Joint Semantic-Geometric Learning for Polygonal Building Segmentation

Weijia Li[1,2*], **Wenqian Zhao**[3†], **Huaping Zhong**[4], **Conghui He**[4,5‡], **Dahua Lin**[1]
[1]CUHK-SenseTime Joint Lab, The Chinese University of Hong Kong
[2]Shanghai SenseTime Intelligent Technology Co., Ltd.
[3]The Chinese University of Hong Kong
[4]SenseTime Group Limited
[5]Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
{wjli,dhlin}@ie.cuhk.edu.hk, wqzhao@cse.cuhk.edu.hk, {zhonghuaping,heconghui}@senstime.com

Abstract

Building extraction from aerial or satellite images has been an important research problem in remote sensing and computer vision domains for decades. Compared with pixel-wise semantic segmentation models that output raster building segmentation map, polygonal building segmentation approaches produce more realistic building polygons that are in the desirable vector format for practical applications. Despite the substantial efforts over recent years, state-of-the-art polygonal building segmentation methods still suffer from several limitations, e.g., (1) relying on a perfect segmentation map to guarantee the vectorization quality; (2) requiring a complex post-processing procedure; (3) generating inaccurate vertices with a fixed quantity, a wrong sequential order, self-intersections, etc. To tackle the above issues, in this paper, we propose a polygonal building segmentation approach and make the following contributions: (1) We design a multi-task segmentation network for joint semantic and geometric learning via three tasks, i.e., pixel-wise building segmentation, multi-class corner prediction, and edge orientation prediction. (2) We propose a simple but effective vertex generation module for transforming the segmentation contour into high-quality polygon vertices. (3) We further propose a polygon refinement network that automatically moves the polygon vertices into more accurate locations. Results on two popular building segmentation datasets demonstrate that our approach achieves significant improvements for both building instance segmentation (with 2% F1-score gain) and polygon vertex prediction (with 6% F1-score gain) compared with current state-of-the-art methods.

Introduction

As a fundamental task for urban planning, disaster and environmental management, geographical information updating, etc., extracting building footprints from aerial or satellite images has been an important and popular research problem in both remote sensing and computer vision domains. Deep convolutional neural networks based segmentation models have become the state-of-the-art methods for building footprint extraction, which assign a semantic class to each pixel of an image. However, the output raster building masks generated from these segmentation models are not in the desirable format of realistic building polygons (with linear edges and specific angles). Complex post-processing procedures are required for converting the segmentation predictions into the vector building polygons for practical applications.

Motivated by this issue, many polygonal building segmentation approaches have been proposed to generate the vectorized outputs. Several studies proposed post-processing methods for simplifying the building segmentation contours. In (Zhao et al. 2018), a multi-step boundary regularization method was proposed to simplify the building instances predicted from Mask-RCNN. Li et al. (Li, Lafarge, and Marlet 2020) proposed a polygonal partition refinement method for vectorizing the output probability maps of a U-Net based model. These methods not only require a complex processing procedure, but also a perfect segmentation map to ensure the quality of the polygonization results. To solve the above limitations, a generative adversarial network based method was proposed in (Zorzi, Bittner, and Fraundorfer 2020) for regularizing the building segmentation maps. Although producing visually pleasing building polygons, the method consists of three separate networks and requires heavy training procedures regarding the hybrid losses of different network components.

The other category of polygonal segmentation methodologies directly predicts the polygon vertices of a building instance, using deep neural networks with CNN-RNN or GCN architectures. PolyMapper (Li, Wegner, and Lucchi 2019), a polygonal building segmentation approach, combines CNN with LSTM model to predict a polygon vertex at each time step. The method was extended from PolygonRNN (Castrejon et al. 2017), a semi-automatic polygonal segmentation method, and outperformed several instance segmentation methods (Mask-RCNN (He et al. 2017) and

*This work has been financially supported by fund of Shanghai Municipal Commission of Economy and Informatization (2019-RGZN-01015).

†This work was done during the author's internship at SenseTime Group Limited.

‡Corresponding author

PANet (Liu et al. 2018)) on CrowdAI mapping challenge dataset. However, despite its capability of producing desirable prediction for simple building polygons (with few vertices and edges), the RNN-based methods usually have difficulty in correctly predicting vertices for complex building polygons, producing vertices with wrong sequential order and self-intersections. Curve-GCN (Ling et al. 2019) is another semi-automatic polygonal segmentation method, which represents an object as a graph with a fixed number of vertex, and predicts an offset for each vertex simultaneously. Although achieving promising segmentation results for common datasets, the fixed vertex topology of Curve-GCN results in over redundant vertices for simple buildings and insufficient vertices for buildings with complex shapes.

In this work, we propose a novel polygonal building segmentation approach to address the above challenges. Our approach consists of three main components, i.e., a multi-task segmentation network, a vertex generation module, and a polygon refinement network, which can be summarized as follows:

- In our multi-task segmentation network, we design two additional tasks for leveraging extra geometric supervision for polygonal building segmentation, i.e., multi-class corner prediction and edge orientation prediction, which are trained jointly with the building semantic segmentation task.
- In the vertex generation module (VGM), we design a simple but effective method for transforming the building segmentation contour into a set of valid vertices. Through jointly utilizing the three types of outputs of the multi-task network, our vertex generation module is not only capable of filtering out redundant edges and vertices and remaining valid ones (even short edges), but also robust to imperfect building segmentation results.
- The polygon refinement network (PRN) further fine-tunes the coordinate of each polygon vertex. PRN regards the VGM generated vertices (with various topologies and proper sequence) as initial nodes of a graph, which effectively predicts a displacement for each node and produces the final building polygons with more accurate vertices.

Our proposed approach is evaluated by two popular building extraction challenge datasets. Compared with current state-of-the-art methods, our approach achieves much better polygonal building instance segmentation results, improving the F1-score by 2% in terms of building segmentation and 6% in terms of vertex prediction.

Related Work

Building Footprint Segmentation

As an important task in remote sensing and geographic information system domain, building footprint segmentation has been extensively studied for decades. Traditional building segmentation approaches were based on shadow index, edge regularity, or line fragment, etc. (Sun, Christoudias, and Fua 2014). In recent years, pixel-wise semantic labeling models based on deep learning have become state-of-the-art methods for building footprint segmentation. Semantic segmentation and instance segmentation models have been widely explored for building segmentation tasks (Li et al. 2019). Among these models, U-Net based architectures have achieved excellent performances in several building extraction challenges such as CrowdAI (Mohanty 2018) and SpaceNet (Van Etten, Lindenbaum, and Bacastow 2018). On the other hand, several recent studies proposed active contour based approaches for building segmentation (Cheng et al. 2019; Marcos et al. 2018; Gur, Shaharabany, and Wolf 2020), and most of these approaches are designed for single building extraction of which the input images have already been cropped by ground truth bounding boxes.

Although pixel-wise segmentation and active contour based methods obtain promising building extraction accuracies, there exist great discrepancy between the outputs of these methods and the desired format of building polygons. The building outlines obtained from these methods are often in a curved format, while the desired building polygons have linear contours with a limited number of edges and vertices. Substantial post-processing procedures are required before the building segmentation predictions can be utilized in practical applications.

Polygonal Instance Segmentation

Post-processing based Polygonization Post-processing based polygonization methods have been widely used for simplifying the segmentation contours of buildings or other object types. Generally, semantic segmentation or instance segmentation results are post-processed via traditional contour simplification methods, such as Douglas-Peucker (Wu and Marquez 2003), polyline decimation (Dyken, Dhlen, and Sevaldrud 2009), etc. (Zhao et al. 2018) proposed a multi-step boundary regularization method to regularize the building instances predicted from Mask-RCNN and generate the simplified building polygons. In (Li, Lafarge, and Marlet 2020), a polygonal partition refinement method was proposed for vectorizing buildings and general objects from segmentation maps. These methods usually require complex procedures of multiple processing steps to generate the final polygons. Their performance heavily depends on the quality of the segmentation map, which deteriorates seriously when the segmentation map is not perfect. (Zorzi, Bittner, and Fraundorfer 2020) designed an approach to regularize the building segmentation maps via a generative adversarial network, which requires a multi-stage training procedure for optimizing different network components.

Deep Neural Network based Vertex Prediction Several methodologies use deep neural networks to directly predict vertices of a polygon. Polygon-RNN (Castrejon et al. 2017) is a semi-automatic polygonal annotation method that directly predicts a polygon vertex at each time step using a CNN-RNN architecture. It was further improved by (Acuna et al. 2018) and extended for automatic building segmentation task (Li, Wegner, and Lucchi 2019). These RNN-based methods usually achieve desirable prediction results for buildings with simple shapes. However, the sequential manner of the recurrent model limits its capability of correctly predicting vertices for complex building polygons.

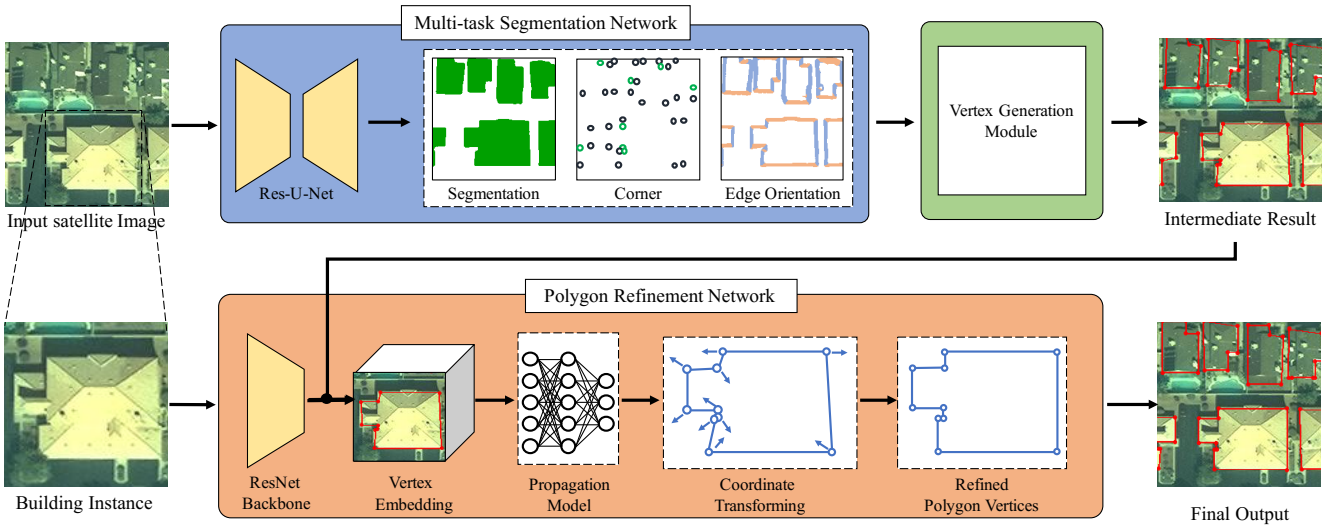

Figure 1: An overview of our proposed method. Taking a remote sensing image as input, the multi-task segmentation network outputs a building segmentation map, a corner prediction map and an edge orientation prediction map. The vertex generation module converts the former three types of outputs into a set of polygon vertices. The polygon refinement network predicts a displacement for each vertex and produces the final building polygons.

Several other vertex-based methods predict all vertices simultaneously in a regression manner. Initial vertices are selected uniformly from a segmentation mask contour using a given distance (Liang et al. 2020), or from an initialization with a fixed vertex quantity (Ling et al. 2019). These methods usually generate over redundant vertices for buildings with simple shape and insufficient vertices for buildings with complex contour.

Multi-task Learning

Multi-task learning has been proved as an effective strategy for building footprint segmentation. A distance transform prediction task was introduced in (Bischke et al. 2019) to improve the building boundary prediction results. Similarly, (Mahmud et al. 2020) proposed a multi-task learning method for predicting building outlines and their heights via jointly learning a modified signed distance function from the building boundaries with other types of supervisions. Several studies exploit orientation or direction related supervisions for various segmentation tasks. In (Bischke et al. 2019), a direction map was jointly learned with other two tasks for road boundary extraction. SegFix (Yuan et al. 2020) also learned the direction away from the boundary pixel to an interior pixel in order to refine the segmentation boundary. A recent study introduced a frame field learning task for polygonal building segmentation (Girard et al. 2020), which defined the frame as two directions denoted by complex numbers for each pixel.

Different from the above distance and direction prediction tasks that were designed for improving the segmentation or boundary prediction results, the edge orientation prediction task in our approach is proposed for directly producing accurate building vertices, which can be further refined by our polygon refinement network.

Methods

Framework Overview

The overall framework of our proposed approach is demonstrated in Figure 1, which consists of three main components: (1) A multi-task segmentation network; (2) A vertex generation module; (3) A polygon refinement network. Taking a large-scale remote sensing image with multiple building instances as input, the multi-task segmentation network is designed for joint semantic and geometric learning of three tasks, i.e., pixel-wise building segmentation, multi-class corner prediction, and edge orientation prediction. Then the vertex generation module effectively utilizes the three types of outputs of the multi-task network, and converts the building segmentation contour into a set of valid vertices. The polygon refinement network takes the output vertices of the former component as the initial nodes of a graph and predicts a displacement for each node, producing the final building polygons with more accurate vertices. In the following, we first give the definitions of the corners and edge orientations, which will be used in the multi-task segmentation network. Then we introduce each of the three main components of our proposed approach. The implementation details are described at the end of this section.

Representation of Corners and Edge Orientations

We design a multi-class corner prediction task and an edge orientation prediction task to leverage extra geometric supervision for polygonal building segmentation. Different from existing methods that simply classify each pixel into background or building corners, we define each pixel as one of three types, i.e., background, convex corners and concave corners, in order to avoid predicting multiple adjacent corners (connected by short edges) as one corner. If the interior

angle of a polygon vertex is smaller than $180°$, the vertex will be defined as a convex corner (denoted by black circles in Figure 1); otherwise it will be defined as a concave corner (denoted by green circles in Figure 1).

The edge orientation is a beneficial property for polygonal building segmentation in many aspects: (1) Orientation is important information of real-world objects in remote sensing images (Ding et al. 2019), especially for artificial objects such as buildings and roads (Girard et al. 2020; Bischke et al. 2019); (2) It is an enumerable property that can be easily formulated as a classification issue and learned via deep neural networks; (3) It can be effectively utilized for converting the semantic representation of a building instance into topology representation. For each pixel on the building edges, its orientation class is obtained via discretizing the orientation angle of the edge into a class. For each pixel at the building corners, its orientation class is randomly assigned with the one of its neighbor pixel of an edge. For each pixel that is not on any edges or corners, its orientation class is defined as zero.

Multi-task Segmentation Network

Our multi-task segmentation network is based on Res-U-Net architecture. The U-Net based models have achieved promising performance in many building segmentation challenges and studies (Demir et al. 2018). The three tasks are all formulated as pixel-wise classification issues and trained jointly with the cross entropy loss (denoted by L) according to formula 1:

$$L = -\sum_{i=1}^{N}\sum_{c=1}^{C} y_{i,c} \times log(p(y_{i,c})) \quad (1)$$

where C is the number of classes of the corresponding task; N is the number of pixels of an image; $y_{i,c}$ is a binary indicator that equals 1 if c is the ground truth label of pixel i or 0 in other cases; $p(y_{i,c})$ is the predicted probability that pixel i belongs to class c. We use L_{seg}, L_{corner} and L_{orient} to denote the classification loss of building area segmentation, multi-class corner prediction, and edge orientation prediction tasks, respectively, and use λ_1, λ_2 and λ_3 to denote the weight of each task. The total loss L_{total} of the three tasks can be summarized as:

$$L_{total} = \lambda_1 L_{seg} + \lambda_2 L_{corner} + \lambda_3 L_{orient} \quad (2)$$

Vertex Generation Module

Based on the outputs of the multi-task segmentation network, we design a vertex generation module (VGM) to transform the raster segmentation masks into polygon vertices. For each predicted building instance, the prediction of the building segmentation task is converted to the mask contour with a width of one pixel. We extract every pixel on the mask contour through dense sampling in an anticlockwise order, constituting a set of initial vertex candidates. Based on the above predictions and two user defined thresholds, i.e., the corner probability threshold (T_{cor}) and the orientation difference threshold (T_{ori}), we define a corner criterion and an edge orientation criterion for jointly selecting the valid vertices from the initial vertex set.

For the corner criterion, the vertices with a corner prediction probability smaller than T_{cor} are removed from the initial vertex candidates. Then each group of adjacent vertices are further converted into one valid vertex, i.e., the local maximum of the corner prediction probability. Meanwhile, in our edge orientation criterion, we calculate the absolute difference of the orientation angle between two neighbouring vertices for each initial vertex candidate. The vertices with an absolute difference greater than T_{ori} are selected as the valid vertices. The valid vertices selected by the corner criterion and the edge orientation criterion are combined into a union vertex set and each group of adjacent vertices are further merged into one vertex, constituting the final output of the vertex generation module.

Polygon Refinement Network

Backbone and Vertex Embedding We adopt a variant of ResNet50 (He et al. 2016) as the backbone of our polygon refinement network (PRN). As shown in Figure 1, the ResNet backbone is served as an encoder for extracting features from the input image with one building instance, producing feature map that will be further used for vertex embedding. Following (Acuna et al. 2018), we add a skip-connection structure to up-sample and concatenate the feature maps of four skip layers. The size of the final feature map for vertex embedding is 1/2 of the original scale, which guarantees a high resolution for accurately representing the vertex coordinates and a proper receptive field for predicting the vertex offsets in a balanced manner. The input dataset of our polygon refinement network consists of the cropped images of each building instance. Specifically, the large-scale remote sensing images are cropped by bounding boxes corresponding to each building instance and rescaled into the same size (denoted by $H_c \times W_c$). The coordinates of building vertices obtained from VGM are transformed accordingly for vertex embedding on the final feature map of the backbone, which are denoted by the red points on the cube above Vertex Embedding in Figure 1. Each vertex is assigned with the features extracted from the channel direction of the cube.

Propagation Model based on GGNN The polygon vertices obtained from the above step can be regarded as nodes of a graph, and every two neighboring vertices constitutes an edge of the graph. Inspired by previous work (Acuna et al. 2018), we adopt a gated graph neural network (Li et al. 2015) to learn the offset for each vertex, i.e., the relative displacement between a predicted valid vertex (obtained from VGM) and its nearest ground truth vertex. GGNN is capable of utilizing the extra information such as the feature of each node (vertex) and the relation between each node of the graph. The details of GGNN propagation model can be found in (Li et al. 2015). After the propagation process, we add two fully-connected layers which output a displacement value for each vertex. The prediction of displacement value is also formulated as a classification issue, and the whole polygon refinement network is trained using the cross entropy loss. In the coordinate transforming step, the output

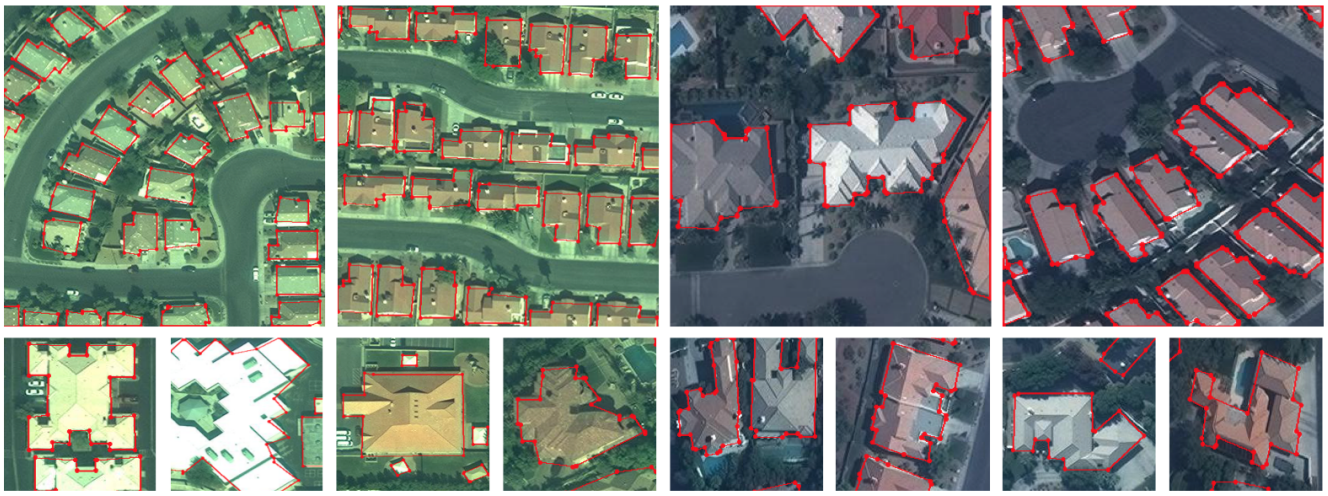

Figure 2: Examples of polygonal building segmentation results of our method. Our method produces vectorized outputs with accurate vertices and edges, even for buildings with complex shapes.

displacement classes of PRN are converted to the displacement coordinates, and added to the corresponding vertex coordinates of VGM to obtain the final building polygons. In this way, the GGNN-based PRN automatically moves the polygon vertices to more accurate locations.

Implementation Details

In our multi-task segmentation network, we use entirely the same Res-U-Net architecture (with ResNet101 as encoder) and training strategies as those used in (Li, Lafarge, and Marlet 2020) (for generating probability maps) for a fair comparison. The original remote sensing images in the training dataset are randomly cropped into 256×256 pixels. Accordingly, the test images are divided into 256×256 pixels with overlaps and the outputs are merged into large-scale images of the original size. The weights of three tasks $(\lambda_1, \lambda_2, \lambda_3)$ are all set as 1. For the vertex generation module, the corner probability threshold T_{cor} is set as 0.5 and the orientation difference threshold T_{ori} is set as 20°. For the polygon refinement network, each image cropped by the bounding box is resized to 224×224 pixels following (Ling et al. 2019). For the ResNet-based backbone, the size of the final feature map for vertex embedding is $112 \times 112 \times 256$. For the GGNN propagation model, the dimension sizes of the two fully-connected layers and the output layer are 256, 256 and 15×15, indicating that the relative moving range of each vertex is [-7,+7] pixels.

Results

Datasets

Following previous polygonal building segmentation studies (Zhao et al. 2018; Li, Wegner, and Lucchi 2019), we evaluate our proposed method using two popular building datasets: (1) CrowdAI mapping challenge dataset (CrowdAI) (Mohanty 2018). (2) SpaceNet building footprint dataset (SpaceNet) (Van Etten, Lindenbaum, and Bacastow 2018). Both datasets provide the vertex coordinates of each building polygon, which ensures an accurate evaluation of vertex prediction. CrowdAI is a large-scale building footprint dataset. The training dataset consists of over 280,000 images with around 2,400,000 annotated building footprints, and the test dataset contains over 60,000 images with around 515,000 buildings. The size of each image is 300×300 pixels. The SpaceNet building dataset contains satellite images and building footprints of several cities located in different continents. We use all the annotated building instances of Las Vegas in our experiment, which are accurately annotated in a relatively unified standard compared with other cities. The dataset of Las Vegas contains 3,851 images (in 650×650 pixels) and around 10,8000 building instances, which are randomly divided into 3,081/385/385 images as the training/validation/test datasets.

Evaluation Metrics

We use the official evaluation metrics of CrowdAI and SpaceNet challenges following (Zhao et al. 2018; Li, Wegner, and Lucchi 2019; Li, Lafarge, and Marlet 2020). Specifically, for the CrowdAI dataset, we use the average precision (AP) and average recall (AR) metrics under different IoU thresholds, which are calculated in the same procedure as (Li, Lafarge, and Marlet 2020; Girard et al. 2020). The metrics AP, AP_{50} and AP_{75} respectively denote the average precision under the IoU threshold of 0.50 to 0.95 (with a step of 0.05), 0.5, and 0.75 (similarly for AR, AR_{50} and AR_{75}). We also report the F1-score calculated from the above three cases. For the SpaceNet dataset, we report the F1-score under the IoU threshold of 0.5 following (Zhao et al. 2018; Demir et al. 2018). We additionally evaluate the vertex prediction results following (Liang et al. 2019; Nauata and Furukawa 2020). The precision, recall, and F1-score between the predicted and the annotated vertex set are calculated under the distance thresholds of 3 and 5 pixels.

Figure 3: Qualitative comparison with state-of-the-art. The top figures show the results of ASIP (Li, Lafarge, and Marlet 2020) and the bottom figures show the results of our method. The building polygons predicted by our method have more accurate vertices in terms of locations, quantities, and angles.

Method	AP	AP_{50}	AP_{75}	AR	AR_{50}	AR_{75}	$F1$	$F1_{50}$	$F1_{75}$
Mask-RCNN (He et al. 2017)	41.9	67.5	48.8	47.6	70.8	55.5	44.6	69.1	51.9
PANet (Liu et al. 2018)	50.7	73.9	62.6	54.4	74.5	65.2	52.5	74.2	63.9
PolyMapper (Li et al. 2019)	55.7	86.0	65.1	62.1	88.6	71.4	58.7	87.3	68.1
FrameField (Girard et al. 2020)	50.5	76.6	59.3	55.3	78.1	64.0	52.8	77.3	61.6
ASIP (Li et al. 2020)	65.8	87.6	73.4	**78.7**	**94.3**	**86.1**	71.7	90.8	79.2
Ours	**73.8**	**92.0**	**81.9**	72.6	90.5	80.7	**73.2**	**91.2**	**81.3**

Table 1. Quantitative comparison on CrowdAI dataset. Our method improves the F1-score of current state-of-the-art by 1.5%, 0.4%, and 2.1% under different IoU thresholds.

Method	U-Net	Mask-RCNN	Zhao et al.	Ours
F1-score	88.5	88.1	87.9	**89.4**

Table 2. Comparison with state-of-the-art on Vegas dataset.

Comparison with State-of-the-art

We compare our approach with several state-of-the-art methods on CrowdAI and Vegas datasets, including pixel-wise segmentation methods (producing raster results) and polygonal building segmentation methods (producing vectorized results). For CrowdAI, our method is compared with two pixel-wise segmentation methods (Mask-RCNN (He et al. 2017) and PANet (Liu et al. 2018)) and three polygonal segmentation methods (PolyMapper (Li, Wegner, and Lucchi 2019), FrameField (Girard et al. 2020), and ASIP (Li, Lafarge, and Marlet 2020)). The input probability maps and parameter setting of ASIP are the same as those used in (Li, Lafarge, and Marlet 2020) for a fair comparison. For the Vegas dataset, our method is compared with U-Net (the winning solution of SpaceNet Building Detection Challenge Round2) (Demir et al. 2018), Mask-RCNN (He et al. 2017), and Mask-RCNN with regularization (Zhao et al. 2018).

Table 1 and Table 2 list the quantitative comparison of different methods. Our method achieves the highest F1-score among four methods on the Vegas dataset. For the CrowdAI dataset, our method obtains the highest precision and F1-scores and the second highest recall among six methods. The superiority of our method is more remarkable when the IoU threshold is high, indicating the more precise polygon prediction results of our method. The decrease in recall compared with ASIP is partly due to the failure cases on small buildings. In Table 3, We further compare the vertex prediction scores of our method with ASIP (the best among five comparison methods on the CrowdAI dataset). Our algorithm significantly outperforms ASIP on vertex prediction scores, achieving an F1-score gain of over 6%. Figure 2 shows some examples of the qualitative results obtained by

	P_{3px}	R_{3px}	$F1_{3px}$	P_{5px}	R_{5px}	$F1_{5px}$
ASIP	51.13	**73.55**	60.32	69.25	**89.27**	78.00
Ours	**64.25**	69.90	**66.96**	**83.81**	85.85	**84.82**

Table 3. Comparison of our method and ASIP in terms of vertex prediction scores. Our algorithm achieves the F1-score gain of 6.64% and 6.82% compared with ASIP.

	P_{3px}	R_{3px}	$F1_{3px}$	P_{5px}	R_{5px}	$F1_{5px}$
Baseline	51.67	50.24	50.94	76.66	74.31	75.47
+ VGM	56.71	52.53	54.54	81.54	75.22	78.25
+ PRN	**69.50**	**61.70**	**65.37**	**86.54**	**76.56**	**81.24**

Table 4. Results of ablation study on Vegas dataset in terms of vertex prediction scores.

our approach. Figure 3 provides a qualitative comparison of the predictions of ASIP (Li, Lafarge, and Marlet 2020) and our method. Results demonstrate that our method is capable of producing building polygons with accurate vertices and edges (even short ones). The predicted building polygons of our method have more accurate vertex quantity and angles compared with ASIP.

Ablation Study and Failure Case Analysis

We conduct an ablation study to further evaluate the effect of each component of our approach. Table 4 lists the vertex prediction scores of Vegas dataset at different stages. The first row shows the evaluation results of the building segmentation task of the multi-task model (denoted by Baseline). As the outputs at this stage are in a raster format, we employ the Douglas-Peucker algorithm (Wu and Marquez 2003) (a popular contour simplification method) to convert the raster building segmentation results into polygon vertices. The second and the third rows show the evaluation results of the vertex generation module and the polygon refinement network (the final output). Figure 4 provides a qualitative comparison of the prediction results at different stages. We also demonstrate the visualized outputs of our multi-task segmentation network in the first row of Figure 4.

Results demonstrate that the vertex generation module produces much better vertex predictions compared with the building segmentation results that are simplified by Douglas-Peucker. Through effectively utilizing the corner and edge orientation predictions, the vertex generation module is capable of filtering out invalid vertices and remaining valid vertices with accurate quantity, and much more robust to poor building segmentation results compared with Douglas-Peucker. The polygon refinement network further improves the vertex prediction F1-scores by adjusting the vertices to more accurate locations. Figure 5 shows three typical examples of failure cases of our proposed method. Our method has difficulties in producing accurate polygons for buildings that are seriously sheltered by trees (left), buildings with multiple extremely short edges (middle), and high-rise buildings with serious parallax effect (right), which should be explored and solved in our future work.

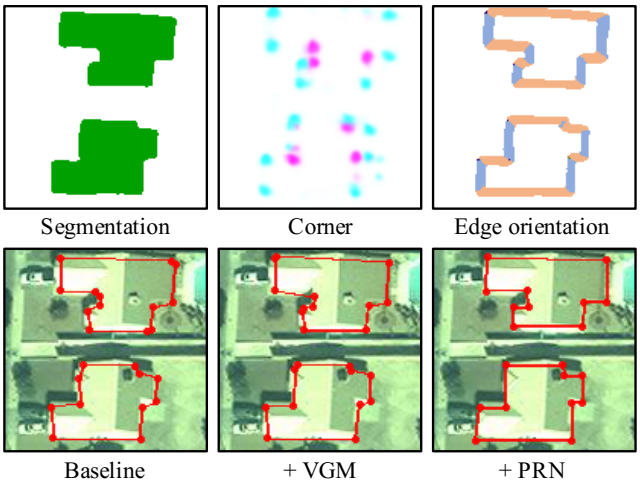

Figure 4: The visualized outputs of our multi-task network (the first row) and a qualitative comparison of the prediction results at different stages of our approach (the second row).

Figure 5: Three typical examples of failure cases of our proposed method. Our method fails to produce accurate polygons for buildings that are seriously sheltered by trees (left), buildings with multiple extremely short edges (middle), and high-rise buildings with serious parallax effect (right).

Conclusion

In this paper, we have presented a novel building segmentation approach that is capable of producing vector building polygons from remote sensing images. Qualitative and quantitative evaluations on two popular building segmentation datasets demonstrate that our proposed approach achieves significant improvements over state-of-the-art methods. The effect of each component of our approach is also verified in the ablation study. We believe that this paper motivates novel ideas for predicting vectorized object representations and provides effective solutions for practical applications in Geographic Information Systems. In our future work, we would like to explore novel methods for more complex application scenarios, such as producing the vectorized roof and footprint polygons for highrise and dense buildings.

References

Acuna, D.; Ling, H.; Kar, A.; and Fidler, S. 2018. Efficient interactive annotation of segmentation datasets with polygon-rnn++. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 859–868.

Bischke, B.; Helber, P.; Folz, J.; Borth, D.; and Dengel, A.

2019. Multi-task learning for segmentation of building footprints with deep neural networks. In *2019 IEEE International Conference on Image Processing (ICIP)*, 1480–1484. IEEE.

Castrejon, L.; Kundu, K.; Urtasun, R.; and Fidler, S. 2017. Annotating object instances with a polygon-rnn. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5230–5238.

Cheng, D.; Liao, R.; Fidler, S.; and Urtasun, R. 2019. Darnet: Deep active ray network for building segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 7431–7439.

Demir, I.; Koperski, K.; Lindenbaum, D.; Pang, G.; Huang, J.; Basu, S.; Hughes, F.; Tuia, D.; and Raskar, R. 2018. DeepGlobe 2018: A Challenge to Parse the Earth through Satellite Images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*.

Ding, J.; Xue, N.; Long, Y.; Xia, G.-S.; and Lu, Q. 2019. Learning roi transformer for oriented object detection in aerial images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2849–2858.

Dyken, C.; Dhlen, M.; and Sevaldrud, T. 2009. Simultaneous curve simplification. *Journal of Geographical Systems* 11(3): 273–289.

Girard, N.; Smirnov, D.; Solomon, J.; and Tarabalka, Y. 2020. Polygonal Building Segmentation by Frame Field Learning. *arXiv preprint arXiv:2004.14875* .

Gur, S.; Shaharabany, T.; and Wolf, L. 2020. End to End Trainable Active Contours via Differentiable Rendering. In *Proceedings of the International Conference on Learning Representations (ICLR) 2020*.

He, K.; Gkioxari, G.; Dollár, P.; and Girshick, R. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, 2961–2969.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Li, M.; Lafarge, F.; and Marlet, R. 2020. Approximating shapes in images with low-complexity polygons. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Li, W.; He, C.; Fang, J.; Zheng, J.; Fu, H.; and Yu, L. 2019. Semantic segmentation-based building footprint extraction using very high-resolution satellite images and multi-source GIS data. *Remote Sensing* 11(4): 403.

Li, Y.; Tarlow, D.; Brockschmidt, M.; and Zemel, R. 2015. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493* .

Li, Z.; Wegner, J. D.; and Lucchi, A. 2019. Topological map extraction from overhead images. In *Proceedings of the IEEE International Conference on Computer Vision*, 1715–1724.

Liang, J.; Homayounfar, N.; Ma, W.-C.; Wang, S.; and Urtasun, R. 2019. Convolutional recurrent network for road boundary extraction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 9512–9521.

Liang, J.; Homayounfar, N.; Ma, W.-C.; Xiong, Y.; Hu, R.; and Urtasun, R. 2020. Polytransform: Deep polygon transformer for instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9131–9140.

Ling, H.; Gao, J.; Kar, A.; Chen, W.; and Fidler, S. 2019. Fast interactive object annotation with curve-gcn. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 5257–5266.

Liu, S.; Qi, L.; Qin, H.; Shi, J.; and Jia, J. 2018. Path aggregation network for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8759–8768.

Mahmud, J.; Price, T.; Bapat, A.; and Frahm, J. M. 2020. Boundary-Aware 3D Building Reconstruction From a Single Overhead Image. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Marcos, D.; Tuia, D.; Kellenberger, B.; Zhang, L.; Bai, M.; Liao, R.; and Urtasun, R. 2018. Learning deep structured active contours end-to-end. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 8877–8885.

Mohanty, S. P. 2018. Crowdai dataset: the mapping challenge. https://www.crowdai.org/challenges/mapping-challenge. Accessed on 1 March 2021 .

Nauata, N.; and Furukawa, Y. 2020. Vectorizing World Buildings: Planar Graph Reconstruction by Primitive Detection and Relationship Inference. In *European Conference on Computer Vision*, 711–726. Springer.

Sun, X.; Christoudias, C. M.; and Fua, P. 2014. Free-shape polygonal object localization. In *European Conference on Computer Vision*, 317–332. Springer.

Van Etten, A.; Lindenbaum, D.; and Bacastow, T. M. 2018. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232* .

Wu, S. T.; and Marquez, M. R. G. 2003. A non-self-intersection Douglas-Peucker algorithm. In *Computer Graphics and Image Processing, 2003. SIBGRAPI 2003. XVI Brazilian Symposium on*.

Yuan, Y.; Xie, J.; Chen, X.; and Wang, J. 2020. Segfix: Model-agnostic boundary refinement for segmentation. In *European Conference on Computer Vision*, 489–506. Springer.

Zhao, K.; Kang, J.; Jung, J.; and Sohn, G. 2018. Building Extraction From Satellite Images Using Mask R-CNN With Building Boundary Regularization. In *CVPR Workshops*, 247–251.

Zorzi, S.; Bittner, K.; and Fraundorfer, F. 2020. Machine-learned Regularization and Polygonization of Building Segmentation Masks. In *2020 IEEE International Conference of Pattern Recognition*.