OEC-RNN: Object-Oriented Delineation of Rooftops With Edges and Corners Using the Recurrent Neural Network From the Aerial Images

Wei Huang, Hong Tang, *Member, IEEE*, and Penglei Xu

***Abstract*— It is an important task to automatically and accurately map rooftops from very high resolution remote sensing images since buildings are very closely related to human activity. Two typical technologies are often utilized to accomplish the task, i.e., semantic segmentation and instance segmentation. The semantic segmentation is to independently allocate a label (e.g., "building" or not) to each pixel, resulting in blob-like segments. On the contrary, one might model the boundary of a rooftop as a polygon to improve the shape of the rooftop by encouraging vertices of polygon to adhere to the rooftop's boundary. Following this line of work, we present a multitask learning approach to predict rooftop corners in a sequent way using the attention learned from where the boundaries are in a given image region. The approach simulates the process of manual delineation of rooftops' outline in a given image, which can produce accurate boundaries of rooftops with sharp corners and straight lines between them. Specifically, the proposed method consists of three components, i.e., object detection, pixel-by-pixel classification of both edges and corners, and delineation of rooftops in a sequent manner using a convolutional recurrent neural network (RNN). It is called as object-oriented, edges and corners (OEC)-RNN in this article. Three image datasets of buildings are employed to validate the performance of the OEC-RNN, which are compared with state-of-the-art methods for instance segmentation. The experimental results show that the OEC-RNN achieves the best performance in terms of overlay, boundary adherence, and vertex location between ground-truth and predicted polygons.**

***Index Terms*— Building extraction, convolutional neural network (CNN), recurrent neural network (RNN), rooftop delineation.**

I. Introduction

AUTOMATIC and accurate mapping of buildings from very high resolution (VHR) remote sensing imagery and aerial imagery is important for a wide range of applications, for example, cartography, urban management, quick response to natural disasters, and so on. Along with the successful application of convolutional neural network (CNN) on image classification [1], deep learning methods have been used to extract buildings from VHR images [2]–[8], which can be technically categorized into two groups, i.e., semantic segmentation [9] and instance segmentation [10].

As for the semantic segmentation of buildings, a class label (e.g., "building" or not) would be allocated onto each pixel in a given image. These kinds of dense classification of images could achieve good global performance statistics such as overall accuracy and building area coverage estimation [11]. However, the classification results produced by these methods are generally not well adherent to the regular geometric shapes of buildings, since the inference of the pixels' label is independent of each other [12]. Semantic segmentation results look like blob-like segments [13] since they do not reason about the geometry of its predictions. Some post-processing strategies, e.g., building boundary regularization [14], [15] are often used to improve the degree of boundary adherence, where boundary adherence is a general name of the commonly used methods for evaluating the segmentation boundary in image segmentation and over-segmentation tasks [16]. Boundary recall and under-segmentation error are standard measures for boundary adherence [17], [18].

Unlike semantic segmentation, instance segmentation aims to allocate a unique label for per-object instance, e.g., individual buildings [18]. A natural approach is to use semantic segmentation results as a part of instance segmentation. For instance, one might discover connected components from the semantic segmentation as object masks, or semantic segmentation of a specific image region discovered by an object detection network, e.g., Mask region-CNN (R-CNN) [19]. However, this kind of method still originates from pixel-wise classifications and is not apt for integrating output with shape priors directly.

Another approach to instance segmentation is to directly model the boundary of an object instance as an active contour [12], [13], [20] or a polygon [21], [22]. Active contour models (ACMs) have been proved to be an extremely popular approach to instance segmentation, which was introduced by Kass *et al.* [23] under the name "snake." An initial contour would be encouraged to move toward the boundaries of an

Manuscript received November 12, 2020; revised March 28, 2021; accepted April 24, 2021. Date of publication May 24, 2021; date of current version January 5, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 41971280 and in part by the National Key Research and Development Program of China under Grant 2017YFB0504104. *(Corresponding author: Hong Tang.)*

The authors are with the State Key Laboratory of Remote Sensing Science, Jointly Sponsored by Beijing Normal University and Institute of Remote Sensing and Digital Earth of Chinese Academy of Sciences, Beijing 100875, China, and also with the Beijing Key Laboratory of Environmental Remote Sensing and Digital Cities, Faculty of Geographical Science, Beijing Normal University, Beijing 100875, China (e-mail: tanghong@bnu.edu.cn).

Digital Object Identifier 10.1109/TGRS.2021.3076098

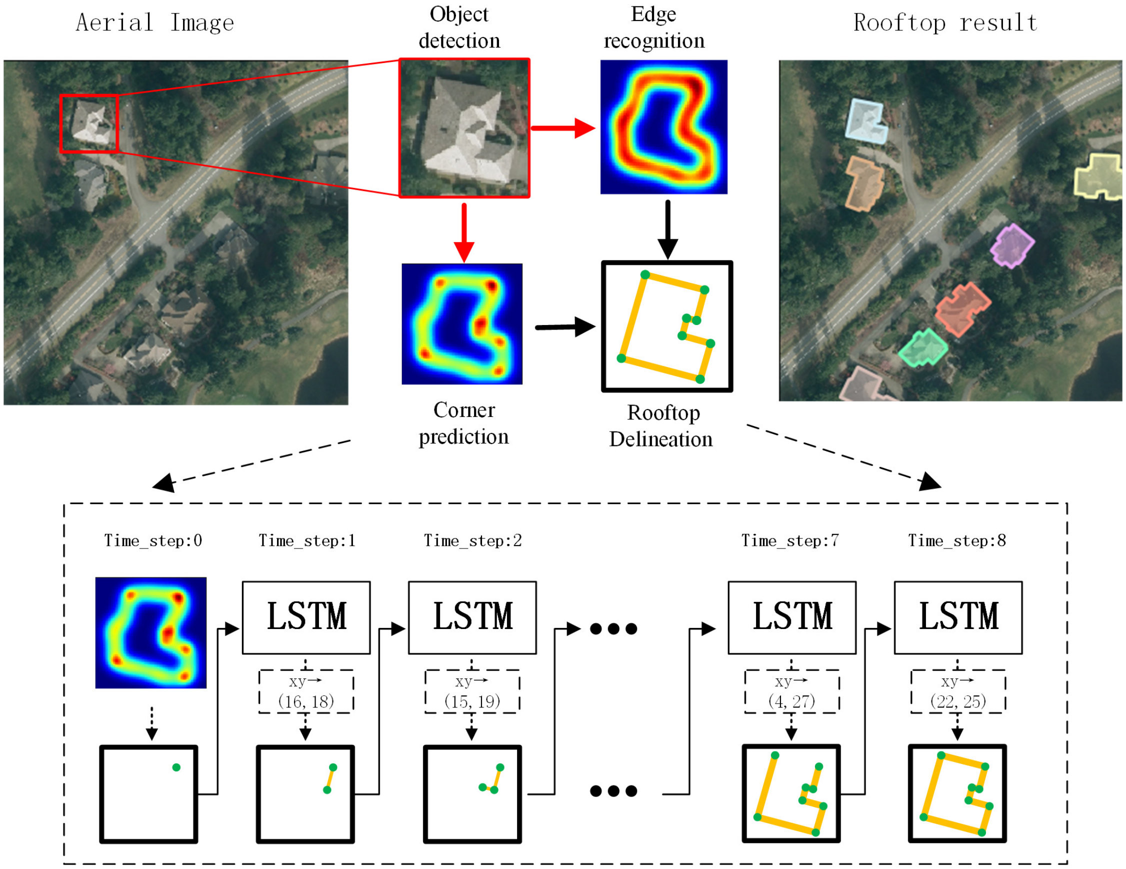

Fig. 1. Components of the OEC-RNN for rooftop delineation of individual buildings.

object by minimizing an energy function, in which some geometric properties can be encoded as priors of the contour, for instance, curvature and area of the contour. Previously, a deep structured active contour (DSAC) [12] model was employed to combine the power of deep CNNs with the classic polygon-based ACM. A structured prediction is integrated with the ACM by minimizing the intersection over union (IoU) between the predicted polygon and ground-truth outline of a building. The DSAC is furthermore extended to the situation that the contour is represented in a polar coordination, i.e., the deep active ray network (DARNet) [13]. This contour then evolves to minimize its energy via gradient descent, and its final position defines the predicted instance segmentation. Gur *et al.* [20] present a simpler framework to create an active contour from its vertices with a fully differentiable rendering, i.e., active contours via differentiable rendering network (ACDRNet). Recently, inspired by the snake model and combined with deep learning, a Deep snake [24] method was proposed. This method uses circular convolution modeling and infers the offset of the initial contour vertex to get more accurate contour results. To speedup manual annotation, a recursive neural network is used to generate polygon vertices in a sequential manner, where it allows human annotators to correct incorrect vertices as needed to produce a precise polygon as possible as annotators can do [21]. Furthermore, it was extended to make use of reinforcement learning [22] and graph convolution networks (GCNs) [25]. To enhance the efficiency of human correction in sequent predictions, a GCN was proposed to simultaneously predict the vertices of a polygon or spline outlining the object.

Although contour or polygon-based methods generally outperform semantic segmentation in terms of instance geometric features, it is still hard to produce a boundary of a rooftop with sharp corners and straight lines between them. The main reason is that only part of the vertices on the boundary of a rooftop is occasionally chosen by prior method or sampled during learning, which is used to represent an instance of individual building. This motivate us to explicitly learn how to predict corners, instead of vertices, of rooftops and delineate the sharp outlines of rooftops by using the most possible corners in a sequent manner under some attention, e.g., the posterior probability of predicted edges on the boundary of a rooftop using the richer convolutional features (RCF) for edge recognition [26].

Specifically, a novel approach is present to accurately delineate the boundaries of individual buildings from aerial images in this article. As shown in Fig. 1, the proposed method consists of three components, i.e., object detection, pixel-by-pixel classification of both edges and corners, and delineation of rooftops in a sequent manner using a convolutional recurrent neural network (RNN). It is called object-oriented, edges and corners (OEC)-RNN, in which a convolutional long short-term memory (ConvLSTM) is used as the recurrent unit [27] and each time step corresponds to the next predicted corner. The maximum number of corners in this article is limited to 71.

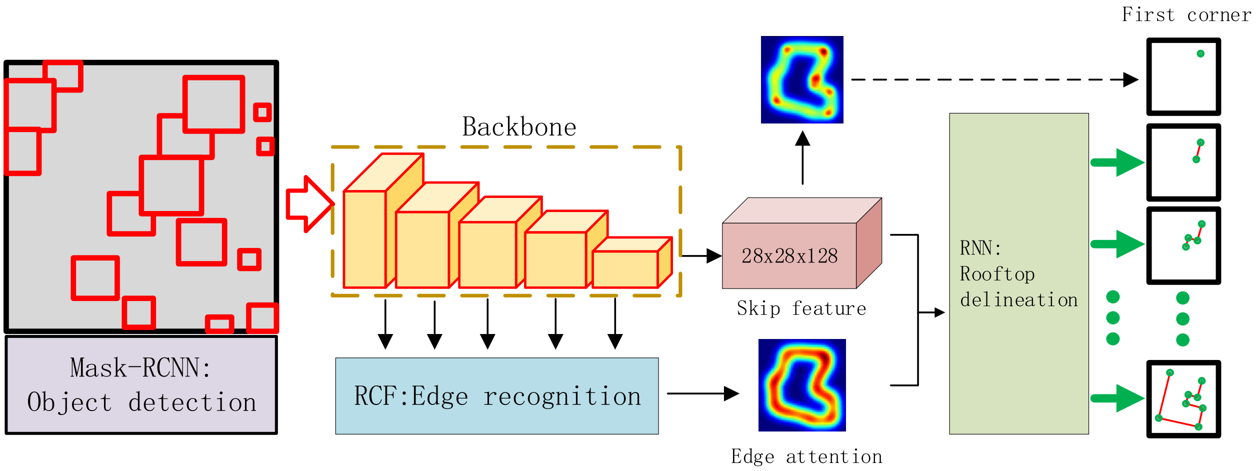

Fig. 2. Framework of neural network in the OEC-RNN.

The rest of this article is organized as follows. The proposed method is described in Section II. Experimental setting and results are given in Section III. In Section IV, we discuss the impact of object detection on the performance in term of quantitative evaluation, and the limitation of instance segmentation using a biased assumption. Some conclusions are drawn in Section V.

II. Method

The framework of neural network in the OEC-RNN is shown in Fig. 2.

1) The Mask R-CNN is used to detect where the buildings are located in an image.
2) Both the RCF and feature extraction with a shared CNN backbone are employed to predict the degree of each pixel belonging to the rooftop edge. And the first corner is inferred from a skip feature.
3) Based on the skip feature coupled with predicted edge attention and first corner, a OEC-RNN is utilized to delineate the boundary of a rooftop in a sequent manner.

In Section II-A–II-C, the specific network structures of its components are described in detail, i.e., object detection, recognition of edges and corners, and delineation of rooftops.

A. Object Detection Using the Mask R-CNN

The Mask R-CNN is a conceptually intuitive and high-performance framework for instance segmentation, which also shows high precision results on bounding box detection [19]. Much of this approach has evolved from the powerful object detection frameworks such as Faster R-CNN [28]. As shown in Fig. 3, the intermediate convolutional layer of CNN backbone is used as the input of the region proposal network (RPN) and then used as the image feature maps of region proposals through the region of interest (RoI) align layer to generate a fixed-size feature map to execute box regression, classification, and mask prediction.

In this article, the class of object is buildings or not. The boxes of buildings would be enlarged by 10% of the short side length of detected bounding boxes to ensure that the buildings are completely located in the boxes, and then cropped for furthermore processing. The results of the mask brunch could be used for qualitative comparisons in Section III-D.

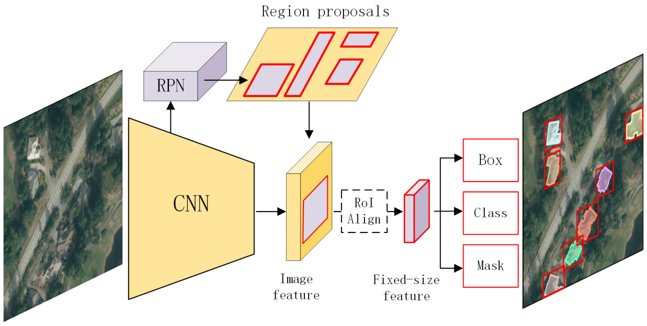

Fig. 3. Network of object detection used in this article.

B. Pixel-by-Pixel Prediction of Edges and Corners

After the object detection using the Mask R-CNN, each image crop contains only one building. In order to pass it to the next network for processing, the image crop needs to be resized to a fixed size (i.e., 224 × 224). In this component, Resnet-101 [29] is used as the backbone for feature extraction of image crop, while both the last max-pooling and fully connected layer in the original network structure were dropped out. Based on extracted features of the Resnet-101 backbone from an image crop including a building, the RCF and residual network with skip connections are utilized to predict possible pixels of edges and corners, respectively. The probability map of edge is used as attention to guide the sequent prediction of corners in the OEC-RNN.

The RCF exhibits excellent performance in terms of edge detection [26], in which image features from multiple stages are fused in a unified framework. As shown in Fig. 4, the backbone is divided into five stages in the RCF. Each stage is followed by a kernel size of 1 × 1 and 21-channel convolutional filters for dimensionality reduction. Then the 21-D output of the stage is deconvoluted to restore the input

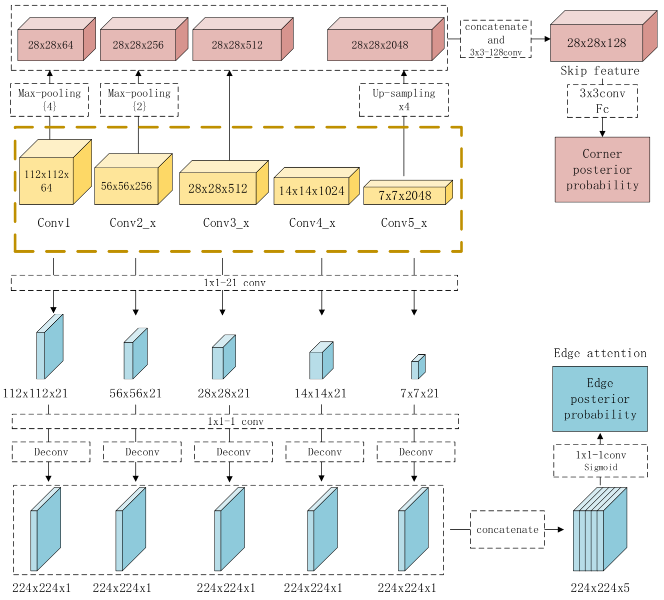

Fig. 4. Structure of the edge recognition component.

size as stage feature map with size of 224 × 224, which is used to calculate the cross-entropy loss. The stage feature map of each stage is concatenated and convoluted by a kernel size of 1 × 1 and one channel convolutional filter in the final stage. The sigmoid is used to estimate the posterior probability of each pixel. The output of the RCF is a probability map which is called "edge attention" in the following.

Given the Resnet-101 backbone, both the Conv1 and Conv2_x are connected to the feature maps with a size of 28 × 28 by max-pooling of size 4 × 4 and 2 × 2, respectively. Both the Conv3_x and Conv5_x are directly copied and up-sampled into the feature maps, respectively. By concatenating the above four feature maps and convoluting the concatenated features with a kernel size of 3 × 3 and rectified linear unit (ReLU) nonlinearity, 128 feature maps with a size of 28 × 28 are outputted as high-level features for corner predictions. We refer to the final feature map as the skip feature [21]. Specifically, the skip feature is followed by a 3 × 3 kernel with a 16-channel convolutional layer and fully connected layer to predict the corners. The pixel with the highest posterior would be chosen as the first corner of a rooftop during delineation of rooftop using a ConvLSTM.

C. Delineation of Rooftops Using a ConvLSTM

The RNN component in the OEC-RNN follows the part of Polygon-RNN [21], [22] to decode the skip feature and model the boundary of a rooftop. The major distinction is that the sharp corners instead of vertices on the boundary of a rooftop would be predicted along with the edge attention inferred by the RCF. ConvLSTM component comes from the Polygon-RNN architecture. We add the network structure of edge recognition, so that the building corner delineation process focuses on the area near edges.

Fig. 5 shows the structure of RNN components in the OEC-RNN, where edge attention has been downsampled to a size of 28 × 28 by combining skip feature and edge attention to generate attention-weighted feature by the Hadamard product. Given one-hot encodings of two previous vertices y_{t-1}, y_{t-2} and the first vertex y_0, the attention-weighted feature is fed to the RNN at time step t.

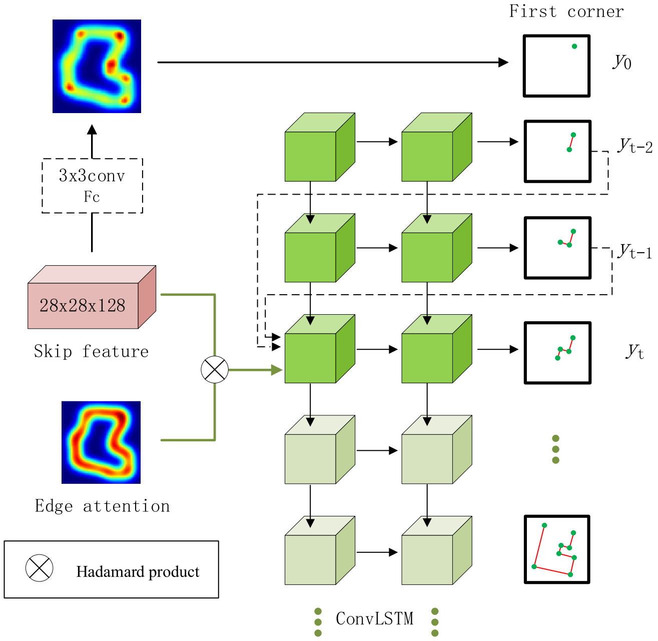

Fig. 5. Structure of the RNN component.

The detailed network structure of ConvLSTM is shown in Fig. 6. The main structure of the RNN is made up of a double-layer ConvLSTM with a 3 × 3 kernel with 64 and 16 channels, respectively, which can preserve spatial information and reduce the number of parameters to learn. The output of the ConvLSTM is the location of corner points at each time step. Rooftop polygons can be naturally drawn by the corner points in order. The output of the ConvLSTM is a point in the 28 × 28 plane, where each point is considered as a classified category (i.e., 784 categories in total), and the polygon loss of the OEC-RNN model is defined as the cross-entropy loss of these 784 categories. Consequently, the problem of locating the corner points is transformed into a classification task.

The total loss function is given in (1). Since the loss of both edges and points is not in the same magnitude as the loss of polygons, the loss value of edges and points is multiplied by a weight to the same magnitude as the loss of polygons

$$\text{Loss} = \text{polygon\_loss} + w_1 * \text{edge\_loss} + w_2 * \text{corner\_loss} \tag{1}$$

where w_1 and w_2 are set to 200 in this article.

III. Experimental Results

In this section, we first specify the experimental setting, for instance, image datasets of buildings, strategy of model training, and hyperparameters. Then six metrics are described for quantitative evaluation. Experimental results under two kinds of scenarios are given in Section III-D and III-E, respectively. The first scenario is qualitative compassion and quantitative evaluation between the proposed method with state-of-the-art

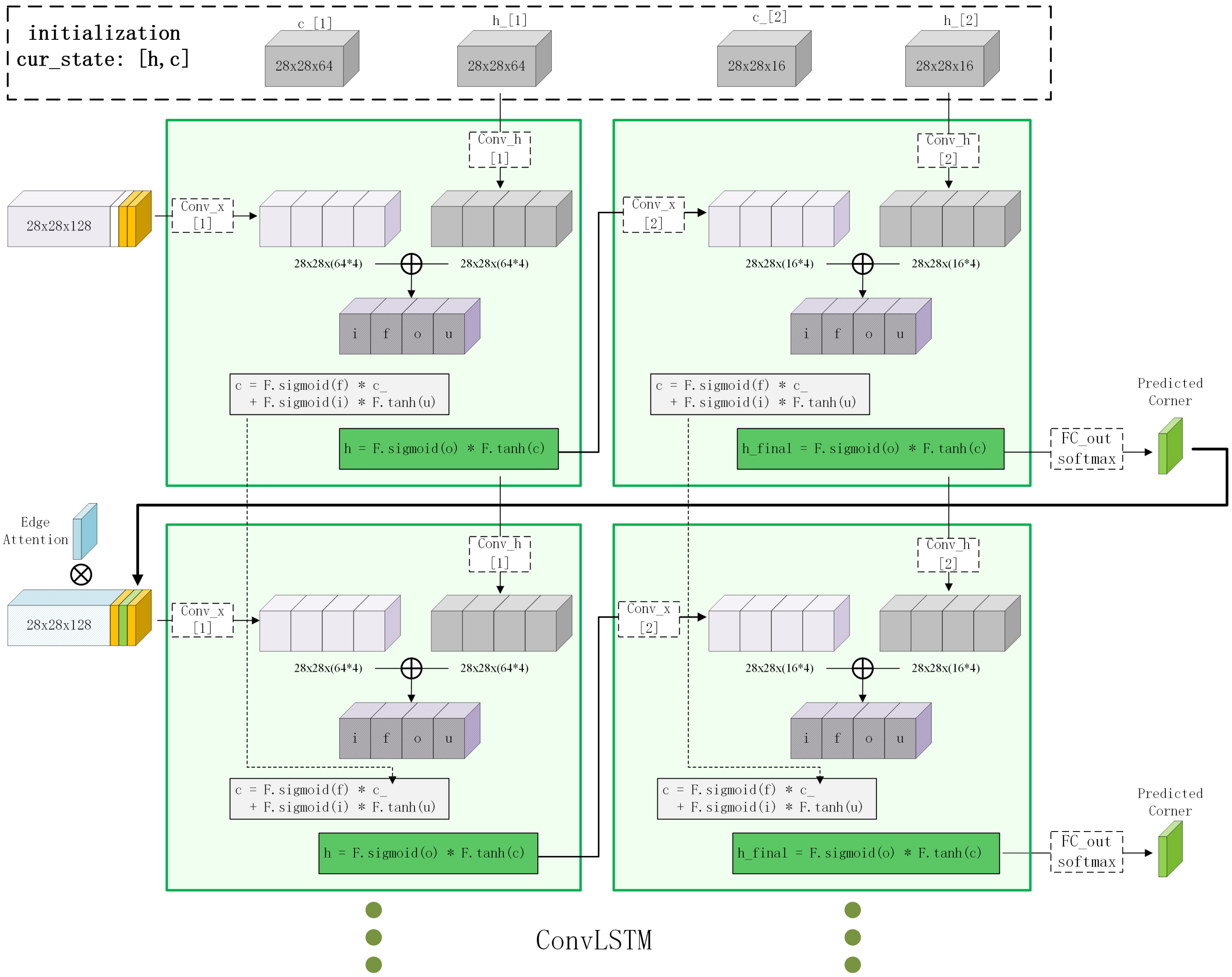

Fig. 6. Network structure of ConvLSTM: $c\_$ [1], $c\_$ [2] and $h\_$ [1], $h\_$ [2] are current state and hidden state of the first and second layers, respectively. Conv_x and Conv_h are the convolution kernel of the input features and the convolution kernel of the hidden state at the previous time step, respectively. C and h are outputs of current state and hidden state, respectively. c will be passed to the next time step, while h will be passed to both the next time step and the second layer at the same time step. I, f, o, and u represent the input gate, forget gate, output gate, and update information in the LSTM structure, respectively.

methods, i.e., Mask R-CNN [19], ACDRNet [20], Polygon-RNN [21], Deep snake [24], and Curve-GCN [25], under the assumption that the image of individual building has been correctly detected and cropped from original aerial images using the given expanded ground-truth bounding box. The second one is making ablation experiments in order to analyze the impact of different components in the OEC-RNN on the delineation of rooftops.

A. Datasets

1) ISPRS-Vaihingen: Vaihingen data [30] is a standard dataset for evaluating object extraction in “semantic labeling contest” of the International Society for Photogrammetry and Remote Sensing (ISPRS) Commission II Working Group (WG) III/4. This dataset consists of 33 pixel-level annotated images with an average size of 1920 × 2650 from Vaihingen in Germany. It is worth mentioning that each image in this dataset is an 8-bit true orthophoto with a resolution of 0.09 m, which includes near infrared response (NIR), and red and green bands.

2) INRIA-Austin: The INRIA aerial image labeling dataset [31] is mainly used for mapping urban buildings in aerial images for semantic segmentation tasks. In this dataset, the region of Austin with high data quality was chosen as the experimental data. The training set contains 36 orthorectified color images of size 5000 × 5000 with a spatial resolution of 0.3 m.

3) Massachusetts Buildings Dataset: There were 137 aerial images with corresponding labels of size 1500 × 1500, and the spatial resolution was 1 m. Compared with the two above-mentioned datasets, the Massachusetts buildings dataset [32] has a lower spatial resolution and covers a wider surface. Therefore, the building objects in the data are relatively smaller and more difficult to be detected.

B. Model Training

For each dataset, we selected 80% of the images as training and validation samples, and the remaining 20% as test samples. For training, we trained the Mask R-CNN model for object detection. Then a ResNet-101 is used as a shared

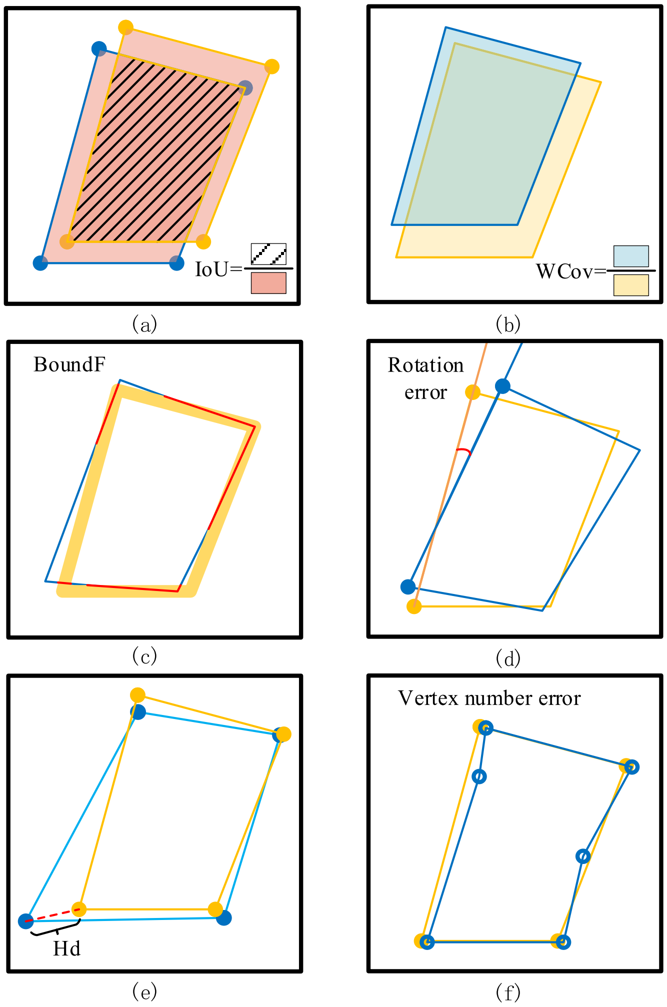

Fig. 7. Evaluation metrics. (a) IoU. (b) WCov. (c) BoundF. (d) RE. (e) Hd. (f) VNE.

backbone between the RCF and corner detection head. Finally, a double-layer ConvLSTM is trained to predict sequent corners for each individual rooftop. We use Adam optimizer with a batch size of 8 and an initial learning rate of 1e − 4. We decay the learning rate by a factor of 0.1 for every ten epochs. Taking INRIA-Austin as an example, the training phase of the model spends about 16 h on Nvidia Tesla P40 GPU and the learned model run at 46 s per image (size of 5000 × 5000) during the inference phase.

C. Evaluation Metrics

As shown in Fig. 7, six metrics are utilized to quantitatively evaluate the performance of models from three different viewpoints, i.e., object, boundary, and vertex between the ground truth and the prediction.

We defined the area of the polygon delineated by the connected lines of the predicted building corner points as A_{p} and the ground truth A_{g}. Given both predicted and ground-truth rooftops, the IoU [33] is the ratio of their intersection to their union, which is as follows:

$$\mathrm{IoU} = \frac{A_{\mathrm{g}} \cap A_{\mathrm{p}}}{A_{\mathrm{g}} \cup A_{\mathrm{p}}}. \tag{2}$$

Weighted coverage (WCov) [34] is the ratio of the area of two polygons, where the larger area is used as the denominator. The WCov is computed according to

$$\mathrm{WCov} = \begin{cases} A_{\mathrm{g}}/A_{\mathrm{p}}, & A_{\mathrm{g}} < A_{\mathrm{p}} \\ A_{\mathrm{p}}/A_{\mathrm{g}}, & A_{\mathrm{g}} \geq A_{\mathrm{p}}. \end{cases} \tag{3}$$

Boundary F-score (BoundF) [35] is the averaged F1-score on the threshold value of 1 to k pixels ($k = 5$, in this article) around the ground-truth boundary of the rooftop according to the following equation:

$$\mathrm{BoundF} = \frac{1}{k} \sum_{T=1}^{k} \frac{2 P_T R_T}{P_T + R_T} \tag{4}$$

where T is the threshold value around the ground-truth boundary and $T = \{1, 2, \ldots, k\}$, k is the maximum threshold. P_T and R_T are the precision and recall under threshold T. P and R are given by

$$P = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{5}$$

$$R = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{6}$$

where the true positive (TP) is the number of pixels correctly predicted as the boundary. False positive (FP) and false negative (FN) are the number of pixels with incorrect boundary and nonboundary, respectively.

Rotation error (RE) is defined as the deflection angle between the orientations of both ground-truth and predicted polygon, where the magnitude of the angle is expressed in degrees. RE is given by

$$\mathrm{RE} = |O_{\mathrm{p}} - O_{\mathrm{g}}| \tag{7}$$

where O_{g} and O_{p} are orientations of ground-truth and predicted polygon. Orientation is defined as the angle between the x-axis and the major axis of the ellipse that has the same second moments as the polygon, ranging from $-\pi/2$ to $\pi/2$ counterclockwise.

The Hausdorff distance (Hd) is used to measure the maximal–minimal distance between two sets of vertex points from ground-truth and predicted polygons, respectively [36]. Given the two sets of vertex points from ground-truth and predicted polygons, e.g., $A = \{a1, a2, \ldots\}$, $B = \{b1, b2, \ldots\}$, Hd is given by

$$\mathrm{Hd}(A, B) = \max[h(A, B), h(B, A)] \tag{8}$$

where

$$h(A, B) = \max_{a \in A} \left[\min_{b \in B} \|a - b\| \right] \tag{9}$$

$$h(B, A) = \max_{b \in B} \left[\min_{a \in A} \|b - a\| \right]. \tag{10}$$

The two distances $h(A, B)$ and $h(B, A)$ are termed as forward and backward Hds between A and B.

Vertex number error (VNE) is to measure the difference between the number of predicted vertices N_{P} and the number of ground-truth vertices N_{g} on a rooftop polygon. VNE is defined by the following equation:

$$\mathrm{VNE} = |N_{\mathrm{p}} - N_{\mathrm{g}}|. \tag{11}$$

D. Comparison With State-of-the-Art Methods

Five state-of-the-art methods are selected to be compared with the proposed method in terms of both quantitative and qualitative performances.

ACDRNet is the method for representing polygons based on active contours. Under the assumption that each image crop contains only one building, the method is initialized by taking the center of the input image crop as the center of the initial circle and 16 pixels as the diameter. The number of initialized vertices is selected to be 32 in this article, at which the performance of the model can reach saturation quickly, as described in [20] and [21].

Deep snake [24] is an effective improvement of the snake method by introducing a circular convolutional structure to process the input contour vertices and obtaining the offset that each vertex needs to be adjusted to enclose the instances as accurately as possible based on the learned features, and then iterating through to obtain more accurate contour results. The experiments show that this method can carry out instance segmentation more rapidly and accurately.

Curve-GCN is to predict polygons using graph CNNs to estimate the displacement at each vertex. The initialization of this method also takes the center of the input image crop as the center of the circle, and takes 70% of the image height as the diameter. The number of initialized control points will be uniformly sampled along the edge of the initial circle. In [25], a better performance can be achieved when 40 points are used as initial points.

Mask R-CNN is an instance segmentation framework for object detection, classification, and semantic segmentation. As the semantic segmentation results of the mask branch in the Mask R-CNN presented blob-like shapes, we used the method in [15] for regularizing comparison and quantitative evaluation.

Polygon-RNN is a way of annotating the object contour using vertices on the boundary both automatically and interactively [21]. The OEC-RNN approach in this article evolved from the method and its extended version polygon RNN++ [22], and it was natural to use it as a baseline for performance comparisons.

1) Qualitative Comparison: Two images of buildings from each of the three datasets are selected for qualitative comparison in terms of delineation of rooftop boundary. It can be seen from Fig. 8 that the OEC-RNN consistently delineates the rooftops with sharpened corners and straight lines between them. The OEC-RNN has a more accurate grasp of the building corner points and accurately depicts the roof polygon without severe corner deviations. Overall, the OEC-RNN exhibits good performance with different resolution images as input.

As shown in the third row in Fig. 8, the bottom-left and top-left corners are missing by the ACDRNet. As can be seen from the fifth and sixth rows, multiple corner points are missing and the corners are mispositioned on the edge. This leads to the result of the ACDRNet severely changing the original geometry of the building rooftop. Most of the sharp corners are missing, and as for the Mask R-CNN, it has been regularized based on its results of semantic segmentation. In the first and second rows, the Polygon-RNN recognized extra isolated corner points out of the edge, which makes the rooftop shape protrude sharply in local areas. The result of the fifth row also has a more obvious corner recognition deviation. Curve-GCN has poor recognition results for some concave building corners, and the description of building polygons is more inclined to convex polygons. The result of Deep snake is similar to Curve-GCN wherein a fixed number of control points are used to approximate the outline of the building, which makes the straight building edges rough. In general, different methods can achieve better visual consistency on objects with relatively simple geometric shapes such as the sixth row, while in the case of very complex boundaries, it is still challenging to make a fine drawing of the building contour like the fourth row in Fig. 8.

TABLE I
QUANTITATIVE EVALUATION RESULTS

		IoU	WCov	BoundF	RE	Hd	VNE
ISPRS	ACDRNet	0.8709	0.8565	0.7652	0.8981	5.39	----
	Mask R-CNN	0.8791	0.8610	0.7706	0.8159	4.91	8.75
	Polygon-RNN	0.8837	0.8628	0.7868	0.8214	5.09	5.37
	Deep Snake	0.8847	0.8706	0.7796	0.7854	5.29	----
	Curve-GCN	0.8864	0.8603	0.7811	0.7668	5.12	----
	Our	**0.9002**	**0.8784**	**0.8075**	0.7459	4.28	**5.05**
INRIA	ACDRNet	0.7881	0.7668	0.7163	0.6301	3.26	----
	Mask R-CNN	0.7812	0.7580	0.7067	0.7397	4.37	7.61
	Polygon-RNN	0.8007	0.7719	0.7225	0.6283	3.48	3.41
	Deep Snake	0.7902	0.7598	0.7066	0.6243	3.74	----
	Curve-GCN	0.7930	0.7618	0.7189	0.6140	3.37	----
	Our	**0.8144**	**0.7776**	**0.7293**	0.5653	2.72	**3.12**
Massachusetts	ACDRNet	0.7404	0.7075	0.6898	0.7085	2.74	----
	Mask R-CNN	0.7403	0.7098	0.6848	0.7689	3.52	6.83
	Polygon-RNN	0.7473	0.7364	0.6872	0.7180	3.11	3.75
	Deep Snake	0.7502	0.7398	0.7109	0.7742	3.04	----
	Curve-GCN	0.7569	0.7451	0.7143	0.8007	2.77	----
	Our	**0.7755**	**0.7544**	**0.7212**	0.6838	2.36	**3.21**

2) Quantitative Evaluation: Table I lists the quantitative evaluation on the three datasets using the six metrics in Section III-C. The OEC-RNN obtained higher precision than other methods in terms of IoU, WCov, and BoundF. Less mistakes are made by the OEC-RNN than the other methods in terms of RE, Hd, and VNE.

The two metrics (IoU and WCov) show from the plane of polygons that the better the conformity between predicted polygons and ground truth is, the higher the value of the metrics would be. Similarly, BoundF represents the consistency of the boundary of the polygons. The OEC-RNN obtained the highest values in three datasets, which was significantly improved compared with other methods.

RE shows the difference in the deflection angle of the polygon. Within the same dataset, the value magnitude of

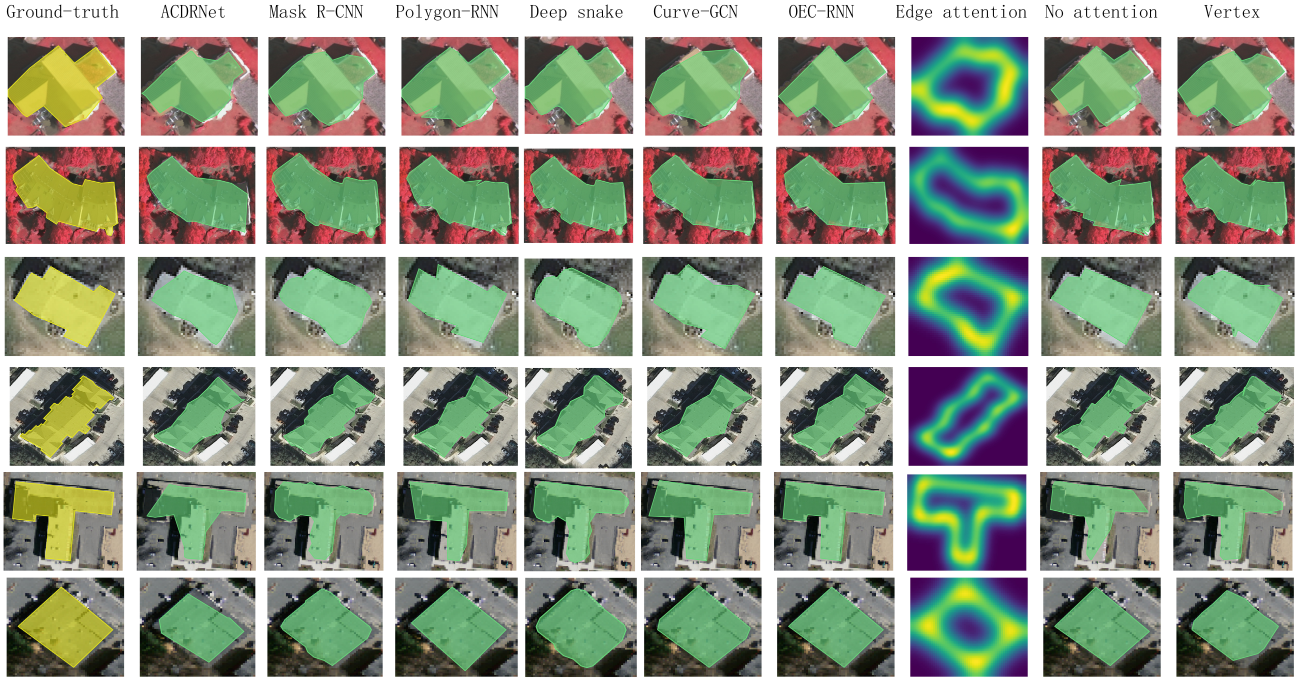

Fig. 8. Qualitative comparison. These building instances are selected from three datasets: the first and second rows are from ISPRS, the third and fourth rows are from INRIA, and the fifth and sixth rows are from Massachusetts building datasets. (Left to Right) the first column shows the building ground truth. The second to the seventh column are results of ACDRNet, Mask R-CNN, polygon-RNN, deep snake, curve-GCN, and OEC-RNN, respectively. The eighth column is the *a posteriori* probability map of building edges in the OEC-RNN. The ninth column is the results without using edge attention in the OEC-RNN. The right-most column is the results obtained by training using both some vertices on edges and corner points in the OEC-RNN.

the metric has little difference. The OEC-RNN achieved the minimum deflection angle on all three datasets.

Hd measures the spatial deviation of two sets of points that make up a polygon, with particular focus on the farthest point. All the methods used in this article are able to control Hd below a distance of ten pixels, and OEC-RNN obtains the lowest metric values of 4.28, 2.72, and 2.36 in the three datasets, respectively, thus demonstrating superior performance of the method in controlling the predicted singularity.

For VNE, since ACDRNet, Curve-GCN, and Deep snake require initialization of a fixed number of control points, the metric for the two methods are not considered for comparison in Table I. The polygon results of the Mask R-CNN are obtained by semantic segmentation with regularization, and the number of vertices that make up the polygon are significantly more than the Polygon-RNN and OEC-RNN that directly predict the building corner points. Although OEC-RNN and Polygon-RNN do not differ significantly in the VNE metric, the OEC-RNN shows a smaller difference in the number of vertices.

E. Ablation Experiments

In this section, two experiments are utilized to reveal the impact of two components on the performance of the OEC-RNN, i.e., edge recognition and corner prediction.

1) Edge Attention or Not: As shown in Fig. 4, the posterior probability of edge pixels learned using the RCF is used as attention to guide the delineation of rooftops in the OEC-RNN. The eight column of Fig. 8 shows the posterior probability map of edge of each individual building. The ninth column shows the results predicted using the OEC-RNN without using the edge posterior probability as attention. It can be seen from the subfigures in the seventh column that the corners predicted by the OEC-RNN is located among the "buffer zone" with higher posterior probability. Without edge attention, the method might miss some corners of rooftops, as shown in the first and fifth buildings in the ninth column. Some unexpected points might be produced by the method occasionally. For example, there exist some acute angles in the second and fourth buildings, which obviously deviates the real boundary of buildings.

TABLE II
QUANTITATIVE EVALUATION FOR ABLATION EXPERIMENT

		IoU	WCov	BoundF	RE	Hd	VNE
ISPRS	No Att	0.8895	0.8613	0.7714	0.7842	4.96	7.24
	Vertex	0.8712	0.8595	0.7823	0.8117	5.23	9.65
INRIA	No Att	0.8056	0.7798	0.7049	0.6132	4.06	4.95
	Vertex	0.7956	0.7863	0.7114	0.6355	5.91	7.98
Massac husetts	No Att	0.7523	0.7487	0.6754	0.6972	3.95	3.88
	Vertex	0.7497	0.7399	0.6962	0.7218	4.06	6.24

The quantitative evaluation results of the ablation experiment are summarized in Table II. Compared with the OEC-RNN, the OEC-RNN without attention behaves in a rather worse way in terms of all of the six metrics. Specifically, BoundF becomes a significant lower value when attention is removed from the OEC-RNN.

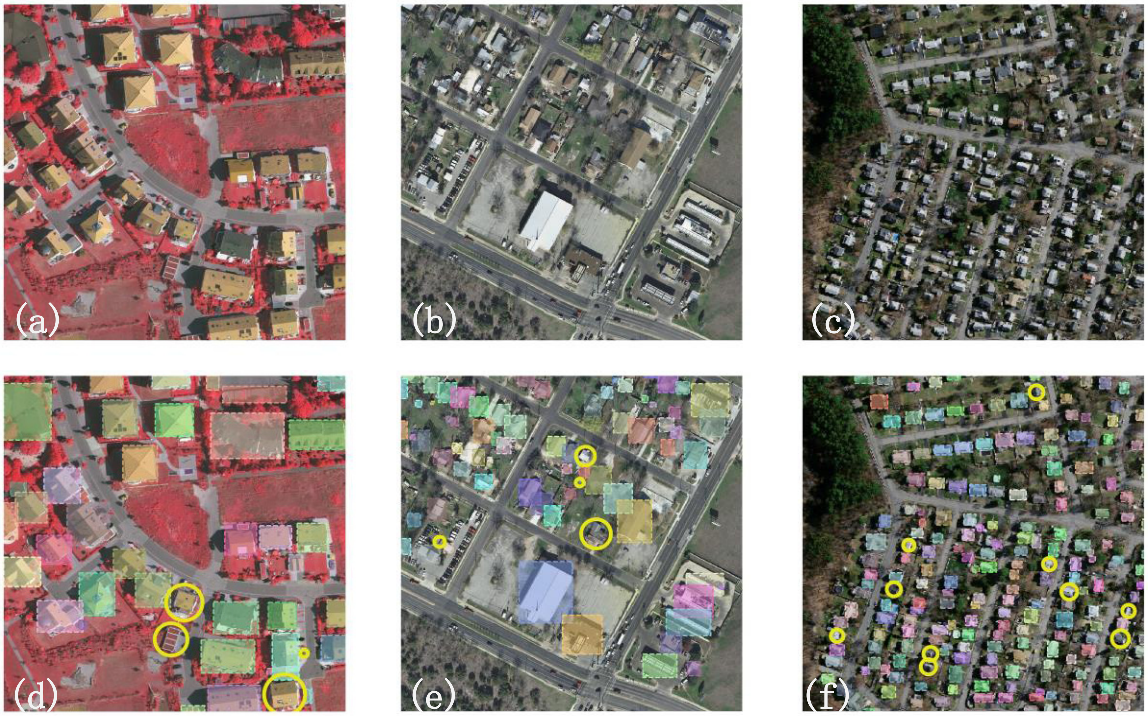

Fig. 9. Results of object detection using the Mask R-CNN. The first row shows three images from the three datasets, respectively, i.e., (a) image of ISPRS, (b) image of INRIA, and (c) image of Massachusetts building datasets. The images coupled with localized buildings are shown in the second row, i.e., (d) object detection results of the image in (a), (e) object detection results of the image in (b), and (f) object detection results of the image in (c).

2) Corner Versus Vertex: Buildings are surface artifacts with special geometries, and it is natural to use the corner points of the rooftop as the key points during manual marking. The corner itself is a relatively special visual element in aerial images. In order to explore the impact of corners on delineation of rooftops from images, we conducted the following experiments. In addition to using corners on the roofs of each building, we added an intermediate point on each edge as vertices for additional annotations in the training sample.

The right-most column in Fig. 8 shows the results of the OEC-RNN learning with more vertices on rooftop boundaries. It can be seen from the third row in the column that many sharp corners of building are replaced with some vertices, whereas the polygons have more "round" corners. The results show that it is very important to replace the corners with vertices during the learning phase.

Table II shows the quantitative evaluation results of the experiment. After the training of edge points is added, the overall accuracy of rooftop extraction decreases, which is in line with our assumption that edge points could be noise in terms of learning the sharp corners. As shown in Table II, the VNE becomes significantly higher that of the OEC-RNN. It means that the OEC-RNN would miss predicting the vertices when more vertices instead of corners are added in to training samples. Through the ablation experiments in Section III-E, the performance of the model can be effectively improved by using both edge attention and training with building corner points.

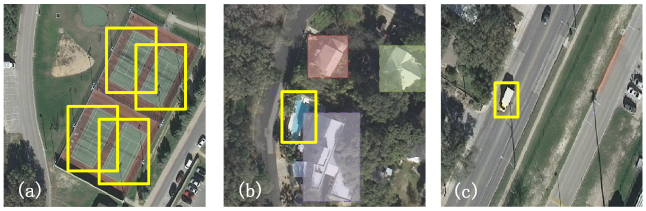

Fig. 10. Result of being incorrectly detected as a building object. (a) Tennis courts. (b) Swimming pool. (c) Large van.

IV. Discussion

A. Object Detection

The first step of the OEC-RNN is to localize buildings using the Mask R-CNN from a large image. As shown with yellow circles in Fig. 9, some buildings with small size are often misdetected by the Mask R-CNN. In addition, there are some ground objects that are inevitably incorrectly detected as a building, such as tennis courts, swimming pools, and vans, which often have uniform colors and regular geometric shapes. The specific examples in the images are shown in Fig. 10.

Whether missed or incorrectly detected, this ultimately has a significant impact on image-based building extraction, which makes the object detection process important in the overall workflow. Since the methods used for building object detection in this article are independent of the building outline delineation sections followed, other object detection methods (i.e., RetinaNet [37], single shot multibox detector (SSD) [38],

TABLE III
PERFORMANCE OF OBJECT DETECTION

	Precision	Recall	F1-score
ISPRS	0.9653	0.9254	0.9449
INRIA	0.9540	0.9132	0.9331
Massachusetts	0.9206	0.8854	0.9026

TABLE IV
PERFORMANCE OF THE OEC-RNN WORKFLOW

	IoU	WCov	BoundF	RE	Hd	VNE
ISPRS	0.8056	0.7521	0.7026	0.7758	9.26	8.62
INRIA	0.7447	0.7155	0.6899	0.6385	7.15	5.20
Massachusetts	0.7156	0.7039	0.6728	0.6492	6.22	4.95

CornerNet [39], and YOLOv4 [40]) could be replaced or extended within the framework of the OEC-RNN.

Table III lists the evaluation metrics in terms of object detection performance on three different datasets using Mask R-CNN. The F1-score value of **ISPRS-Vaihingen** is 0.9449, which is higher than both **INRIA-Austin** and **Massachusetts**. The recall in **Massachusetts** is 0.8854, which is poor in three datasets. The lower resolution of the **Massachusetts** building dataset causes a building of the same size to appear as smaller in the image of **Massachusetts**, and thus more buildings are lost in the object detection process. The results show that the Mask R-CNN can obtain better results on images with higher spatial resolution than that with a lower one.

In the practical application, the detection of building objects and the depiction of building rooftops are a unified workflow. The accuracy of object detection has a significant impact on the final mapping, including missing and incorrect detection of building objects. Table IV lists the results of performance of the complete experimental procedure. It shows that the performance of all metrics has declined due to the incorrect object detection. When the object detection model leads to the missing of buildings, the value of IoU corresponding to the building in ground truth is 0. We only calculated the IoU of the detected building with the ground truth.

Compared to the result of using the given ground-truth object box to crop the image as input, there is a very significant degradation in the performance of the result using the Mask R-CNN object detection bounding box. The OEC-RNN uses a staged building extraction framework in this article: object detection is carried out first, followed by careful mapping of building objects by sequentially predicting the corner points of building rooftops. The phased workflow inevitably causes errors in the different phases that are passed on to the subsequent work.

B. Backbone Architecture

In this article, the backbone used by the OEC-RNN to extract features is Resnet-101. In Section III, different backbones are used in the compare methods, for example, the backbone in the original Polygon-RNN is VGG-16, and the Resnet-50 is adopted in both polygon RNN++ and Curve-GCN. To investigate the impact of different backbones on the performance, we evaluate the performance of the OEC-RNN using one of the four backbones, i.e., VGG-16, Inception-V4 [41], ResNet-50, and ResNet-101. The experimental results are listed in Table V.

TABLE V
PERFORMANCE OF THE OEC-RNN USING DIFFERENT BACKBONES

	Backbone	IoU	WCov	BoundF	RE	Hd	VNE
ISPRS	VGG-16	0.8106	0.7846	0.7254	0.7685	5.27	5.78
	Inception-v4	0.8242	0.8025	0.7458	0.7656	5.09	5.61
	ResNet-50	0.8985	0.8711	0.8056	0.7471	4.33	5.16
	ResNet-101	**0.9002**	**0.8784**	**0.8075**	0.7459	4.28	**5.05**
INRIA	VGG-16	0.7345	0.7109	0.6613	0.6041	3.66	4.95
	Inception-v4	0.7618	0.7366	0.6752	0.5988	3.59	4.72
	ResNet-50	0.8129	0.7781	0.7306	0.5628	2.66	3.07
	ResNet-101	**0.8144**	**0.7776**	**0.7293**	0.5653	2.72	**3.12**
Massachusetts	VGG-16	0.7011	0.7219	0.6852	0.7435	3.52	4.17
	Inception-v4	0.7243	0.7368	0.6994	0.7158	3.44	4.06
	ResNet-50	0.7613	0.7522	0.7198	0.7014	2.67	3.94
	ResNet-101	**0.7755**	**0.7544**	**0.7212**	0.6838	2.36	**3.21**

TABLE VI
AVG. INFERENCE TIME PER OBJECT

	Time(ms)	fps
Polygon-RNN	240.1	4.1
ACDRNet	116.2	8.6
Deep Snake	28.4	34.6
Curve-GCN	27.9	35.8
OEC-RNN	60.1	16.7

It can be seen from Table V that the performance of the ResNet-50 is almost the same as that of the ResNet-101, and both of them are superior to other backbones. Therefore, it is better to choose the ResNet as the backbone in the OEC-RNN. However, the ResNet-101 instead of ResNet-50 is not the key point for the OEC-RNN to achieve improved performance.

C. Inference Times

We compared the running efficiency of different contour-based methods in the same device environment. In the INRIA dataset, each 5000 × 5000 image contains an average of 765 building objects. Therefore, the unit of measurement of computational efficiency is the building object. Timings are reported in Table VI.

In the Polygon-RNN, the vertices of the outer contour of the object are used for training. Many noncorner points need to be

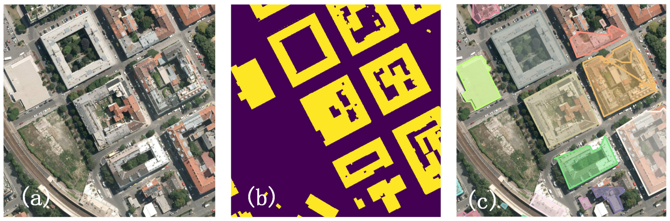

Fig. 11. Failure case of Vienna in INRIA datasets. (a) Aerial image. (b) Ground truth. (c) Results of the OEC-RNN.

predicted during inference. And the number of corner points of buildings is less for general buildings, thus OEC-RNN can infer faster when predicting corner points. Contour-based methods, like Curve-GCN and Deep Snake, need to be given a fixed initial condition: the number of control points that define the contour and that control points are much more than the corner points of the buildings, which causes the building contour to show a blob-like shape.

The average time used by the OEC-RNN to infer a single building is 60.1 ms, which is slower than the 27.9 ms of Curve-GCN and the 28.4 ms of Deep Snake. However, the frames per second can still reach 16, which is acceptable for real-time applications.

D. Limitation Due to the Biased Assumption

In addition to requiring regular edges for rooftops, another important assumption in the current instance segmentation by modeling the outline of a building as a polygon is that the rooftop is assumed to be a hole-free polygon. Fig. 11 shows the aerial image, ground truth, and polygon drawing results. Under the condition of accurate object detection, the OEC-RNN cannot correctly delineate the rooftop polygon with multiple holes. These methods are inclined to delineate the outer contours of the rooftop and neglect the inner geometric characteristics of the polygons.

V. Conclusion

In this article, we present a multitask learning approach to predict rooftop corners in a sequent way using the edge attention learned from where the boundaries are in a given image region. Experimental results show that the OEC-RNN achieved the best results in polygon delineation compared with stat-of-the-art methods in terms of both qualitative and quantitative evaluation methods. This article demonstrates that building rooftops can be more accurately represented by geometric elements such as points, lines, and polygons.

There are some limitations to the existing approach, such as the difficulty in depicting complex shaped rooftop and no end-to-end learning styles, which can be improved in the future.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2012, pp. 1097–1105.

[2] J. Yuan, "Learning building extraction in aerial scenes with convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 11, pp. 2793–2798, Nov. 2018.

[3] Q. Li, Y. Shi, X. Huang, and X. X. Zhu, "Building footprint generation by integrating convolution neural network with feature pairwise conditional random field (FPCRF)," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 7502–7519, Nov. 2020.

[4] L. Sun, Y. Tang, and L. Zhang, "Rural building detection in high-resolution imagery based on a two-stage CNN model," *IEEE Geosci. Remote Sens. Lett.*, vol. 14, no. 11, pp. 1998–2002, Nov. 2017.

[5] H. L. Yang, J. Yuan, D. Lunga, M. Laverdiere, A. Rose, and B. Bhaduri, "Building extraction at scale using convolutional neural network: Mapping of the united states," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 11, no. 8, pp. 2600–2614, Aug. 2018.

[6] S. Ji, S. Wei, and M. Lu, "Fully convolutional networks for multisource building extraction from an open aerial and satellite imagery data set," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 1, pp. 574–586, Jan. 2019.

[7] R. D. Majd, M. Momeni, and P. Moallem, "Transferable object-based framework based on deep convolutional neural networks for building extraction," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 12, no. 8, pp. 2627–2635, Aug. 2019.

[8] Y. Xie *et al.*, "Refined extraction of building outlines from high-resolution remote sensing imagery based on a multifeature convolutional neural network and morphological filtering," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 1842–1855, 2020.

[9] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, Apr. 2017.

[10] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, Apr. 2014, pp. 740–755.

[11] N. Yang and H. Tang, "GeoBoost: An incremental deep learning approach toward global mapping of buildings from VHR remote sensing images," *Remote Sens.*, vol. 12, no. 11, p. 1794, Jun. 2020.

[12] L. Zhang *et al.*, "Learning deep structured active contours end-to-end," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 8877–8885.

[13] D. Cheng, R. Liao, S. Fidler, and R. Urtasun, "DARNet: Deep active ray network for building segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7431–7439.

[14] K. Zhao, J. Kang, J. Jung, and G. Sohn, "Building extraction from satellite images using mask R-CNN with building boundary regularization," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2018, pp. 242–246.

[15] S. Wei, S. Ji, and M. Lu, "Toward automatic building footprint delineation from aerial images using CNN and regularization," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 3, pp. 2178–2189, Mar. 2020.

[16] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to state-of-the-art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012.

[17] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[18] A. Lucchi, K. Smith, R. Achanta, G. Knott, and P. Fua, "Supervoxel-based segmentation of mitochondria in EM image stacks with learned shape features," *IEEE Trans. Med. Imag.*, vol. 31, no. 2, pp. 474–486, Feb. 2012.

[19] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 386–397, Feb. 2020.

[20] S. Gur, T. Shaharabany, and L. Wolf, "End to end trainable active contours via differentiable rendering," 2019, *arXiv:1912.00367*. [Online]. Available: http://arxiv.org/abs/1912.00367

[21] L. Castrejon, K. Kundu, R. Urtasun, and S. Fidler, "Annotating object instances with a polygon-RNN," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4485–4493.

[22] D. Acuna, H. Ling, A. Kar, and S. Fidler, "Efficient interactive annotation of segmentation datasets with Polygon-RNN++," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 859–868.

[23] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, Jan. 1988.

[24] S. Peng, W. Jiang, H. Pi, X. Li, H. Bao, and X. Zhou, "Deep snake for real-time instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8530–8539.

[25] H. Ling, J. Gao, A. Kar, W. Chen, and S. Fidler, "Fast interactive object annotation with curve-GCN," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 5252–5261.

[26] Y. Liu *et al.*, "Richer convolutional features for edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 1939–1946, Aug. 2019.

[27] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," 2015, *arXiv:1506.04214*. [Online]. Available: http://arxiv.org/abs/1506.04214

[28] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2015, pp. 91–99.

[29] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*. [Online]. Available: http://arxiv.org/abs/1512.03385

[30] *International Society for Photogrammetry and Remote Sensing, 2D Semantic Labeling Contest*. Accessed: Nov. 3, 2020. [Online]. Available: http://www2.isprs.org/commissions/comm3/wg4/semantic-labeling.html

[31] E. Maggiori, Y. Tarabalka, G. Charpiat, and P. Alliez, "Can semantic labeling methods generalize to any city? The inria aerial image labeling benchmark," in *Proc. IEEE Int. Geosci. Remote Sens. Symp. (IGARSS)*, Jul. 2017, pp. 3226–3229.

[32] V. Mnih, "Machine learning for aerial image labeling," Ph.D. dissertation, Dept. Comput. Sci., Univ. Toronto, Toronto, ON, Canada, 2013.

[33] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Jun. 2010.

[34] S. Wang *et al.*, "Toronto city: Seeing the World with a million eyes," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 3028–3036.

[35] F. Perazzi, J. Pont-Tuset, B. Mcwilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung, "A benchmark dataset and evaluation methodology for video object segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Dec. 2016, pp. 724–732.

[36] J. Avbelj, R. Müller, and R. Bamler, "A metric for polygon comparison and building extraction evaluation," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 1, pp. 170–174, Jan. 2015.

[37] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, Feb. 2020.

[38] W. Liu *et al.*, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Oct. 2016, pp. 21–37.

[39] H. Law and J. Deng, "CornerNet: Detecting objects as paired keypoints," *Int. J. Comput. Vis.*, vol. 128, no. 3, pp. 642–656, Mar. 2020.

[40] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "YOLOv4: Optimal speed and accuracy of object detection," 2020, *arXiv:2004.10934*. [Online]. Available: http://arxiv.org/abs/2004.10934

[41] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. Alemi, "Inception-v4, inception-ResNet and the impact of residual connections on learning," 2016, *arXiv:1602.07261*. [Online]. Available: http://arxiv.org/abs/1602.07261

Wei Huang received the M.S. degree in photogrammetry and remote sensing from the Shandong University of Science and Technology, Qingdao, China, in 2017.

His interests focus on remote sensing image processing and pattern recognition.

Hong Tang (Member, IEEE) received the B.S. and M.S. degrees from the China University of Mining and Technology, Xuzhou, China, in 1998 and 2001, respectively, and the Ph.D. degree in pattern recognition and intelligence systems from Shanghai Jiao Tong University, Shanghai, China, in 2006.

From March 2006 to March 2008, he worked as a Post-Doctoral Researcher in IMEDIA Project with INRIA, Paris, France. He is a Professor with Beijing Normal University, Beijing, China. His research interests include remote sensing image processing, pattern recognition, and natural disaster reduction.

Penglei Xu received the B.S. degree in engineering of surveying and mapping from Hohai University, Nanjing, China, in 2019. He is pursuing the M.S. degree with the Faculty of Geographical Science, Beijing Normal University, Beijing, China.

His research interests include remote sensing image processing and pattern recognition.