Method for constructing a network of urban roads using satellite images

E M Vikhtenko* and A A Tusikova
Pacific National University, 136 Tikhookeanskaya St., Khabarovsk, 680035, Russia

*E-mail: vikht.el@gmail.com

Abstract. An automatic method for recognizing the urban network of highways on high-resolution satellite images has been developed and implemented, based on the use of the Mask-RCNN convolutional neural network. The results of the neural network operation are post-processed by the Hough transformation and algorithms of mathematical morphology. The proposed method has acceptable accuracy and high speed of detection of asphalt urban roads.

1. Introduction
At present, automated information systems are becoming more and more widespread in the world, the functions of which include providing the user with various information obtained using space satellites. The results of the analysis of satellite images are used in the study of natural resources, monitoring of natural disasters, construction and design and survey work and in many other areas of human activity. In conditions of active construction of highways, their development and an increase in the volume of traffic flows, it is important to periodically update the maps of road networks. For these purposes, it seems natural to use satellite images. There are various methods for recognizing and classifying roads on satellite images, including a method using GPS data, LEGION, Bat model, a method based on the use of geometric and colour signs of roads, and others. They give quite good results for recognizing roads in rural areas, but for urban images, the accuracy of road identification drops noticeably. This is due to the fact that images of flat roofs, parking lots and some other objects have colour and texture characteristics close to the characteristics of asphalt roads.

The aim of this work is to develop and implement an automated method for recognizing an urban network of highways based on the Mask-RCNN convolutional neural network and having acceptable accuracy and high speed of detecting asphalt road surface. In the course of the work, satellite images of the city of Khabarovsk, Russia were used.

To successfully solve this problem, satellite images of very high resolution with four channels (RGB and near infrared) are required, so it was decided to use multispectral images with a spatial resolution of 2 meters from the WorldView-2 satellite. The radiometric resolution of these images is 16 bits per channel.

2. Using convolutional neural networks for road recognition
The method of using neural networks [1-3] is that a snapshot of the city is fed to the input of the network, and a binary image is taken from the output, each pixel of which belongs to the class “road” or “not road” (figure 1).
Figure 1. An example of road detection using a neural network.

This approach gives rather accurate results of road surface detection within the city limits; however, it requires a large sample size to train the neural network. Moreover, designing a network architecture is challenging. Recently, convolutional neural networks (CNN) have become very popular for image processing tasks [4-7].

The classical architecture of a convolutional neural network is used when an image reduced to a certain size is fed to the input, on which one object of a certain class is located. Defining this class is the task of a convolutional neural network. The satellite image shows many different objects of different sizes. To apply a network with a classical architecture, it is necessary to search for each object using a huge number of iterations with different sizes of the scanning window, then submit the found area to the input of the neural network. This approach is extremely slow and computationally expensive. In this work, we used the Mask-RCNN network architecture [8], which showed good results in image processing. Mask-RCNN belongs to the R-CNN family of networks [9-10] (figure 2).

The main idea of the R-CNN network [11] is to find areas (regions) in which objects for recognition are located, affine transformation of the generated regions to sizes suitable for processing by an ultra-precise neural network, extraction of the feature vector for each region, classification based on the method support vectors.

Figure 2. R-CNN family of networks.
The architecture behind the R-CNN has been significantly modified over time. Now, for the search for regions, extraction of feature maps, classification, separate neural networks are used, the training of which occurs simultaneously. Changes have also been made to the training algorithms for networks, which significantly speed up the training process and get more accurate results.

In this work, the software implementation of the Mask-RCNN network is made in Python 3.6 using the TensorFlow 1.13 library. The Mask-RCNN neural network model is implemented in the MaskRCNN class, when creating its object, the Build () method is called, in which the keras model is directly built. The method of its construction differs depending on the mode: training or determination (detection).

First, the input layers are initialized, or rather the shapes of their tensors. When training, these are the image (input_image), anchor matches (input_rpn_match) and bounding boxes (input_rpn_bbox) in RPN, target (obviously true) class identifiers (input_gt_class_ids), bounding boxes (input_gt_boxes) and masks (input_gt_masks) of the model. In real network operation, the image and anchors are determined.

Then there is the “backbone” of Mask-RCNN, namely, ResNet-FPN, RPN and the layer for generating the proposed regions (ProposalLayer). When training before RPN, anchor pyramids are generated for a given image shape using the GetAnchors () method. At the RPN output, a classifier is built by the softmax activation function (rpn_classifier_logits), the classifier probability (rpn_probabilities), and the regressor (rpn_bbox_regressor). The last two output parameters and anchors are fed to the input of the ProposalLayer, the output is the proposed regions (rpn_rois).

In the training mode of the model, its “head” is constructed as follows. Target detections are generated: rois, target_class_ids, target_bbox, target_mask by creating an object of the DetectionTargetLayer class. Then the classifier, regressor and masks of the Mask-RCNN model are determined using FPN and RoI Align. Next, the values of the loss functions rpn_class_loss, rpn_bbox_loss, class_loss, bbox_loss and mask_loss are calculated. As a result, the model is assembled using Keras tools.

In the detection mode, the model classifier and regressor are first determined using FPN and RoI Align. An object of the DetectionLayer class is created in order to refine classified offers and filter out overlaps. After that, masks are formed for the obtained objects also using FPN and RoI Align.

3. Preparing a set of images for training
For high-quality training of a neural network, a sufficiently large set of labelled images is required, i.e. images with the necessary objects highlighted. There are ready-made training kits such as ImageNet, MS COCO and others. But the freely distributed sets do not contain the data we need - satellite images with annotated roads, houses and other objects. Therefore, when implementing a system for automatic construction of a road network, we had to prepare a training sample independently. In this case, the set of images turned out to be more than modest (42 images), since annotating images is a laborious process.

There are many tools for annotating images, but they are not all designed to work with multispectral satellite imagery, which is .tif format, four channels and 16 bit depth. For this reason, the author had to sacrifice the quality of the images and convert them to RGB-images of the .png format and 8-bit depth using the ArcMap software product. The VGG Image Annotator (VIA) was used as a tool for annotating the resulting images, since it supports the .png format (not only .jpg and .bmp), is convenient and easy to use, and can save annotations of the easily processed VIA standard in the. json. An example of image annotation is shown in figure 3.
Figure 3. An example of image annotation in VGG Image Annotator.

4. Post-processing of images received from the neural network
After the operation of the neural network, a network of roads is highlighted in the images, however, the image has road breaks and other defects (figure 4). To eliminate errors, additional processing of the resulting image is carried out. This processing is done in two stages:

- using the Hough transformation, straight lines are found, and their gaps are eliminated;
- methods of mathematical morphology are applied.

Figure 4. Highlighting the road network in the image: (a) original image; (b) the result of the Mask-RCNN network.

In the simplest case, the Hough transform [12] is a linear transformation for finding lines. For convenience, polar coordinates \((r, \theta)\), are used, where the parameter \(r\) - is the length of the radius vector of the point on the line closest to the origin (i.e., the normal to the straight line drawn from the origin), and \(\theta\) is the angle between this vector and the abscissa. The straight line is represented as a point with coordinates \((r, \theta)\) in the parameter space (phase space, Hough space). If in the figure the straight lines
passing through two different points intersect in Hough space, then the coordinate of their intersection determines the parameters of the straight line passing through both points. Phase-space intersection analysis for image points eliminates discontinuities in lines connecting road points.

Mathematical morphology [13] is used to analyze and process geometric structures in images and is based on set theory. The processed binary image is considered as a set of pixels. The methods of mathematical morphology are used for pixel-by-pixel image processing by some primitive, with the help of which one or another operation is performed. The basic operations are erosion (narrowing) and dilatation (expansion). As the names suggest, as a result of the application of erosion, the thickness of the lines in the image decreases, and after dilatation, it expands. In this work, the closure operation is used, which consists in the sequential execution of first dilatation and then erosion. As a result, the image becomes more solid, breaks in lines are eliminated or become smaller (figure 5).

![Figure 5](image_url)

**Figure 5.** Result of the closing operation: (a) original image; (b) the processed image.

Post-processing of images obtained after using the Mask-RCNN neural network made it possible to significantly improve the result of detecting roads on satellite images (figure 6).

![Figure 6](image_url)

**Figure 6.** Highlighting the road network in the image: (a) the result of the Mask-RCNN network; (b) post-processing result.
5. Conclusion
The paper implements a method for identifying a network of highways on satellite images, based on a convolutional neural network Mask-RCNN. This neural network showed good recognition results, however, due to the small data set for training the network, the constructed images contain defects.

The developed algorithm can be used in other tasks of image segmentation. The recognized images will allow to build a graphological map of city roads and perform traffic flow control tasks.

Acknowledgments
This work was supported by the Ministry of Science and Higher Education of the Russian Federation, Supplementary Agreement No. 075-02-2020-1529/1 dated April 21, 2020.

References
[1] Kirthika A and Mookambiga A 2011 Automated road network extraction using artificial neural network IEEE-International Conference on Recent Trends in Information Technology 1061-5
[2] Mnih V and Hinton G E 2010 Learning to detect roads in high-resolution aerial images Computer Vision – ECCV 2010, Lecture Notes in Computer Science 6316 210-23
[3] Mnih V 2013 Machine learning for aerial image labeling Doctor of Philosophy Graduate Department of Computer Science University of Toronto
[4] Convolutional_neural_network http://en.wikipedia.org/wiki/Convolutional_neural_network
[5] Saha S A comprehensive guide to convolutional neural networks the eli5 https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53
[6] Khan S, Rahmani H, Shah S A A, Bennamoun M, Medioni G and Dickinson S 2018 A Guide to Convolutional Neural Networks for Computer Vision (Morgan & Claypool)
[7] Karim M R, Sewak M and Pujari P 2018 Practical Convolutional Neural Networks : Implement Advanced Deep Learning Models Using Python (Birmingham, Packt Publishing)
[8] Johnson J W 2018 Adapting Mask-RCNN for automatic nucleus segmentation IEEE Conference on Computer Vision and Pattern Recognition 7
[9] A brief history of CNNs in image segmentation: from R-CNN to Mask R-CNN https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4
[10] Object detection for dummies 3: R-CNN Family https://lilianweng.github.io/lil-log/2017/12/31/object-recognition-for-dummies-part-3.html
[11] Girshick R, Donahue J, Darrell T and Malik J 2014 Rich feature hierarchies for accurate object detection and semantic segmentation IEEE Conference on Computer Vision and Pattern Recognition 580-7
[12] Lee S Lines detection with Hough transform: An algorithm to find lines in images https://towardsdatascience.com/lines-detection-with-hough-transform-84020b3b1549
[13] Serra J 1982 Image Analysis Mathematical Morphology (Academic Press, London)