IMPROVING THE ROAD TOPOLOGICAL RELATION BASED ON A ROAD INTERSECTION DETECTION METHOD

Jun He[*], *Jinpeng Li*[*], *Weijia Li*[†]

School of Geospatial Engineering and Science, Sun Yat-sen University,
Zhuhai, Guangdong, 519082, China

ABSTRACT

Road network vector extraction has important practical significance for travel navigation, urban planning and many other applications. At present, road network vector extraction mainly involves segmenting road information from remote sensing images, converting raster road network information into vector road network data through vectorization method, and obtaining the final results by modifying the topological relationship manually. The current road extraction method is difficult to obtain road intersection information, especially when roads are stacked. Therefore, although relatively complete segmentation results can be obtained, it is difficult to show correct topological relations in vectorization. Therefore, this paper proposed a method combining object detection and node iterative search to correct the topological relationship of stacked road vectors. Among three mainstream detection frameworks, i.e., Faster R-CNN, Libra R-CNN, and YOLO v7, YOLO v7 achieved the best results in intersection object detection, which can reach 0.931 mAP_{50} and 0.585 $mAP_{50:95}$ for the test set. In addition, the road vector modified by node iteration search showed a more real road topology relationship in the shortest path analysis.

Index Terms— Road vectorization, object detection, High-resolution remote sensing images, Road topology.

1. INTRODUCTION

Road network data is a basic data type for current geographic information system analysis. Extracting road network plays a significant role in travel navigation, urban planning, and geographic information update. At present, road network information is usually represented by two kinds of data structures: raster road network data and vector road data. Road vector data plays a key role in the current geographic information system, therefore the correctness of the road vector information plays an important role in the construction planning, path analysis and other aspects of the geographic information system. The current method of extracting vector data of road network is to segment the road information from the image by combining high-resolution remote sensing images and using methods such as deep learning, and then convert the raster road network information into vector road network data by vectorization method, and finally modify the topological relationship by manual means.

In recent years, road extraction methods have been developed in two aspects: pixel-level road extraction based on semantic segmentation and road centerline tracking. Segmentation-based methods consider the road as a two-tuple semantic segmentation problem to distinguish between background and foreground. Zhang et al. proposed a neural network to extract road regions using residual module and U-Net backbone [1]. Zhou et al. proposed a semantic segmentation network D-LinkNet for road extraction from high-resolution satellite images, which can deal with the narrow, complex and long-span characteristics of roads to a certain extent [2]. Mei et al. implemented the connectivity attention network CoANet by strip convolution and introducing attention mechanism module [3]. For road centerline tracking, Cheng et al. proposed a cascade network to predict the road surface and road centerline [4]. Bastani et al. proposed RoadTracer, an iterative search strategy based on CNN decision function [5]. The centerline is tracked on the aerial image and expanded into the road surface with certain width information according to a given threshold. However, for both pixel-level road segmentation based on semantic segmentation and centerline extraction methods, although the integrity of the road can be preserved, the obtained segmentation results cannot cope with the situation that multiple roads are stacked and decide whether it can be passed.

For road vectorization, it can be regarded as the vectorization of raster data. At present, many scholars have proposed relevant methods, such as the method based on node search proposed by Shen et al., which first generates arcs from nodes, and then generates polygons according to the arcs [6]. Xu et al. used the priority principle of candidate points to determine the true candidate points by excluding internal pixel corners and pseudo-adjacent pixel corners, tracking the vertices of the boundary polygon, and quickly vectorizing the patches [7]. Wang et al. used morphological processing, image segmentation and other techniques to extract the region of interest, and then obtained the target boundary by edge detection, and

[*]Equal Contribution.
[†]Corresponding author.

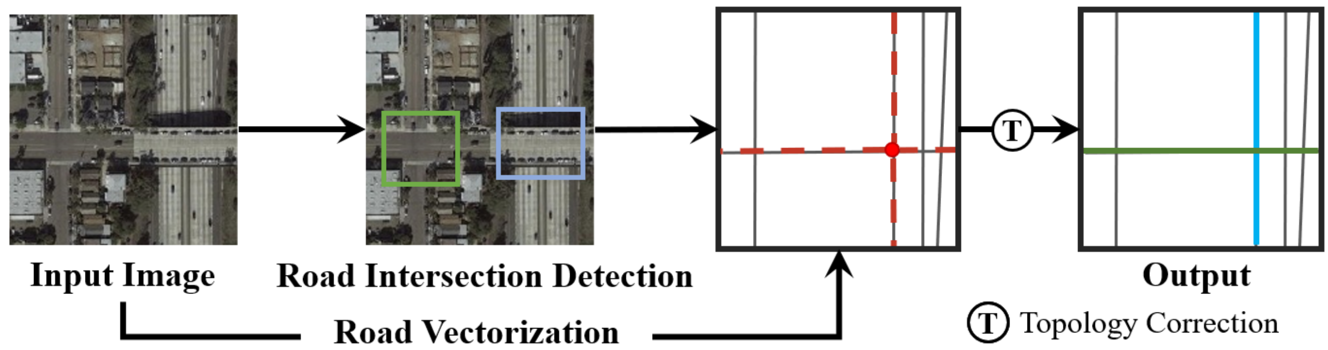


Fig. 1. Overview of our method, including road intersection detection, road vectorization and topology correction. Note that the road vectorization is based on the ground truth of RoadTracer dataset.

then vectorized the target [8]. However, the vectorization of roads needs to be based on the result map extracted from the road, which is usually a binary image, and it cannot obtain the accessibility information at the road intersection. Therefore, the correct topological information can not be obtained by vectorization in the road stacking area.

To solve the above problems, as shown in Figure 1, this paper proposes a method to improve the road topological relation based on a road intersection detection method. First, the positions of normal road intersections and stacked road intersections are obtained by object detection methods. Then the stacked road intersections are matched with the nodes of the preliminary road vectorization. Finally, an iterative correction method based on road nodes is used to modify the wrong topology information and obtain the correct road vector data.

2. METHODOLOGY

2.1. Road Intersection Detection

Based on the high-resolution aerial images provided by RoadTracer dataset, we established a new object detection dataset for the identification of ordinary and stacked intersections. We crop the images in the above dataset into sub-images, and then manually annotate the images including the intersections according to the morphological characteristics of the stacked intersections and ordinary intersections. Finally, all the annotated images are divided into training set, test set and validation set according to a certain proportion, and we ensure that there are no common regions between different subsets.

In order to verify the accuracy difference between different methods, Faster R-CNN [9], Libra R-CNN [10] and YOLO v7 [11] are used as the object detection models for detecting the true and false intersections. For the network results with RCNN as the framework, ResNet101-FPN was used as the object detection baseline, while for YOLO v7, the most basic YOLO v7 pre-training model was used for network training.

2.2. Topology Correction

By vectorizing the ground truth road mask, we obtain the initial road vector, which is stored as a set containing all edges. Therefore, a node connecting multiple edges must appear in the node sets of multiple edges. According to the above characteristics, we first match the minimum distance between the road node and the center point of the detection box of the stacked intersection, and then judge the micro-direction of the road node according to the angle between two road sections at the node. By judging the micro-direction adjacent to the node, the road connectivity is judged, and the road sections with the largest micro-direction angle are connected. If the node is connected to more than one road, the corrected results are brought back to the original detection box successively for topology correction, until there is no node at the stacked intersection.

3. EXPERIMENTS

3.1. Dataset and Implementation Details

The RoadTracer dataset consists of 15 aerial images of 8192×8192 pixels and 300 aerial images of 4096×4096 pixels, each with a spatial resolution of 0.6m. We crop 300 small aerial images into 640×640 pixel sub-images, and manually label the normal road intersections and stacked road intersections with the labels of True Crossing and False Crossing, respectively. Then we delete the images without label information. A total of 1722 datasets with at least one label were obtained. We split the training, validation, and test sets using a ratio of 4:1:1, ensuring that there are no overlapping regions between the different subsets, resulting in a training set with 1148 images and a validation and test set with 287 images. Finally, we use horizontal and vertical flipping, random brightness adjustment (original brightness ±20%) to perform data augmentation on the training set images, and obtain the final training set of 3444 images. The dataset is formatted into MS COCO and YOLO data formats

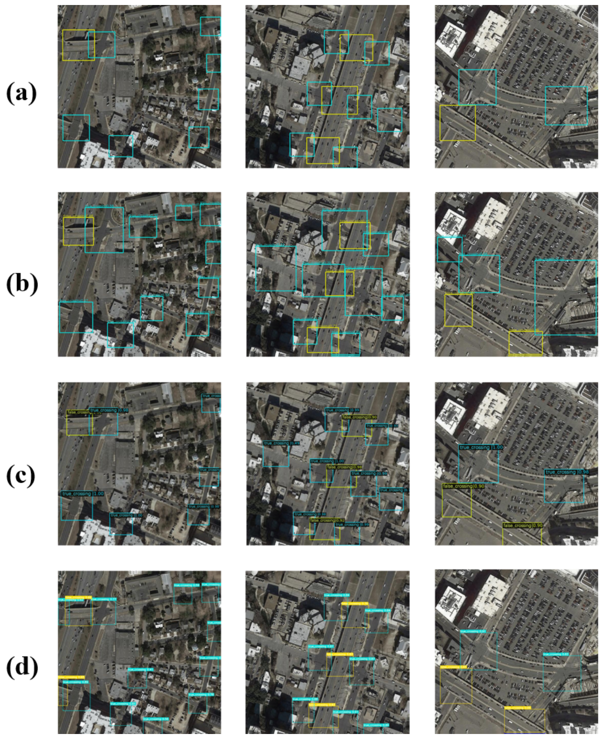


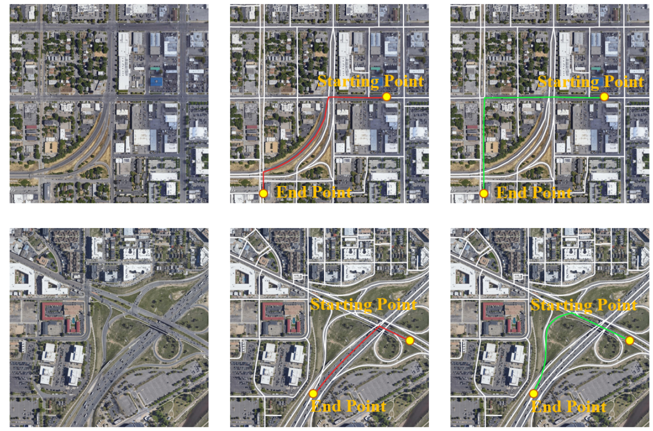
Fig. 2. Examples of the road intersection detection results obtained from different methods. The cyan and yellow bounding boxes denote the true and false intersections, respectively. (a) Ground Truth. (b) Faster R-CNN. (c) Libra R-CNN. (d) YOLO v7.

for training.

3.2. Road Intersection Detection

The training of the object detection model is mainly implemented based on Detectron [12] and MMDetection [13]. In the model training phase, we set the initial learning rate as 1×10^{-4}, and gradually decrease linearly with the iteration, but the minimum learning rate is limited to 1×10^{-6} to avoid the learning rate is too small to cause too slow convergence results. Limited by the memory size, we set the Batch Size of Faster R-CNN/Libra R-CNN/YOLOv7 to 8/2/4 and the number of epochs to. Figure 2 shows the results of road intersection detection.

The detection results of road intersection targets are shown in Table 1, where Montreal and Paris are the two cities in the RoadTracer dataset and belong to the test set of our intersection detection dataset. From the results, for the whole test set, the detection result of YOLO v7 reaches 0.931 in terms of mAP_{50}, and 0.585 in terms of $mAP_{50:95}$. For Montreal city with many stacked intersections, the detection scores of the three detection methods are lower than the overall test set. This is because the numbers of labels on the training set have great discrepancy between stacked



(a) (b) (c)

Fig. 3. Examples of the shortest path analysis result. (a) Input image, (b) the shortest path analysis result with uncorrected road vector, (c) the shortest path analysis result with corrected road vector.

intersections and ordinary intersections, which leads to worse detection results of stacked intersections than ordinary intersections. In the city of Paris, there is a large number of ordinary intersections and almost no stacked intersections, which makes the detection score obviously high.

3.3. Intersection Topology Correction

In this paper, we vectorize the ground truth road mask provided by RoadTracer to obtain a preliminary vector file, and then correct the topology of the obtained vector file according to the method mentioned in Section 2.2 in the paper, and finally analyze the obtained vector file for the shortest path that is commonly used in geographic information system. From the results Fig. 3, the road vector obtained by the method proposed in this paper has a more complete topology structure. The uncorrected road vector shows the characteristics of turning at the stacked intersection, which is not consistent with the real situation. The corrected road network avoids this problem and realizes a more scientific and reasonable road accessibility analysis for shortest path planning.

4. CONCLUSION

This paper provides a new method for road vector correction. Aiming at the problem of vector correction at stacked intersections, this paper proposes a road vector correction framework combining object detection and node iterative update, and provides a set of object detection datasets for road stacked intersections and normal intersections detection. The results show that YOLO v7 has better performance on our dataset, of which the mAP_{50} and $mAP_{50:95}$ on the test set reach 0.931 and 0.585, respectively. It is worth mentioning that, for the

Table 1. Comparison of different methods for road intersection detection in Montreal and Paris

Method	$mAP_{50}^{Montreal}$	$mAP_{50:95}^{Montreal}$	mAP_{50}^{Paris}	$mAP_{50:95}^{Paris}$
Faster R-CNN	0.553	0.291	0.912	0.380
Libra R-CNN	0.788	0.481	**0.975**	**0.597**
YOLO v7	**0.877**	**0.511**	0.968	0.498

area where the numbers of stacked intersections and ordinary intersections have a large discrepancy, Libra R-CNN effectively solves the problem of sample imbalance and improves the test performance of the model in this area. At the same time, the vector file output by our method can more correctly express the real topological relationship of the road, and show more realistic and objective results in the shortest path analysis, reachability analysis and other GIS related analysis.

5. ACKNOWLEDGMENTS

This work was supported partially by National Natural Science Foundation of China (No. 42201358).

6. REFERENCES

[1] Z. Zhang, Q. Liu, and Y. Wang, "Road extraction by deep residual u-net," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 5, pp. 749–753, 2018.

[2] L. Zhou and C. Zhang, "Linknet with pretrained encoder and dilated convolution for high resolution satellite imagery road extraction. 2018 ieee," in *CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 192–1924, 2018.

[3] J. Mei, R.-J. Li, W. Gao, and M.-M. Cheng, "Coanet: Connectivity attention network for road extraction from satellite imagery," *IEEE Transactions on Image Processing*, vol. 30, pp. 8540–8552, 2021.

[4] G. Cheng, Y. Wang, S. Xu, H. Wang, S. Xiang, and C. Pan, "Automatic road detection and centerline extraction via cascaded end-to-end convolutional neural network," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 55, no. 6, pp. 3322–3337, 2017.

[5] F. Bastani, S. He, S. Abbar, M. Alizadeh, H. Balakrishnan, S. Chawla, S. Madden, and D. DeWitt, "Roadtracer: Automatic extraction of road networks from aerial images," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4720–4728, 2018.

[6] Z. Shen and R. Wan, "A new approach for converting raster to vector—node searching. j," *Image and Graph*, vol. 3, pp. 318–321, 1998.

[7] Y. Xu, C. Cui, and W. Tao, "A simple boundary vectorization algorithm for raster data polygons," *Science of Surveying and Mapping*, vol. 39, no. 139-141+129, 2014.

[8] M. Wang, R. Chen, J. Huang, R. Li, and S. Zheng, "Vectorization algorithm for raster data based on boundary-searching," *Intelligent Computer and Applications*, vol. 9, pp. 255–228+233, 2019.

[9] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015.

[10] J. Pang, K. Chen, J. Shi, H. Feng, W. Ouyang, and D. Lin, "Libra r-cnn: Towards balanced learning for object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 821–830, 2019.

[11] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv preprint arXiv:2207.02696*, 2022.

[12] Y. Wu, A. Kirillov, F. Massa, W.-Y. Lo, and R. Girshick, "Detectron2." https://github.com/facebookresearch/detectron2, 2019.

[13] K. Chen, J. Wang, J. Pang, Y. Cao, Y. Xiong, X. Li, S. Sun, W. Feng, Z. Liu, J. Xu, Z. Zhang, D. Cheng, C. Zhu, T. Cheng, Q. Zhao, B. Li, X. Lu, R. Zhu, Y. Wu, J. Dai, J. Wang, J. Shi, W. Ouyang, C. C. Loy, and D. Lin, "MMDetection: Open mmlab detection toolbox and benchmark," *arXiv preprint arXiv:1906.07155*, 2019.