. In *27th Annual Conference on Neural Information Processing Systems (NIPS 2014)*; Curran Associates, Inc.: Red Hook, NY, USA, 2014; pp. 2204–2212.
48. Brauwers, G.; Frasincar, F. A General Survey on Attention Mechanisms in Deep Learning. *IEEE Trans. Knowl. Data Eng.* **2023**, *35*, 3279–3298. [CrossRef]
49. Hu, J.; Shen, L.; Albanie, S.; Sun, G.; Wu, E. Squeeze-and-excitation networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; IEEE Press: New York, NY, USA, 2018; pp. 7132–7141.
50. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. CBAM: Convolutional block attention module. In Proceedings of the Computer Vision—ECCV 2018, Munich, Germany, 8–14 September 2018; Springer International Publishing: Cham, Switzerland, 2018; pp. 3–19.
51. Yang, L.; Zhang, R.; Li, L.; Xie, X. SimAM: A Simple, Parameter-Free Attention Module for Convolutional Neural Networks. In Proceedings of the 38th International Conference on Machine Learning, Virtual, 18–24 July 2021; Volume 139, pp. 11863–11874.
52. Liu, Y.; Shao, Z.; Teng, Y.; Hoffmann, N. NAM: Normalization-Based Attention Module. *NeurIPS Workshop ImageNet_PPF*. 2021. Available online: https://arxiv.org/abs/2111.12419 (accessed on 24 November 2021).
53. Yan, X.; Ai, T.; Yang, M.; Tong, X. Graph convolutional autoencoder model for the shape coding and cognition of buildings in maps. *Int. J. Geogr. Inf. Sci.* **2021**, *35*, 490–512. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.