Recognition of buildings on satellite images

Qing Bu
CETC
Les Information
System Co., Ltd
39020765@qq.com

Aleksei Miroevskiy
Faculty of Applied Mathematics
and Computer Science
Belarusian State University
Minsk, Belarus

***Abstract*—The paper is devoted to the methods for determining objects in images. The authors focus on practical: development of methods for solving the problem of determining structures on images using neural network approaches.**

Keywords—recognition of buildings on satellite images, convolutional neural networks.

I. INTRODUCTION

At present, automatic building recognition in satellite images is a relevant task that requires a significant amount of time and effort. However, there is no universally accepted methodology for solving this problem. The most common methods in this field are linear classifiers, analysis methods using self-organizing maps, and algorithms based on artificial neural networks.

Additionally, to improve the accuracy of building recognition in satellite images, additional data can be used, such as geospatial data, information about building heights and shapes, and data about the surrounding environment. However, developing more efficient and accurate methods for automatic building recognition in satellite images remains a relevant task for researchers in the field of computer vision and image processing.

The extraction of buildings in satellite imagery is applied in terrain mapping, urban planning, detection of illegally constructed objects, and other areas. The complexity of the task lies in the large variability of structural forms of buildings.

Various computer vision algorithms are also used to extract buildings in satellite images, which allow for the automatic detection and classification of objects in the image. However, such algorithms can make errors, especially in conditions of low image quality or strong shadows. Therefore, expert assessment plays an important role in the process of building extraction in satellite images, as it allows for refining the results of automatic processing and identifying possible errors.

Automatic building recognition will reduce the time and material costs of updating a geographic information system database and eliminate human error in solving this task.

In this work, an approach using a convolutional neural network (CNN) is demonstrated for building extraction and segmentation in images. The implemented method involves binary classification of pixels, dividing the set of pixels into two classes: building and non-building.

For training the CNN, annotated data - satellite images with marked buildings - were used. Various image preprocessing methods, such as noise filtering and contrast enhancement, were applied to improve the recognition quality. As a result of experiments, high-quality building recognition was achieved on test data. However, additional research and method improvement are necessary to achieve optimal results.

The developed approach can be used in various areas where automatic building recognition in satellite images is required. This work also includes an analysis of the obtained results and a comparison with other methods of building recognition in satellite images.

II. FEATURES OF SATELLITE AND DRONE IMAGES

Compared to drone imagery, satellite images usually provide a wider view and allow for information about a larger area. They can be useful for monitoring changes on the Earth's surface over a long period of time. However, the quality of the images may be limited by resolution, lighting, and weather conditions [1].

On the other hand, drone imagery allows for more detailed information about specific objects and land areas. Drones can fly at lower altitudes, which allows for clearer images, and they can also use different cameras and sensors to collect various types of data. However, drones can only be used in limited areas and require special training and permits for flights.

The resolution of terrain images obtained from drones depends on the camera used and camera settings. In general, the resolution can range from 0.3 to 50 megapixels (MP). The higher the resolution, the more detailed the image will be [2].

However, increasing the resolution also leads to larger file sizes and more complex data processing. Additionally, high resolution may only be necessary in certain cases, such as precise geodetic measurements or detailed terrain analysis. In other cases, a more modest resolution may be sufficient to obtain the desired information.

Resolution characteristics include the number of pixels in the image (width x height), file size, file format (JPEG, RAW), compression level, and image quality. Some cameras may also have the ability to record video at different resolutions and frame rates.

Resolution characteristics may also include the ISO range, which determines the camera's sensitivity to light, as well as the size of the sensor, which determines the level of detail and sharpness of the image. Additionally, resolution can be fixed or adjustable, allowing users to choose optimal settings for different shooting conditions. Some cameras may also have an autofocus feature, which helps to capture sharper and more focused images.

Another important aspect of resolution is the number of pixels per inch (PPI), which determines the pixel density on a screen or printed surface. High PPI provides a clearer and more detailed image, especially when zoomed in. Additionally, some cameras may have an image stabilization feature, which prevents blurring of the image when shooting in low light conditions or when the camera is in motion. The file format is also an important aspect of resolution, as it can be compressed or uncompressed, affecting the quality and size of the file.

Terrain imagery captured by drones using synthetic aperture radar (SAR) cameras has several characteristics compared to regular optical cameras. Firstly, SAR images are created based on electromagnetic waves, rather than visible light.

III. DATASET DESCRIPTION

The training dataset consists of 40 images. The images are diverse, including both rural landscapes and urban architecture. Each image has marked areas that contain building elements.

Fig. 1 Example of image from dataset

```
0 0.5375 0.67421875 0.134375 0.09375
0 0.546875 0.5921875 0.13125 0.0859375
0 0.5390625 0.5203125 0.13125 0.0890625
0 0.53984375 0.453125 0.13125 0.090625
0 0.5359375 0.37265625 0.1046875 0.08125
0 0.659375 0.60625 0.1203125 0.0859375
0 0.65234375 0.4546875 0.1203125 0.0859375
0 0.76328125 0.38359375 0.14375 0.096875
0 0.80859375 0.24765625 0.125 0.0859375
0 0.64453125 0.31484375 0.1296875 0.0859375
0 0.7796875 0.61328125 0.1390625 0.1125
0 0.65078125 0.534375 0.125 0.0953125
0 0.778125 0.534375 0.146875 0.096875
0 0.8765625 0.559375 0.115625 0.0859375
0 0.8859375 0.634375 0.1 0.071875
0 0.91796875 0.4109375 0.1484375 0.0921875
0 0.2890625 0.62109375 0.090625 0.0890625
0 0.45390625 0.55859375 0.090625 0.09375
0 0.3890625 0.55078125 0.084375 0.09375
0 0.54453125 0.29453125 0.0953125 0.078125
0 0.7890625 0.31796875 0.1265625 0.0984375
0 0.27265625 0.53515625 0.084375 0.09375
```

Fig. 2 Example of image annotations

Figures 1 and 2 show examples of the image and its markup, from the test dataset.

To increase the accuracy of the model, a decision was made to increase the size of the original data set by augmentation. The original images were rotated by 45 and -45 degrees, as shown in Figure 3. The transformation data was also displayed on the markup coordinates. The above transformation made it possible to increase the training data set by 3 times.

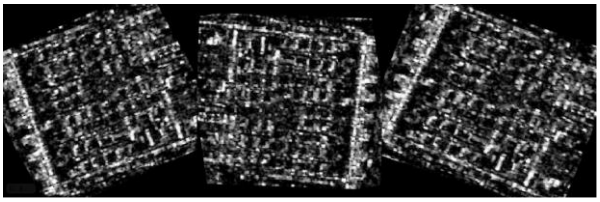

Fig. 3 Augmentation

IV. METRICS

In my work, I use the IoU (Intersection over Union) metric to check the quality of the algorithm. This metric allows you to determine the percentage of correctly classified pixels by calculating the ratio of the intersection of the set of pixels that are buildings and the set of pixels that have been classified as buildings by the neural network to the union of the set of pixels that are buildings and the set of pixels classified by the neural network as a building.

IoU is one of the most common metrics for evaluating the quality of computer vision and image processing algorithms. It allows you to get an accurate idea of how well the algorithm copes with the task and what improvements can be made to increase its efficiency. In addition, you can be used to compare different algorithms and choose the best one for a specific task.

In a formula form this relation looks like this:

$$IoU = \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

A is the set of pixels with constructions obtained using the algorithm,

B is the set of pixels that are constructions (from the original data set).

V. NEURAL NETWORK DESIGN

In my research I used the following tools:

Kaggle notebook is an online environment for creating, running, and sharing Jupyter notebooks on the Kaggle platform. Kaggle notebooks allow you to perform data analysis, create machine learning models, conduct experiments, and share results with other members of the Kaggle community. In Kaggle notebook, you also have access to Python libraries for scientific computing, machine learning, and data visualization, making it a powerful tool for working with data.

For implementing the algorithm, I used the Python programming language (version 3.11).

Ultralytics is a deep learning and computer vision framework - YOLO (You Only Look Once), which is used for object detection in images and videos.

The size of the dataset, the number of epochs, and the batch size are three key parameters in machine learning that can affect the model's quality.

The number of epochs refers to the number of times the model will go through all the data in the dataset. The

more epochs, the more time the model will spend on training and the more accurate results it can provide. However, if the number of epochs is too large, the model may start overfitting on the training data, which can lead to poor performance on the test data.

A batch is the amount of data that the model processes at once. Larger batches can speed up training but may also result in a loss of accuracy because the model may not get enough information about each data point. On the other hand, smaller batches can slow down the training process but may yield more accurate results as the model receives more detailed information about each data point.

The dataset size is constant, so it's important to find the optimal number of epochs and batch size. Usually, the number of epochs is chosen based on the validation results of the model. This means that the model is trained on the training data for several epochs and then evaluated on a validation set to assess its performance. If the model's performance improves with each epoch, training can continue until a certain model accuracy is achieved or until the performance stops improving.

The number of batches can also affect the model's performance. A larger batch size can speed up training but may lead to memory overflow and decreased model performance. A smaller batch size can improve the model's quality but may take more time for training. Dynamic batch sizes can also be used, where the batch size adjusts based on memory usage and the model's learning speed.

Overall, the optimal number of epochs and batch size for each machine learning model can only be determined through experimentation and analyzing the results.

Stage 1.

In the first stage of the experiment, the choice of the number of epochs and batch size was mostly random. The number of epochs was set to 300, and the batch size was set to 30:

!yolo task=detect mode=train model=yolov8n.pt data=/kaggle/input/diploma-buildings-detection/Diploma_dron_detection/data.yaml epochs=300 imgsz=640 batch=30 cache=True show=True

The metric reached 85%. Overall, the result is quite good, but the model clearly overfit, as can be seen in Figure 4, which reflects the model's accuracy.

Stage 2.

Based on the information logging during training from the previous stage, I concluded that the training accuracy stopped improving after 48 epochs. Therefore, this time I am setting the number of epochs to 48. I will keep the batch size the same for the sake of experiment purity.

!yolo task=detect mode=train model=yolov8n.pt data=/kaggle/input/diploma-buildings-detection/Diploma_dron_detection/data.yaml epochs=48 imgsz=640 batch=30 cache=True show=True

The metric is 89%. It's better, but there is still room for improvement.

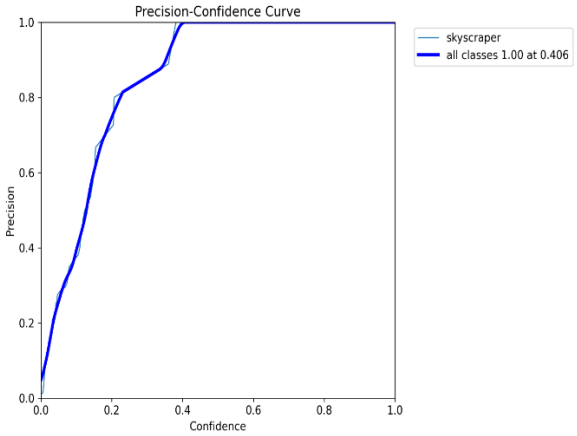

Fig. 4 Model training

Stage 3.

In this stage, I will change the batch size to 20.

!yolo task=detect mode=train model=yolov8n.pt data=/kaggle/input/diploma-buildings-detection/Diploma_dron_detection/data.yaml epochs=48 imgsz=640 batch=20 cache=True show=True

The metric reached 92%. I consider the experiment finished at this point. Figure 5 shows an example of an image processed by the obtained model.

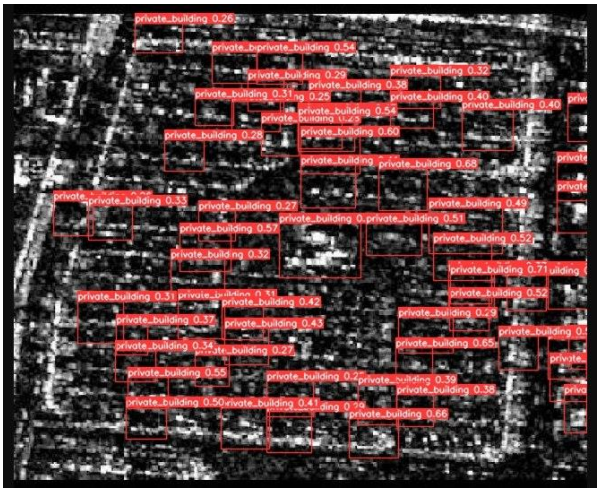

Fig. 5 Model result

VI. CONCLUSION

As a result, an algorithm was developed that allows for building segmentation on satellite images based on the YOLO model. The average accuracy of the obtained model was 92%, which is a very good result. However, to improve accuracy, more images with a higher number of channels can be used.

The obtained model can be used as a ready-made or partial solution for tasks related to creating maps, urban planning, searching for illegally constructed objects, and other areas.

Another important advantage of this algorithm is its ability to work with large volumes of data and automatically process information, which significantly

speeds up the analysis process and allows for more efficient results. Additionally, the use of the YOLO model enables high training and processing speeds, which is also crucial when working with large data volumes.

REFERENCES

[1] A. N. Averkin and S. A. Yarushev, "Prospects for the Application of Satellite Image Recognition Model for Macroeconomic Situation Analysis," Plekhanov Russian University of Economics, 2023, 102 p.

[2] S. Valero, P. Salembier, and J. Chanussot, "Hyperspectral image representation and processing with binary partition trees," IEEE Transactions on Image Processing, vol. 22, no. 6, pp. 2141–2157, 2013.