Building Extraction from Satellite Images Using Mask R-CNN with Building Boundary Regularization

Kang Zhao[1], Jungwon Kang[1], Jaewook Jung[2], Gunho Sohn[1]

[1]York University
4700 Keele Street, Toronto,
ON M3J 1P3, Canada
{kangzhao, jkang99, gsohn}@yorku.ca

[2]Thales Canada
105 Moatfield Drive, North York,
ON M3B 0A4, Canada
jwjung00@gmail.com

Abstract

The DeepGlobe Building Extraction Challenge poses the problem of localizing all building polygons in the given satellite images. We can create polygons using an existing instance segmentation algorithm based on Mask R-CNN. However, polygons produced from instance segmentation have irregular shapes, which are far different from real building footprint boundaries and therefore cannot be directly applied to many cartographic and engineering applications. Hence, we present a method combining Mask R-CNN with building boundary regularization. Through the experiments, we find that the proposed method and Mask R-CNN achieve almost equivalent performance in terms of accuracy and completeness. However, compared to Mask R-CNN, our method produces better regularized polygons which are beneficial in many applications.

1. Introduction

Automatic extraction of buildings from massive satellite images is still a challenging problem. The DeepGlobe Building Extraction Challenge (DG-BEC)[1] has encouraged people to present automated methods for extracting buildings from satellite images. The DG-BEC provides satellite images of four urban cities including Las Vegas, Paris, Shanghai, and Khartoum. There are four types of images including panchromatic (PAN), 8-band multi-channel (MUL), pan-sharpened version of RGB bands from the multispectral product (RGB-PanSharpen), and lastly pan-sharpened version of MUL (MUL-PanSharpen). The DG-BEC poses the problem of localizing all building polygons in the given images.

The problem has been normally tackled by the combination of two processes: (i) *segmentation*, the extraction of building regions from the given area, and (ii) *instantiation*, the identification of individual buildings. Two processes have been combined in a different order. One approach [1, 2] is to perform the segmentation first, followed by instantiation. Here, the whole area in an image is segmented into building and non-building regions. Then, each individual building is identified by grouping connected pixels in a building region. The other approach [3, 4], commonly called *instance segmentation*, is to perform instantiation first and then segmentation. In this approach, each individual building is detected in a bounding box. Then, each bounding box is segmented into building and non-building regions. Because this approach is suitable for handling urban areas where separating adjacent buildings is needed, we follow this approach of the instance segmentation.

As a base network, we adopt Mask R-CNN [4] due to its simplicity in the network structure and hyper-parameter tuning. After running Mask R-CNN, we produce polygon points of each individual building. These polygon points correspond to building boundaries and normally have irregular shapes. However, we observe that most of building footprints have regularized boundaries. Moreover, such irregularly shaped polygons often cause difficulties to be directly applied to many cartographic and engineering applications. Motivated by this observation we then formulate the problem as the creation of regularized polygons for buildings. Therefore, we utilize a building boundary regularization method, which is adopted from works [5, 6]. Then, we convert polygons generated by Mask R-CNN into the regularized polygons. In addition, rather than training one universal building extractor, we make four building extractors, where each building extractor is trained for handling a specific city.

2. Related Works

We divide building extraction into deep learning based-building labeling and building boundary regularization and investigate prior work in these two fields respectively.

[1] http://deepglobe.org

2.1. Deep Learning-based Building Labeling

There have been recent research efforts in applying Convolution Neural Network (CNN) for high resolution satellite image labeling. However, challenges still exist in finding optimal architecture of CNN for the best solution to such problems. Mnih [7] created building classification datasets over Massachusetts, covering 340 km^2 and trained a CNN model for building labeling. Maggiori *et al.* [8] proposed a multi- layer perceptron approach to balance the trade-off between localization and classification for building labeling. By introducing a new cascaded multi-task loss and took the boundary distance into account, Bischke *et al.* [9] addressed the problem of preserving semantic segmentation boundaries in high resolution satellite imagery. In addition to creating new architecture, researchers also fused information from different sources. Marmanis *et al.* [10] combined the information from edge detection to produce explicit class boundaries for building extraction. The use of OpenStreetMap (OSM) data was investigated by Audebert *et al.* [11] to produce a coarse to fine solution for semantic labeling of satellite images.

2.2. Building Boundary Regularization

For recent research on building boundary regularization, Jung *et al.* [5] proposed a data-driven modeling approach to reconstruct 3D rooftop models at city-scale from airborne laser scanning (ALS) data. The focus of the proposed method is to implicitly derive the shape regularity of 3D building rooftops from given noisy information of building boundary in a progressive manner. Maggiori *et al.* [12] proposed a novel method, which formulated the polygonization problem into a mesh-based approximation of the input binary classification map. The regularization problem was also investigated in [13] applying a new CNN architecture which introduced the polygon boundary loss into the loss function.

3. Methodology

Our pipeline for building extraction is a combination of Mask R-CNN and polygon regularization, as in Figure 1. Given an input image, Mask R-CNN generates initial polygons for buildings. Then, by our polygon regularization method, the initial polygons are converted into regularized ones.

3.1. Mask R-CNN for Initial Polygon Generation

The Mask R-CNN [4] is an extension of Faster R-CNN [14], which adds a network branch to the original Faster R-CNN for predicting segmentation masks on each Region of Interest (RoI). The added branch is a small FCN [15] which is applied to each RoI and predicts a pixel-wise segmentation mask for building and non-building regions.

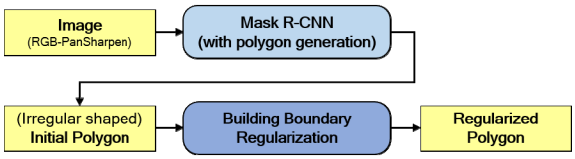


Figure 1. Our pipeline for building polygon generation.

By tracing the border of a building region, we get the initial polygon for the building.

Input: Among the given four types of images, we only use RGB-PanSharpen images considering that they have the highest resolution, sharpened characteristic and smaller memory size. The image has 3 channels and its image size is 650×650. However, as the input of the network, we enlarge the images from 650×650 to 1024×1024 to handle buildings in various scales.

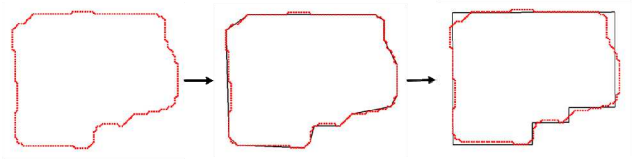
Network Configuration: The Mask R-CNN consists of two parts: (i) the convolutional backbone architecture used for feature extraction over an entire image, and (ii) the network head for classification, bounding box recognition and mask prediction that is applied separately to each RoI. We apply ResNet-101-FPN as the backbone architecture and Faster R-CNN with ResNet as the head architecture. We follow the publicly open implementation[2] of the Mask R-CNN and adopt most of the hyper-parameters used for training the COCO dataset. However, the hyper-parameters MINI_MASK_SHAPE, MASK_SHAPE, which are used for improving the training speed, are found to largely affect the overall detection performance. The instance masks are resized to a smaller size, MINI_MASK_SHAPE to save loading memory and we use MASK_SHAPE as the size of the output masks. For the hyper-parameters, we therefore set [128, 128] and [28, 28] as MINI_MASK_SHAPE and MASK_SHAPE, respectively. The full list of the hyper-parameters will be posted in our project website[3].

Training: Inspired by the training process on COCO dataset in [4], we train the network through the following three phases: the first phase for training the head with 40 epochs, the second phase for training 4th and more rear stage of ResNet-101 with 20 epochs, and the last phase for training all layers with 20 epochs.

3.2. Building Boundary Regularization

The initial polygons produced by Mask R-CNN show irregular and noisy outlines due to the locality of pixel-wise labeling conducted by Mask R-CNN. To convert the initial polygons into regularized ones, we modify previous work [5, 6] to be applicable in an image domain for implicitly regularizing noisy building boundaries in an iterative manner. The boundary regularization process, described in Figure 2, takes the following steps: Initial modeling,

[2] https://github.com/matterport/Mask_RCNN
[3] https://github.com/yorku-ausml/deep_satellite_image_segmentation



(a) (b) (c)

Figure 2. Building boundary regularization process: (a) Initial polygon points (the red points) from the result of Mask R-CNN; (b) Simplified shaped polygons (the black lines) from Douglas-Peucker algorithm; (c) Polygons with regularized boundary (the black lines) from our algorithm.

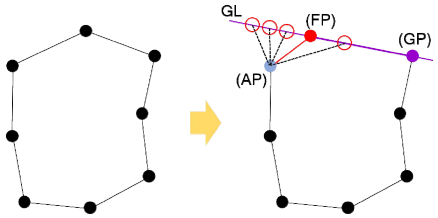


Figure 3. Examples of hypothesis generation from initial polygon points given at the left of the figure.

Hypothesis generation, and Minimum Description Length (MDL) optimization.

Initial Modeling: The initial polygon points are first converted into simplified shaped polygons, by the Douglas-Peucker (DP) algorithm [16]. A set of representative line slopes are estimated based on the results of DP, with which the initial polygon is adjusted by applying weighted least-square adjustment method.

Hypothesis Generation: A triplet of vertices are selected (non-selective to the selection order) from the initial polygon, as described in Figure 3. We label the triplet points as Anchor Point (AP), Floating Point (FP) and Guiding Point (GP) in a sequential order. Then, we generate two basis lines: Floating Line FL, which is a set of AP and FP, and Guiding Line GL, which is a set of GP and FP. A group of local hypothetical models are generated by moving FP along GL following the representative line directions estimated. We also allow the elimination of FP for hypothetical model generation. In this case, new FP and GP are selected by shifting the previously selected point triplet in a sequential order. Both clock-wise and counter-clockwise are selected to generate local model hypotheses for each point triplet.

MDL Optimization: MDL framework [6] is selected for determining an optimal model hypothesis among the generated candidate models. The description length (DL) of a model in MDL framework is decomposed into two parts: (i) model closeness favoring low residuals between boundary points extracted by boundary tracing algorithm and hypothesized model; (ii) model complexity favoring simpler model with respect to the number of vertices, the number of representative line slopes and closeness to orthogonal angles. The detail of the MDL encoder adopted in this study is described in [6]. The MDL optimization process is applied for determining the best model hypothesis locally over point triplet selected. Then, a globally optimized hypothesis is chosen by selecting a model to produce the minimum DL among all local optimum solutions. The same process is sequentially applied to all point triplets.

Method	F1 Score (Individual City)				Total F1 Score
	Las Vegas	Paris	Shang-hai	Khar-toum	
Nofto	0.787	0.584	0.520	0.424	0.579
Wleite	0.829	0.679	0.581	0.483	0.643
XD_XD	**0.885**	0.745	0.597	0.544	0.683
Mask R-CNN	0.881	**0.760**	**0.646**	**0.578**	**0.717**
Ours	0.879	0.753	0.642	0.568	0.713

Table 1. F1 Scores of building extraction results.

4. Experimental Results

4.1. Training and Testing

For training, we used a pre-trained model trained on the COCO dataset. It took three days for training the networks for all the cities using NVIDIA GeForce 1080 Ti. The dataset is divided into training (80%) and validation sets (20%). For testing, we ran our pipeline on about 3500 test images.

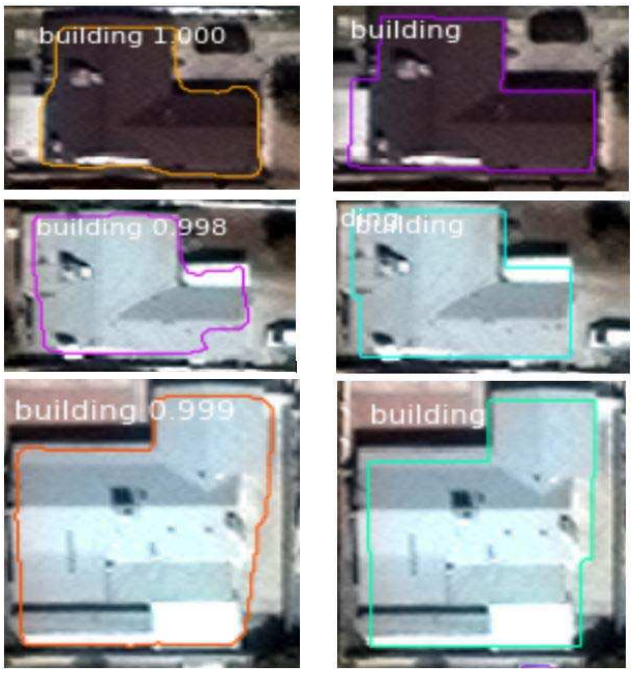
4.2. Evaluation

The performance of the algorithm is evaluated based on the Intersection over Union (IoU) metric. The DG-BEC provides F1 score, a harmonic average of precision and recall, combining the accuracy in the precision measure and the completeness in the recall measure. We analyze our results by the following aspects:

Accuracy and completeness of building extraction: The round 2 of the SpaceNet Building Detection Challenge[4] is chosen as a baseline result because both SpaceNet Challenge and DG-BEC use the same data, evaluation metric and evaluation code. Table 1 describes our F1 scores compared to that of the top3 winners (Nofto, Wleite, and XD_XD) of the SpaceNet Challenge. Our algorithm outperforms all the others in terms of F1 score.

Effect of building boundary regularization on building extraction: Table 1 also shows the comparison of F1 scores of Mask R-CNN and our method. Although MASK R-CNN shows slightly higher F1-score than our method, two scores from both methods are almost

[4] https://www.topcoder.com/spacenet



(a) (b)

Figure 4. Polygons results produced by (a) Mask R-CNN, and (b) our method.

(a) (b)

Figure 5. Extraction of special types of buildings: (a) small buildings in red circle, and (b) multiple buildings in close distance. Different colors are used to represent different building footprints extracted.

equivalent. However, our methods cut off the number of polygon points by 86%, thus dramatically simplified the generated polygons. These polygons can be directly imported to many applications. Figure 4 shows examples of the results from Mask R-CNN and our method. Our method produces obviously better representation of building footprints with more regular boundaries.

Handling special types of buildings: Extracting some special types of buildings such as small buildings and closely located buildings is challenging. As shown in Figure 5, our method can successfully recognize and localize them for both cases.

5. Conclusions

We present a building extraction method combining Mask R-CNN with building boundary regularization. The proposed method and Mask R-CNN produced almost equivalent F1 scores which are the evaluation metric from DG-BEC. However, compared to Mask R-CNN that generates irregular shaped polygons, our method produces regularized polygons, which are directly applicable to numerous cartographic and engineering applications.

Acknowledgements

We would like to acknowledge the supports from Natural Science and Engineering Research Council of Canada (NSERC) Discovery Program and Ontario Research Fund (ORF) – Intelligent Systems for Sustainable Urban Mobility (ISSUM) Program. Also, we would like to thank Dr. Connie Ko at York University who helped us to revise the manuscript.

References

[1] A. Kirillov, E. Levinkov, B. Andres, B. Savchynskyy, and C. Rother. Instancecut: from edges to instances with multicut. In *CVPR*, 2017.

[2] P. O. Pinheiro, T. Y. Lin, R. Collobert, and P. Dollár. Learning to refine object segments. In *ECCV*, 2016.

[3] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei. Fully convolutional instance-aware semantic segmentation. In *CVPR*, 2017.

[4] K. He, G. Gkioxari, P. Dollar, and R. Girshick. Mask R-CNN. In *ICCV*, 2017.

[5] J. Jung, Y. Jwa, and G. Sohn. Implicit regularization for reconstructing 3D building rooftop models using airborne lidar data. *Sensors*, vol. 17, no. 3, p. 621, 2017.

[6] G. Sohn, Y. Jwa, J. Jung, and H. Kim. An implicit regularization for 3D building rooftop modeling using airborne lidar data. In *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. I-3, no. September, pp. 305–310, 2012.

[7] V. Mnih. Machine Learning for aerial image labeling. University of Toronto, 2013.

[8] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez. High-resolution aerial image labeling with convolutional neural networks. *IEEE Trans. Geosci. Remote Sens.*, vol. 55, no. 12, pp. 7092–7103, 2017.

[9] B. Bischke, P. Helber, J. Folz, D. Borth, and A. Dengel. Multi-task learning for segmentation of building footprints with deep neural networks. *arXiv:1709.05932v1*, 2017.

[10] D. Marmanis, K. Schindler, J. D. Wegner, S. Galliani, M. Datcu, and U. Stilla. Classification with an edge: Improving semantic image segmentation with boundary detection. *ISPRS Journal. Photogramm. Remote Sens.,* vol. 135, pp. 158–172, Dec. 2018.

[11] N. Audebert, B. Le Saux, and S. Lefevre. Joint learning from earth observation and OpenStreetMap data to get faster better semantic maps. In *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017.

[12] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez. Polygonization of remote sensing classification maps by mesh approximation. In *ICIP*, 2017.
[13] N. Girard and Y. Tarabalka. End-to-end learning of polygons for remote sensing image classification. In *IEEE International Geoscience and Remote Sensing Symposium*, 2018.
[14] S. Ren, K. He, R. Girshick, and J. Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, 2017.
[15] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.
[16] D. H. Douglas and T. K. Peucker. Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartographica: The International Journal for Geographic Information and Geovisualization*, vol. 10, no. 2, pp. 112–122, 1973.