3-D Instance Segmentation of MVS Buildings

Jiazhou Chen, Yanghui Xu, Shufang Lu, Ronghua Liang, *Senior Member, IEEE*, and Liangliang Nan

***Abstract*— We present a novel 3-D instance segmentation framework for multiview stereo (MVS) buildings in urban scenes. Unlike existing works focusing on semantic segmentation of urban scenes, the emphasis of this work lies in detecting and segmenting 3-D building instances even if they are attached and embedded in a large and imprecise 3-D surface model. Multiview red green blue (RGB) images are first enhanced to RGB height (RGBH) images by adding a heightmap and are segmented to obtain all roof instances using a fine-tuned 2-D instance segmentation neural network. Instance masks from different multiview images are then clustered into global masks. Our mask clustering accounts for spatial occlusion and overlapping, which can eliminate segmentation ambiguities among multiview images. Based on these global masks, 3-D roof instances are segmented out by mask back-projections and extended to the entire building instances through a Markov random field optimization. A new dataset that contains instance-level annotation for both 3-D urban scenes (roofs and buildings) and drone images (roofs) is provided. To the best of our knowledge, it is the first outdoor dataset dedicated to 3-D instance segmentation with much more annotations of attached 3-D buildings than existing datasets.[1] Quantitative evaluations and ablation studies have shown the effectiveness of all major steps and the advantages of our multiview framework over the orthophoto-based method.**

***Index Terms*— 3-D urban scene, dataset, instance segmentation, multiview clustering.**

I. Introduction

IN RECENT decades, the multiview stereo (MVS) technique has been widely used in the geographic information system (GIS) domain. Multiview images are captured by unmanned aerial vehicles (UAVs) and used to automatically reconstruct dense 3-D mesh models of large urban scenes [1], [2]. The reconstructed 3-D mesh models provide a visually pleasing representation of urban scenes. However, due to the lack of semantic information, they can hardly be used directly in various real-world applications, such as urban planning, simulation, and solar potential estimation.

Buildings are the most important part of a city, and its segmentation is the core of the semantic analysis of urban scenes. Rather than semantic segmentation, we focus on the instance segmentation of buildings, as it separates different building instances, even if they are attached. Thus, our goal is to segment all building instances in a large 3-D urban scene precisely and automatically, as shown in Fig. 1.

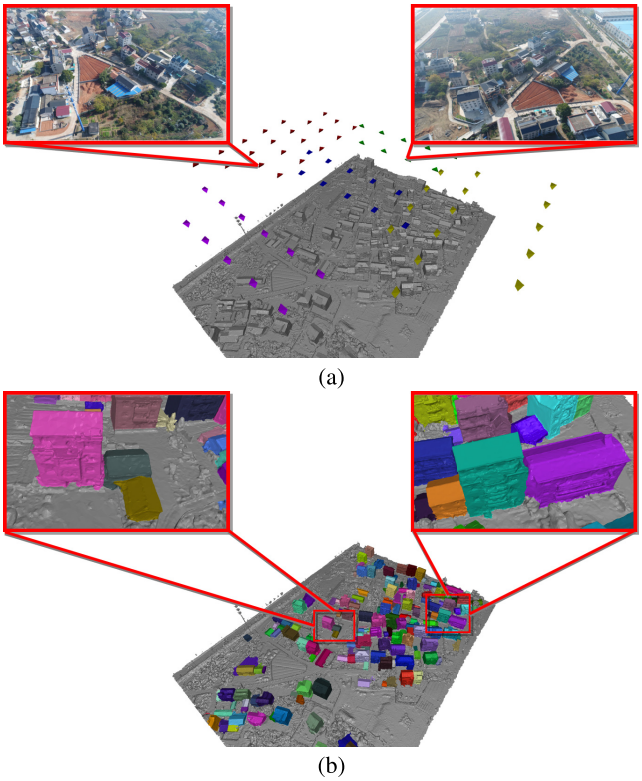


Fig. 1. Instance segmentation of 3-D buildings in a large urban scene. The pyramids above the 3-D scene model in (a) indicate the position and orientation of the cameras. (a) Input: 3-D model and (optional) UAV images. (b) Output: 3-D building instances.

Recent advances in deep learning have achieved great success in image instance segmentation, but applying these techniques to 3-D mesh models is still challenging and has not been sufficiently explored, especially for 3-D buildings in large-scale urban scenes. Feature extraction of 3-D data is the key to applying deep learning to 3-D model segmentation. Volumetric-based methods [3], [4] and point cloud-based methods [5], [6] are limited by memory and computing power, thus are mainly used for the segmentation of relatively small indoor scenes [7]–[12]. Besides, the lack of annotated 3-D instance segmentation datasets for outdoor scenes also hinders the application of deep learning on instance segmentation of 3-D urban scenes.

Instead of directly segmenting 3-D models, segmenting images first and projecting them to the 3-D models is a potential alternative, as it can utilize powerful neural networks for image segmentation. Orthophoto maps could be the first candidate, due to their unified projection directions. However, buildings in orthophoto maps have severe self-occlusion, for example, walls cannot be seen. Thus, its

Manuscript received 15 August 2021; revised 14 April 2022; accepted 21 May 2022. Date of publication 16 June 2022; date of current version 30 June 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 62172367 and in part by the Natural Science Foundation of Zhejiang Province under Grant LGF22F020022. *(Corresponding author: Ronghua Liang.)*

Jiazhou Chen, Yanghui Xu, Shufang Lu, and Ronghua Liang are with the School of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310014, China (e-mail: rhliang@zjut.edu.cn).

Liangliang Nan is with the Faculty of Architecture and the Built Environment, Delft University of Technology, 2628BL Delft, The Netherlands.

Digital Object Identifier 10.1109/TGRS.2022.3183567

[1]The datasets are available at https://californiachen.github.io/datasets/InstanceBuilding

segmentation inaccuracy will be conducted to the 3-D models during the 2-D–3-D projection. For this sake, we employ a multiview 3-D segmentation framework in this article. It first employs existing learning-based 2-D instance segmentation to segment roofs in drone images, back-projects roof instance masks to the 3-D scenes considering the spatial occlusion, and finally constructs Markov random fields to segment out all buildings from 3-D scenes.

Such a multiview framework for 3-D semantic segmentation tasks is straightforward, and a winner-take-all mechanism works well for most semantic segmentation tasks. However, a multiview framework for 3-D instance segmentation is much more challenging, as instances have to keep separating even if they are spatially connected. Projecting instance masks back to the 3-D scene without correspondences will lead to instance ambiguities, as 2-D instance masks from different views may be inconsistent. Buildings in multiview images are often partially occluded by attached other buildings or tall trees, which increases ambiguities in mask correspondences.

In this article, we propose a novel instance mask clustering method to build mask correspondences among multiview UAV images. It solves the issue of instance ambiguities robustly, making our multiview 3-D instance segmentation method outperform the orthophoto-based method. To improve segmentation accuracy for diversely distributed buildings, we enhance multiview red green blue (RGB) images to RGB height (RGBH) images by adding an extra channel encoding the height information. To ease the spatial occlusion challenge, we perform instance segmentation of roofs instead of entire buildings, since roofs have higher visibility than other parts of buildings.

Though our method incorporates existing image instance segmentation techniques, it includes the following core contributions.

1) A multiview instance segmentation framework that segments 3-D buildings in large urban scenes efficiently and precisely.
2) An occlusion-aware clustering method for instance masks, which robustly eliminates ambiguities in mask correspondences among multiview images.
3) A benchmark dataset *InstanceBuilding* for instance segmentation evaluation of 3-D buildings in large urban scenes, which consists of pixel-level instance annotation for both UAV images and 3-D urban models.

II. Related Work

There is a large volume of research in instance segmentation for various data sources. In this section, we review the existing work of instance segmentation for common images, aerial images, and 3-D data.

A. Common Image Instance Segmentation

Existing instance segmentation methods for natural images can be classified into two categories: object-detection-based approaches and metric learning-based ones.

1) Object-Detection-Based Approaches: Object-detection-based approaches work in a top-down manner and highly depend on object detection or proposal. R-CNN first introduces convolutional neural network (CNN) in the field of object detection [13]. To improve computational efficiency, Fast R-CNN proposes an improved spatial pyramid pooling (SPP) [14] structure named RoI pooling [15], and Faster R-CNN uses region proposal networks instead of selective searching to extract object candidates, which forms an efficient end-to-end network for object detection [16].

Based on Faster R-CNN [16], Mask R-CNN combines with feature pyramid network (FPN) [17] to detect objects with different sizes and uses RoIAlign instead of RoI pooling to form a simple, flexible, and effective instance segmentation network [18]. Recently, mask scoring R-CNN uses mask scores to improve the category scores used in Mask R-CNN [19]. Path aggregation network for instance segmentation (PANet) uses a bottom-up annotation structure to shorten the information path and enhances the feature pyramid with accurate localization signals existing at low levels [20]. Hybrid task cascade for instance segmentation (HTC) combines detection and segmentation with a multitask and multistage hybrid cascade structure [21]. Swin Transformer [22] proposes a hierarchical Transformer whose representation is computed with shifted windows. Its hierarchical architecture has flexibility at various scales and linear computational complexity concerning image size, which helps it to achieve outstanding segmentation performance.

2) Metric Learning-Based Approaches: Many other dense instance segmentation methods are based on metric learning [23]. These methods work in a bottom-up manner, generate embedding features [24], [25] for each pixel and use post-processing methods such as clustering [26], [27] or graph theory [28] to classify these pixels. Inspired by full convolutional instance-aware semantic segmentation (FCIS) [29] and You Only Look At Coefficients (YOLACT) [30], BlendMask uses a blender module to merge top-level coarse instance information with lower-level fine granularities [31].

B. Aerial Image Segmentation

In the last decade, instance segmentation methods for aerial images of urban scenes have also been proposed because of the wide applications of aerial images. Montoya *et al.* use an α-shape algorithm to calculate the boundary polygons of building objects, which are further optimized by conditional random field (CRF) [32]. By combining the CNN backbone with FPN and recurrent neural network (RNN), Li *et al.* propose an end-to-end deep neural network to predict polygon outlines of buildings and road topology maps [33]. Convolutional message passing neural network for structured outdoor architecture reconstruction (Conv MPN) uses GNN (graph neural network) [34] to reconstruct the building plan from a single image [35]. Deep active ray network for building segmentation (DARNet) employs a polar representation of contours to predict contours that are free of self-intersection and a loss function consisting of a data term, a curvature term, and a balloon term, which not only encourages the predicted contours to match ground-truth building boundaries but also prefers low-curvature solutions [36].

Besides instance segmentation, many semantic segmentation methods for aerial images have been proposed recently in the remote-sensing domain. They are also referred to as remote-sensing image classification. Besides single-modality images, researchers in the remote-sensing domain are also

interested in employing deep learning techniques in the pixel-level classification of multimodality images, including multispectral ones and hyperspectral ones, which are proved to overcome the challenge of information diversity [37]. For instance, Hong *et al.* introduce graph convolutional networks into hyperspectral image classification in a minibatch fashion [38] and also propose a new transform-based network that learns locally spectral representations from multiple neighboring bands instead of single bands [39]. Since multispectral and hyperspectral images require expensive and heavy spectrometers to acquire, RGB images are more common for UAVs. In this article, height maps are automatically generated and added to corresponding RGB images, respectively. They cannot provide as rich information as hyperspectral images, but this geometric information is a very important supplement that can significantly improve segmentation accuracy.

C. 3-D Instance Segmentation

Unlike images that inherently have a grid structure, the vertices and faces in discrete surfaces (i.e., 3-D meshes) do not have regular spatial structures to be directly convoluted. Volumetric methods ease this issue by using a 3-D grid representation, which is notoriously expensive in terms of computational efficiency and memory consumption [4], [12], [40]–[43].

Various strategies have been proposed to address the memory issue of volumetric methods. For example, OctNet uses an octree structure to avoid unnecessary cells [3], thus reducing memory consumption. PointNet uses T-net and max-pooling to achieve rotation invariance and the capability of handling unordered 3-D point clouds. It fuses both local and global features, making it an efficient and effective feature extractor for point cloud data [5]. Through point grouping and multilevel feature extraction, PointNet++ can better extract discriminative features for point clouds with uneven density [6].

Based on features extracted by PointNet, PointNet++, and PointCNN [44], many 3-D instance segmentation methods for point clouds have also been proposed. Similarity group proposal network for 3-D point cloud instance segmentation (SGPN) predicts the instances by learning the similarity matrix between point clouds. However, the size of its similarity matrix tends to explode as the number of points increases [8]. Generative shape proposal network for 3-D instance segmentation in point cloud (GSPN) extends the structure of Mask R-CNN (that was originally developed for images) to process 3-D data [7]. Joint instance and semantic segmentation of 3-D point clouds (JSNet) [45] and Associatively segmenting instances and semantics in point clouds (ASIS) [10] both learn the instance embedding space and combine semantic features and instance features of the point clouds to jointly improve the accuracy of semantic segmentation and instance segmentation.

Great progress has been made for instance segmentation of indoor scenes. However, existing methods are designed for processing point cloud data. It is still a challenge to extend these methods to outdoor scenes without sufficient annotated data and generalize them to handle the fast accumulation of urban models in the form of meshes. In this work, we establish a 3-D instance segmentation dataset for urban scenes and propose the first framework for 3-D instance segmentation of buildings from urban MVS meshes.

III. Methodology

Compared to the lower parts of buildings that are more likely to be occluded by the nearby buildings and trees, building roofs usually have better visibility in aerial imaging. This observation motivates us to approach instance segmentation of entire buildings by looking into the segmentation of building roofs. In contrast to the previous work directly segmenting entire buildings [46], we perform roof segmentation by using a deep neural network. This strategy significantly improves the accuracy of the segmentation stage and simplifies the manual annotation in the data preparation stage.

One characteristic of our method is the hybrid process of 2-D images and the corresponding 3-D meshes, in which spatial occlusion is fully considered in processing the two distinctive data sources. Fig. 2 shows the proposed multiview 3-D instance segmentation framework that consists of three major steps as follows.

1) *2-D Roof Instance Segmentation:* Roofs in multiview images are automatically segmented by an instance segmentation neural network that is fine-tuned using our RGBH imagery dataset.
2) *Instance Mask Clustering:* An occlusion-aware clustering method for roof instance masks is exploited, which correlates instance masks from multiview images to eliminate ambiguities. The mask clustering is the core of our method, which projects the 3-D urban scene to the image space to measure the spatial overlap between arbitrary pairs of instance masks.
3) *3-D Building Instance Segmentation:* The clustered masks are projected back to the 3-D space to segment 3-D roof instances and the entire buildings are segmented in the end through an Markov random field (MRF) optimization.

In the following part of this section, we introduce these three major steps, the benchmark dataset, and the implementation details.

A. 2-D Roof Instance Segmentation

The challenge in building instance segmentation lies in that the roofs of adjacent (or even attached) buildings may have very similar appearances despite the difference in the height of the buildings. In this work, we take advantage of the complementary characteristics of the images and the 3-D model of the scene by enhancing each RGB aerial image to an RGBH image that provides additional geometric cues. Specifically, we render a heightmap for each drone image using the 3-D urban models reconstructed from the drone images and camera parameters. We add an additional channel encoding the height information to the drone images to obtain RGBH images, in which the height values are separately normalized for each image. With the RGBH images, we apply Swin Transformer [22] to segment roof instances automatically.

According to our quantitative evaluation of the benchmark dataset, the average precision (AP) [47] of the segmentation of roof instances on RGBH images reaches 0.582, which is significantly higher compared to the 0.563 achieved on RGB aerial images. This demonstrates the advantage of height information on the roof instance segmentation. A visual comparison is shown in Fig. 3. Ground objects like trees and vegetable fields are successfully separated from buildings, even though

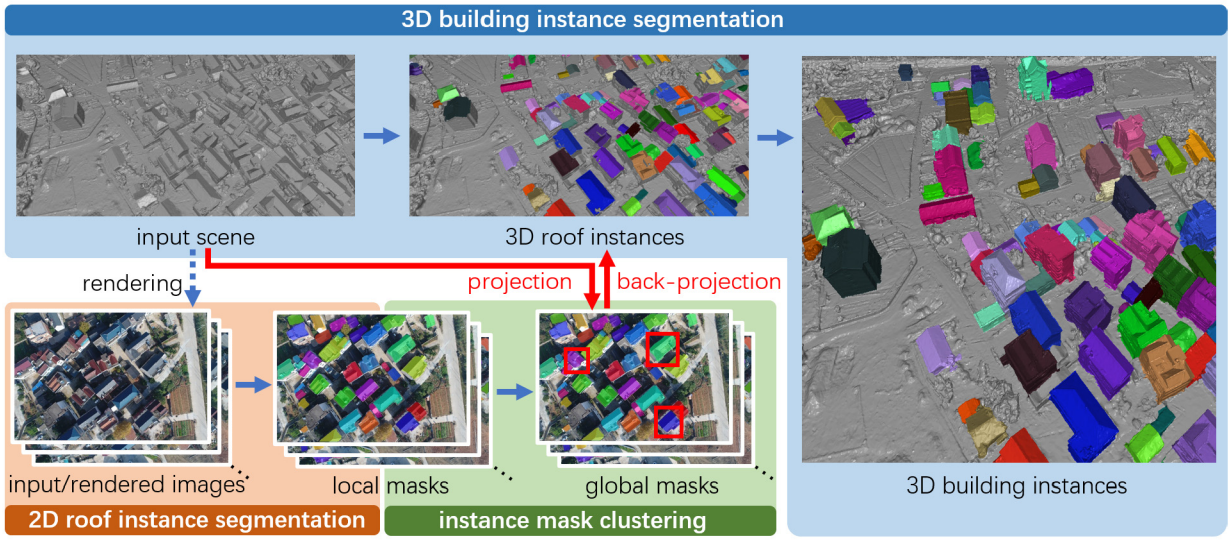


Fig. 2. Overview of the proposed method. Our method takes a 3-D urban scene and optionally multiview UAV images as input and segments all 3-D building instances as results. It contains three major steps: 2-D roof instance segmentation, instance mask clustering, and 3-D building instance segmentation. The multiview images are not obligatory, as they can also be generated by the rendering of the input 3-D scene with textures (noted by the dotted arrow in the figure). The red rectangles highlight a few global masks selected by our clustering method. The projection and back-projection operations noted by the red arrows in the figure contribute to both instance mask clustering and the occlusion-aware 3-D roof segmentation.

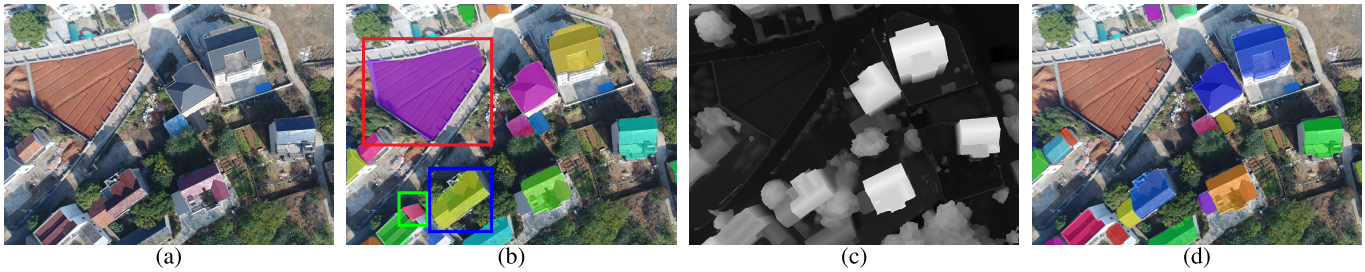


Fig. 3. Comparison of the segmentation results without and with height information. (a) Input test drone image. (b) Segmentation result using only the RGB image. The rectangles highlight three misclassified regions, that is, a vegetable field (in red) and a wall (in green) of a building are segmented as roofs, and two roofs of attached buildings (in blue) are not separated. (c) Heightmap obtained by rendering the scene using the 3-D model and the camera parameters. (d) Segmentation result using both the RGB image and the heightmap, where the vegetable field, wall, and roofs are all correctly separated.

some of them have visually indistinguishable textures from building roofs. The Swin Transformer neural network also computes a probability for each instance mask to represent its prediction confidence. To avoid low-confident masks, we only use roof instance masks whose prediction confidence is higher than 70%.

B. Instance Mask Clustering

After the roof instance segmentation, we obtain a set of roof instance masks from multiview images, where multiple masks may correspond to the same roof. Since roofs of the same building in multiple views have been segmented independently, the correspondences between roof instance masks are not known. This results in the number of instance masks being much larger than the number of roofs in the scene. To establish the correspondences between the masks from multiview images, we refer to 3-D roof instances by back-projecting the 2-D roof masks onto the 3-D model of the scene using the camera parameters. However, identifying masks that correspond to the same roof is challenging due to two main reasons. First, the 2-D instance segmentation may have errors due to the limited capability of the neural network and the complex structure of the building roofs, as shown in the first row of Fig. 4. Second, the instance masks from different views are ambiguous due to different levels of spatial occlusion, as shown in the second row of Fig. 4.

To tackle these two challenges, we propose an instance mask clustering method that divides instance masks into different groups such that each group corresponds to a unique roof instance of an individual building. Representative masks are first selected from the segmented instance masks, and the remaining masks are merged with them according to mask similarity measures. For clarity, we refer to all roof instance masks in multiview images as local masks, while the representative masks selected for clustering as global masks, as they represent unique building roofs across different images.

We first build a similarity matrix $\mathcal{M}$ to measure the spatial overlap for each pair of local masks. Based on $\mathcal{M}$, a mask with confidence value $\mathcal{C}$ to be selected as a global mask is computed for each local mask. Finally, all local masks are sorted in descending order to select reliable global masks and are clustered into groups according to their similarities with the global masks. Note that each mask group contains only one global mask. We establish the mapping between all local masks and global masks. In the following, we elaborate on these steps in detail.

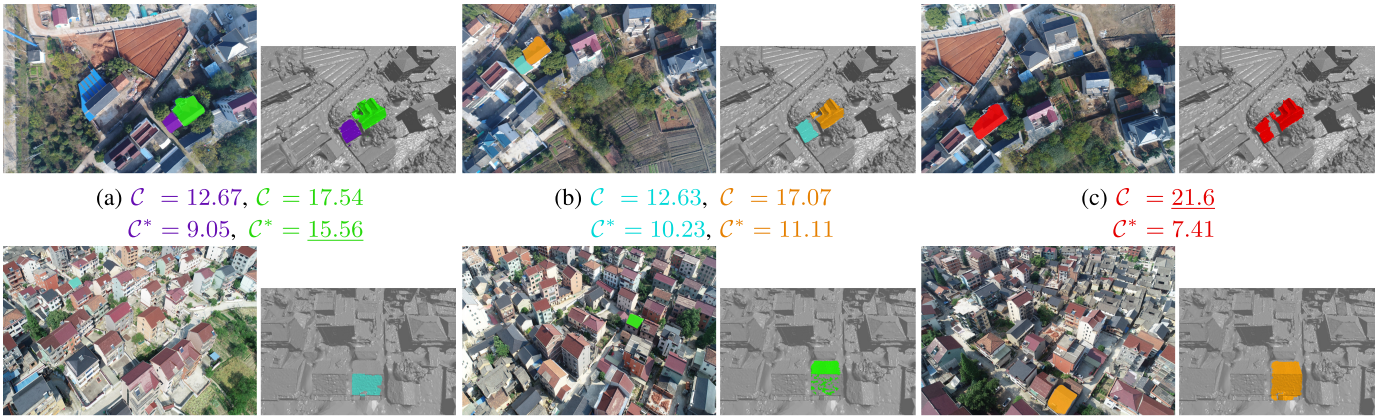


Fig. 4. Ambiguities in 2-D roof instance segmentation. In each row, instance masks are shown in both the drone images (in different views) and the corresponding 3-D mesh models (in an identical view). Top row: two roofs are separated in (a) and (b) but incorrectly mixed together in (c). Bottom row: (d) and (e) only cover a small part of the same roof, while (f) covers the entire roof. Original mask confidence values (denoted by $\mathcal{C}$) and improved mask confidence values (denoted by $\mathcal{C}^*$) are given in the subcaptions. The underlined numbers are the highest confidence values in each row.

1) Occlusion-Aware Mask Similarity: To measure the spatial overlap of two masks, we project 3-D mesh triangles to the image using the camera parameters. We render a depth map with the graphics processing unit (GPU) acceleration for each multiview image and employ a depth test to check the visibility for all the vertices. For the ith local mask, we record a set of triangles S_i whose centers are projected within this local mask region. A similarity matrix $\mathcal{M}_{n\times n}$ is then computed to quantify the spatial overlap between every pair of local masks, where n is the number of all local masks. The similarity element m_{ij} measures the intersection over union (i.e., *IoU*) between the ith and the jth local masks, that is,

$$m_{ij} = A(S_i \cap S_j)/A(S_i \cup S_j) \tag{1}$$

where $A(S)$ is the surface area of the triangles in the set S. $\mathcal{M}_{n\times n}$ is a symmetric matrix as $m_{ij} = m_{ji}$.

2) Mask Confidence: Generally, an ideal global mask should have the most overlap with local masks that correspond to the same roof and have the least overlap with local masks that correspond to roofs of different buildings. To select such global masks, we estimate a confidence value $\mathcal{C}$ for each local mask to evaluate the overall overlap with all other local masks in the scene. It is calculated as the sum of similarity elements on the ith row of the similarity matrix $\mathcal{M}$

$$\mathcal{C}_i = \mathcal{P}_i \cdot \sum_{j=1}^{n} \mathcal{P}_j \cdot m_{ij} \tag{2}$$

where $\mathcal{P}_i$ is the probability value produced by the Swin Transformer neural network, and $\mathcal{C}_i$ sums up the probability-weighted similarity elements of local masks. It makes sense because local masks with higher prediction confidences are more likely to be global masks.

However, there is still one drawback in (2): local masks with large areas may suppress smaller ones because they likely overlap more with other local masks, and thus they obtain larger mask confidence values. Local masks with larger areas are not always the ideal global masks. The top row of Fig. 4 shows such a counter-example. To solve this issue, we define a binary term Δ_{ij} to avoid such unexpected suppression

$$\Delta_{ij} = \delta(m_{ij} - \beta) \tag{3}$$

where $\delta(\cdot)$ is the delta function

$$\delta(x) = \begin{cases} 0, & \text{if } x \le 0 \\ 1, & \text{if } x > 0. \end{cases} \tag{4}$$

With this binary term, the mask confidence is updated to

$$\mathcal{C}_i^* = \mathcal{P}_i \cdot \sum_{j=1}^{n} \Delta_{ij} \cdot \mathcal{P}_j \cdot m_{ij}. \tag{5}$$

$\mathcal{C}_i^*$ sums up the similarity elements m_{ij} whose values are higher than a threshold $\beta \in [0, 1]$ for each local mask. When β is close to 0, $\Delta_{ij} = 1$ for most cases and thus $\mathcal{C}_i^*$ degenerates to $\mathcal{C}_i$. When β is close to 1, Δ_{ij} and $\mathcal{C}_i^*$ are both close to 0, which means that the confidence values of all local masks to be selected as global masks are close to 0. In such a case, the clustering cannot distinguish global masks from local masks, resulting in false clustering in the end. In this work, we set $\beta = 0.5$ in all our experiments, indicating that local masks with similarities (i.e., the ratio of the overlapping area) higher than β are considered in the computation of mask confidence values. More details about the evaluation of the parameter β can be found in the implementation details in Section IV-C.

3) Mask Clustering: One key observation of this work is that local masks with higher confidence values are consistent with other masks and thus should have higher priority to be selected as global masks, as shown in Fig. 4.

Based on the mask confidence, we employ a simple yet efficient order-based mask clustering. We first sort all local masks according to their confidence values $\mathcal{C}^*$ and then traverse them in descending order to select global masks. In the traversing loop, if a local mask has not been marked, we mark it as a new global mask, and other nonmarked local masks whose similarities with this global mask are higher than β are considered consistent with this global mask, that is,

$\delta(m_{lg} - \beta) = 1$, where l and g are the indices of the local mask and this global mask, respectively. If a local mask has been already marked, we traverse to the next local mask until all of them are marked.

With the precomputation of mask confidence values, the traversal is required only once. For efficiency, we establish a mapping table $\mathbb{M}$ between all local masks to their corresponding global masks. It is worth noting that even though we cannot guarantee each global mask in $\mathbb{M}$ corresponds to a building instance in the real scene at this stage, a few false correspondences will not affect the final 3-D instance segmentation. This will be explained in Section III-C1.

C. 3-D Building Instance Segmentation

1) 3-D Roof Instance Segmentation: The existing multiview semantic segmentation framework projects the 2-D semantic labels back to the 3-D model with the highest probability [48], [49]. For 3-D instance segmentation, this framework is not suitable because the correspondences between local masks from multiview images are unknown. Our mask clustering establishes the correspondences between the local masks from multiple views. We first project each vertex of the 3-D model to all images and check its visibility using fields of view and depth maps. Then, using its corresponding local mask index at its projected position in the image, we retrieve its corresponding global mask from the mask mapping table $\mathbb{M}$.

For a vertex v, let MVI_v denote the set of multiview image index in which v is visible and GMI_p denote the global mask index corresponding to a multiview image p. In some cases, a vertex v on the 3-D surface model may be projected within multiple global masks. We denote the set of these global masks as $\{\text{GMI}_p | p \in \text{MVI}_v\}$ and thus $\text{GMI}_p = -1$ represents the background. From these global masks, the one with the largest quantity of corresponding local masks that were projected to by this vertex is associated with this vertex

$$\text{RID}_v = \text{maxCount}(\{\text{GMI}_p | p \in \text{MVI}_v\}) \tag{6}$$

where roof index (RID_v) is the roof IDentity (ID) of vertex v, and the function maxCount(S) extracts the most occurring element in the set S. In case more than one global masks have the maximum count in the set S, the global mask with a smaller value of GMI will be chosen, as a smaller GMI value corresponds to a higher confidence of the global mask.

The advantage of determining the roof IDs in this way is that the most confident global mask can be automatically selected, and thus a user does not have to provide a specific number of target clusters (i.e., the number of global masks). This is because local masks with large errors in 2-D roof segmentation are normally divided into groups containing small numbers of local masks. The global masks derived from these local masks usually have wrong predictions and therefore will be ignored in the upcoming 3-D roof segmentation step, since only the global mask with the largest quantity of corresponding local masks is selected. Therefore, our 3-D roof instance segmentation achieves higher prediction precision than the roof instance segmentation on the multiview images. Their AP/AP50/AP75 values can be found in Section III-A and Table II.

With the roof IDs for all vertices, the segmentation of the 3-D model can be easily obtained. Specifically, the roof ID of a triangle face is determined by the majority of the vertices that indicate the same roof ID, that is,

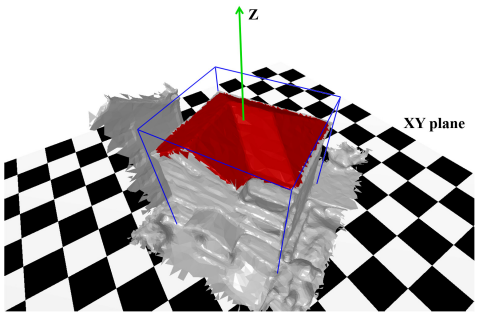


Fig. 5. HOBB. The top and bottom faces of the HOBB are horizontal, and the other four faces are oriented according to the principal component analysis (PCA) orientation estimation.

$$\text{RID}_t = \text{maxCount}(\{\text{RID}_v | v \in V_t\}) \tag{7}$$

where RID_t represents the roof ID of triangle t, and V_t represents the three vertices of the triangle t.

2) MRF-Based 3-D Building Segmentation: Based on 3-D roof segmentation, the next step is to segment the entire 3-D buildings. We first estimate a horizontally oriented bounding box (HOBB) for each roof instance, as shown in Fig. 5. To segment an entire building from a 3-D scene, we expand the HOBB on its four sides by a certain offset value (4 m in all of our experiments). The triangles within the expanded HOBB (excluding those that have been labeled by other roof instances) are selected as a candidate building. We denote its triangle set as $T = \{t_i\}$ and its edge set as $E = \{e_{ij}\}$, in which t_i represents the ith triangle in T, and e_{ij} represents the edge connecting t_i and t_j. We formulate the building segmentation as a foreground/background labeling process that minimizes the following energy function

$$\psi(l) = \sum_{t_i \in T} \psi_{\text{data}}(l_i) + \sum_{e_{ij} \in E} \psi_{\text{smooth}}(l_i, l_j) \tag{8}$$

where l_i denotes the label of triangle t_i given by MRF-based segmentation. $l_i = 1$ indicates the foreground (i.e., a building triangle) and $l_i = 0$ the background (i.e., a nonbuilding triangle).

The data term $\psi_{\text{data}}(l_i)$ represents the penalty of assigning a label l_i to a triangle t_i. The 3-D roof segmentation provides us a good foreground initialization, we denote its triangle set as T_f. We take triangles on the boundary of T as a background initialization, denoted as T_b. We further denote the set of other triangles in T as T_r. As shown in Fig. 7(a), T_f, T_b, and T_r are visualized in red, green, and gray, respectively.

For triangles in T_r, our data term $\psi_{\text{data}}(l_i)$ is defined as

$$\psi_{\text{data}}(l_i) = \begin{cases} (1 + d_i) + \theta_i \cdot (1 + d_i), & \text{if } l_i = 1 \\ 1, & \text{if } l_i = 0. \end{cases} \tag{9}$$

Before explaining the meaning of d_i and θ_i, we define $\mathcal{P}$, a 2-D polygon representing the simplified roof boundary edges, as shown in Fig. 6. We first extract boundary edges of the roof triangles using the Alpha Shape algorithm [50]. Then, we reduce the dimension of these boundary edges to the 2-D horizontal plane by discarding their height and simplifying them through a random sample consensus

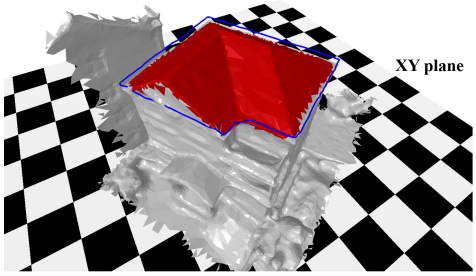


Fig. 6. Simplified roof boundary edge $\mathcal{P}$. The blue polygon represents the flat boundary of the roof on the XY-plane. To make it easy to observe, we set its Z-coordinate to be close to the height of the roof.

(RANSAC) process [51]. The RANSAC process iteratively merges outline points of roofs to their neighbors if they are approximately collinear. The process stops until no more points can be merged. The simplified roof boundary edges constitute a 2-D polygon that is referred to as the roof profile, denoted as $\mathcal{P}$. For each triangle $t_i \in T_r$, we find a line segment in $\mathcal{P}$ with a minimal distance to the center of t_i. We denote this minimal distance as d_i, and the cosine of the horizontal angle between the normal of t_i and the direction of this line segment as θ_i. $\theta_i = 0$ when the normal of t_i is perpendicular to this line segment, and $\theta_i = 1$ when their horizontal angle is 0. Then d_i is normalized by the maximum distance in the HOBB, and we sign the distance as negative if t_i is inside of $\mathcal{P}$. To make the foreground penalty and background penalty comparable on the roof profile (i.e., both penalties equal to 1 when $d_i = 0$ and $\theta_i = 0$), we use $1 + d_i$ instead of d_i.

In (9), the $(1 + d_i)$ term is a distance constraint that guarantees only triangles close to $\mathcal{P}$ are taken into the building. The $\theta_i \cdot (1+d_i)$ term is an orientation constraint that guarantees that only triangles having similar orientations with the closest line segment are taken. The $(1+d_i)$ multiplication is to reduce the orientation constraint if the triangles are far away from $\mathcal{P}$.

The smoothness term $\psi_{\text{smooth}}(l_i, l_j)$ penalizes adjacent triangles t_i and t_j being assigned with different labels. We take the cosine angle between normals of adjacent triangles as the penalty for assigning different labels to the adjacent triangle pair, that is,

$$\psi_{\text{smooth}}(l_i, l_j) = \begin{cases} ll||n_i \cdot n_j||, & \text{if } l_i \neq l_j \\ 0, & \text{if } l_i = l_j. \end{cases} \tag{10}$$

Using the angles between faces, $\psi_{\text{smooth}}(l_i, l_j)$ favors segmentation at sharp edges rather than at planar regions.

D. Benchmark Dataset

We have created a benchmark dataset *InstanceBuilding* that contains annotation for both UAV images and 3-D urban scenes simultaneously. To evaluate our 3-D instance segmentation method, we annotated 3-D roofs and buildings for four large 3-D urban scenes, which are reconstructed using *Bentley Acute3D ContextCapture*[2] from UAV images. Table I shows detailed information about these scenes, and their visualization can be found in Fig. 8 and the supplementary video. Note that the town is quite crowded, thus about two-thirds of the buildings are attached to others, as shown in the last column of Table I. To facilitate the 3-D annotation, we have developed a simple but efficient brush-based annotation tool. Similar to most 2-D annotation tools [52] which semiautomatically extract pixels of an object by marking the closed boundary polygon of the object, our tool allows a user to segment a 3-D building by casually drawing strokes on the building boundaries.

[2]https://www.bentley.com/en/products/brands/contextcapture

TABLE I
STATISTICS ON THE 3-D MODELS OF OUR *InstanceBuilding* DATASET

Scene	#Vertices	#Triangles	Area (km^2)	#Images (resolution)	#Buildings All / Attached
#1	1.13M	2.26M	0.076	79 (5472×3648)	185 / 145
#2	0.87M	1.72M	0.097	64 (5472×3078)	119 / 40
#3	1.14M	2.28M	0.081	284 (1916×994)	322 / 232
#4	0.60M	1.20M	0.18	240 (1536×994)	266 / 185

Our *InstanceBuilding* dataset also contains 608 annotated images with high resolutions. They are selected from around 20 000 images acquired in more than ten different cities. Some are directly captured by a consumer DaJiang (DJI) drone Phantom 4 Pro with different cameras and flight altitudes, others are rendered by 3-D models with textures as orthophotos with similar resolutions. There are about 16 000 buildings in all these images, and their roofs are all manually annotated for the training of our 2-D roof instance segmentation neural network. These annotated images are divided into two groups, 524 images for training and 84 images for validation.

For accuracy and efficiency consideration, we modified the LabelMe [52] toolkit to visualize the corresponding heightmap window alongside the color image window. By synchronizing the annotation on both the image window and the heightmap window, volunteers can freely annotate on either of them. Based on our time recording of volunteer annotation, this double-window strategy saves the volunteers more than half of the time.

Our *InstanceBuilding* dataset contains building instance annotation for both 3-D urban scenes and UAV images simultaneously, which makes it unique. Most of existing 3-D datasets are designed for semantic segmentation, such as Vaihingen3D [53], Swiss3DCities [54], Hessigheim3D [55], and Semantic Urban Meshes (SUM) [56]. The most related work about 3-D instance segmentation dataset is the Urban drone dataset (UDD) [57] and UrbanScene3D [58]. UDD evaluates their 3-D segmentation accuracy by projecting them to drone images, thus cannot be regarded as a 3-D dataset. UrbanScene3D does not contain corresponding UAV images, has only 485 annotated buildings, and very few buildings are attached to others. As shown in Table I, our *InstanceBuilding* dataset has 892 annotated buildings, of which 602 are attached to others. In such crowded urban scenes, instance segmentation has a more significant advantage over semantic segmentation.

Compared to existing natural image datasets, such as Microsoft Common Objects in Context (MSCOCO) [47], Cityscapes [59], and Pattern Analysis, Statistical Modeling and Computational Learning (PASCAL) Visual Object Classes (VOC) [60], our annotation on UAV images focuses on roof instances for UAV images. Compared to other remote-sensing imagery datasets, such as SpaceNet [61] and xView [62], our UAV images have higher resolution and more viewing angles, thus are an important supplement to existing datasets.

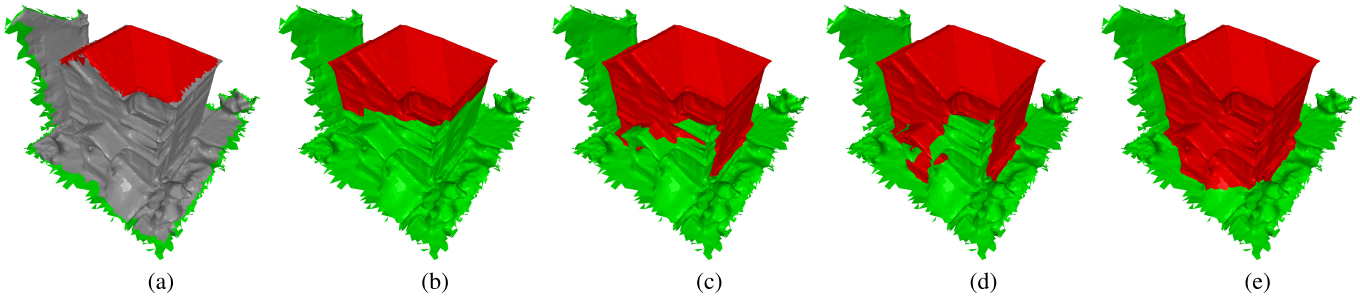


Fig. 7. 3-D building segmentation based on MRF optimization. (a) Initial segmentation by the projection of the 2-D roof onto the 3-D model (foreground in red and background in green). (b) Result of direct MRF optimization, where the initial roof segmentation misses the balcony and the gate shelter. (c) Our segmentation result without the orientation constraint. (d) Our segmentation result without the distance constraints. (e) Our segmentation result using both orientation and distance constraints.

Though there are also many drone datasets in public, such as VisDrone [63] and dataset for object detection in aerial images (DOTA) [64], few of them focus on instance segmentation at the pixel level. With the increasing popularity of low-altitude UAV capturing devices, we believe that our dataset will play an important role in applications such as urban planning, smart cities, and other related fields.

E. Implementation Details

1) 2-D Roof Segmentation: We use the Pytorch implementation of Swin Transformer released by [22] for roof instance segmentation from images. The manually labeled images (see Section III-D) were used to fine-tune the Swin Transformer network trained previously using the Common Objects in Context (COCO) dataset [47]. For 3-D urban scenes that have corresponding UAV images, we directly segment these images. For 3-D urban scenes that do not have corresponding UAV images, multiview images are rendered using these 3-D scenes from a set of different viewpoints sampled randomly at 70 m higher than the average height of these scenes. At each viewpoint, five virtual cameras facing down, front, back, left, and right are placed.

2) MRF-Based Segmentation: With the aforementioned MRF setting, we treat (8) as a classical max-flow/min-cut optimization and solve it using the graph cut algorithm [65]. The MRF-based segmentation result is shown in Fig. 7(e). To prove the validity of (9), we simply set the penalties to 1 for both foreground and background in (9), as the triangles in T_r have no explicit foreground/background priorities. As a result, it does not produce correct segmentation as expected, as shown in Fig. 7 (b). Comparisons in Fig. 7(c) and (d) show that the introduction of the orientation and the distance constraint overcomes the interference of structural variations and noises in the MRF optimization, thus improving the 3-D building segmentation.

IV. Results and Discussion

We have evaluated our method with both drone images and virtually rendered images. All experiments were carried out on a machine with an Intel Core i7 processor, 32 GB memory, and an NVidia GeForce 1080 GPU.

A. Qualitative Results

We tested our method on four scenes (see Fig. 8), for which the statistics are shown in Table I. Scene #1 and Scene #2 have UAV images and the corresponding camera parameters. Scene #3 and Scene #4 do not have UAV images, and we rendered images from multiple viewpoints instead, as explained in Section III-E. The roof instance segmentation results are shown in Fig. 8 (middle column) and the 3-D building instance segmentation results are shown in Fig. 8 (right column). We can see that our approach successfully segmented most buildings, even if they are dense, varying in style, and attached.

B. Ablation Analysis

We have evaluated our results on the 3-D mesh models in the *InstanceBuilding* benchmark dataset. We first computed the *IoUs* between predictions and the ground truth based on the area of the mesh triangles. Then we used the commonly used instance-level evaluation metrics, namely AP, AP50, and AP75, in all evaluations [47], where AP50/AP75 indicates the AP when *IoU* threshold is set to 0.5/0.75, and AP is averaged over ten *IoU* thresholds of 0.5:0.05:0.95.

1) Height Information: As demonstrated in Fig. 3, using RGBH images significantly improves the 2-D roof instance segmentation. To understand how the height information improves 3-D roof instance segmentation and its effects on 3-D building instance segmentation, we have conducted a comparison on the four large 3-D scenes from *InstanceBuilding* both with and without height information, using two different clustering methods, that is, the spectral clustering and our method. The results of the comparison are given in Tables II and III. These comparisons have revealed that our method using RGBH images achieves higher accuracy than the one without height information (i.e., using RGB images). This is because the additional heightmaps provide spatial information of the urban scenes to the neural network model, which makes the roofs and buildings more distinguishable.

2) Multiview Framework: The 2-D segmentation in our method is applied to multiview images, rather than orthophoto maps. To understand the advantages of this multiview framework, we have also implemented an orthophoto-based instance segmentation solution for comparison. We first generated the orthophoto maps and their corresponding heightmaps of the same scenes through a render-to-texture technique. These two types of maps were combined to form the orthophoto RGBH maps, which were then used to detect the 2-D roof instances. Since there was almost no occlusion for the roofs in the orthophoto maps, mask clustering was not necessary and we

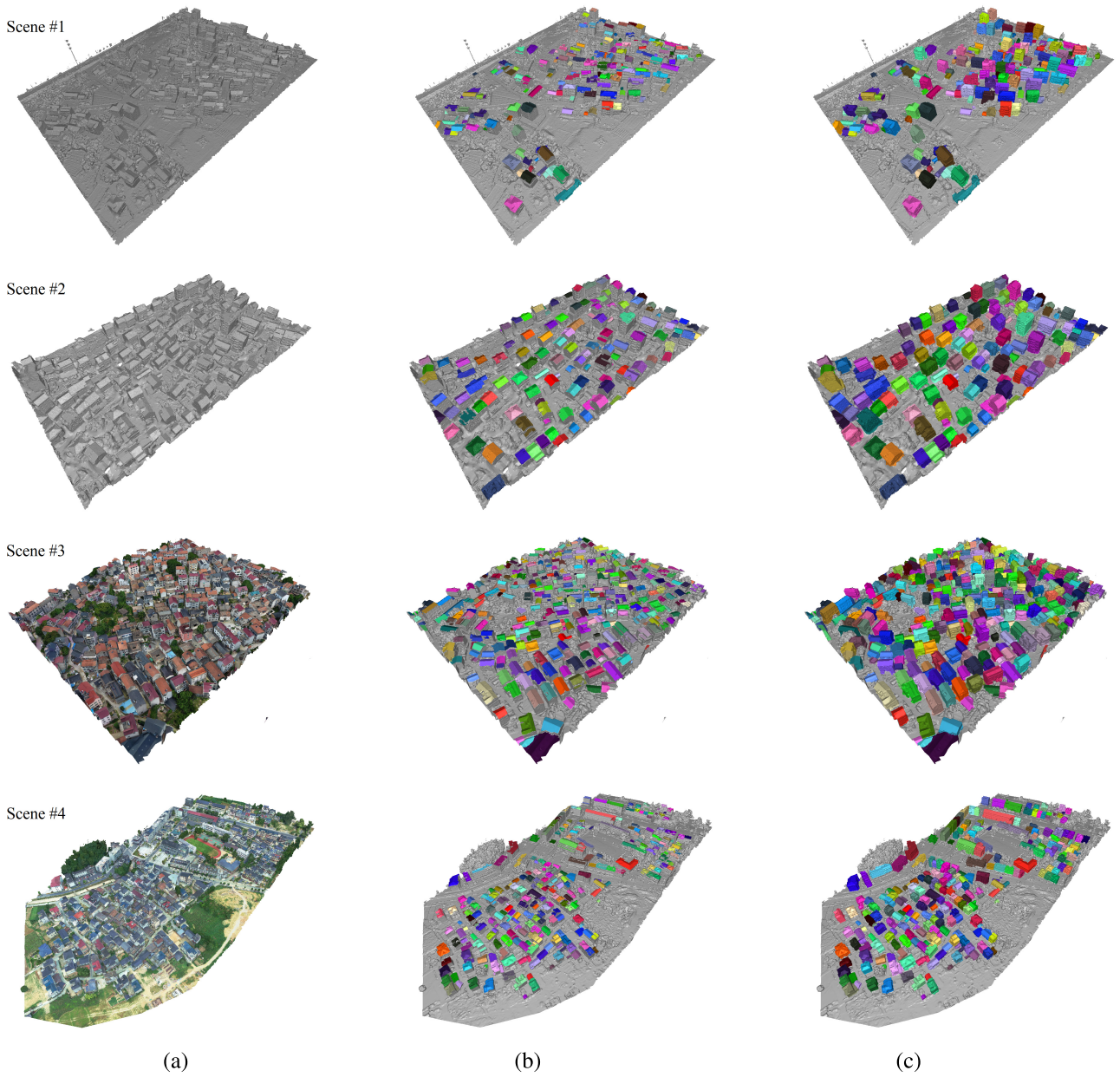


Fig. 8. Building instance segmentation results of four scenes. The two scenes on the top have UAV images with camera parameters, while the scenes #3 and #4 in the bottom do not have UAV images, for which we render images from multiple viewpoints instead. (a) Initial scene models. (b) 3-D Roof segmentation results. (c) 3-D Building segmentation results.

TABLE II

COMPARISON OF FOUR DIFFERENT CLUSTERING METHODS ON 3-D *Roof* INSTANCE SEGMENTATION USING SWIN TRANSFORMER

Scene	Spectral with RGB (K_1)			Spectral with RGBH (K_1)			Spectral with RGB (K_2)			Spectral with RGBH (K_2)			Ours with RGB			Ours with RGBH		
	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75
#1	0.5858	0.7841	0.5966	0.6551	0.8239	0.7045	0.5989	0.7784	0.6364	0.6665	0.85227	0.7159	0.6455	0.8409	0.6818	**0.7114**	**0.8693**	**0.7727**
#2	0.5329	0.6973	0.5502	0.4579	0.6446	0.4463	0.5436	0.7021	0.5633	0.5826	0.8347	0.5785	0.5651	0.7190	0.5880	**0.6610**	**0.8395**	**0.6707**
#3	0.5778	0.7663	0.6095	0.5501	0.7284	0.6068	0.5900	0.8162	0.6482	0.5966	0.8398	0.6396	0.6179	0.8476	0.6604	**0.6434**	**0.9072**	**0.6618**
#4	0.4447	0.7444	0.4398	0.4526	0.7256	0.4549	0.4583	0.8008	0.4323	0.4635	0.7820	0.4474	0.5011	0.8383	0.4962	**0.5248**	**0.8459**	**0.5301**

directly applied the 3-D building segmentation to produce the final results. The statistics of the results are reported in Table IV.

From this comparison, we can see that our multiview method achieves higher AP than the orthophoto-based method. There are three main reasons for this improvement. The first advantage of using multiview images lies in the mask clustering that integrates multiple segmentation results of the identical roofs from different viewpoints. This mask clustering process can be regarded as a cross-correction process, thus

TABLE III
COMPARISON OF FOUR DIFFERENT CLUSTERING METHODS ON 3-D *Building* INSTANCE SEGMENTATION USING SWIN TRANSFORMER

Scene	Spectral with RGB (K_1)			Spectral with RGBH (K_1)			Spectral with RGB (K_2)			Spectral with RGBH (K_2)			Ours with RGB			Ours with RGBH		
	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75
#1	0.5881	0.7784	0.6054	0.6362	0.8162	0.6973	0.6184	0.8216	0.6270	0.6843	0.8703	0.7351	0.6486	0.8324	0.6649	**0.7130**	**0.8919**	**0.7730**
#2	0.5311	0.7059	0.5210	0.4739	0.6302	0.4705	0.5615	0.7699	0.5666	0.5798	0.8235	0.5798	0.5709	0.7479	0.5966	**0.6655**	**0.8403**	**0.6807**
#3	0.4373	0.6429	0.4658	0.5369	0.7764	0.5745	0.5469	0.8168	0.5590	0.5730	0.8354	0.5994	0.6357	0.8882	0.6925	**0.6671**	**0.9286**	**0.7174**
#4	0.5739	0.7632	0.6128	0.5752	0.7481	0.6090	0.6192	0.8346	0.6504	0.6248	0.8459	0.6429	0.6534	0.8571	0.6992	**0.6726**	**0.8759**	**0.7105**

eliminating most of the incorrect instance masks from individual multiview images. Second, it is very difficult to separate two roofs if they are attached and have similar textures using orthophoto maps. In contrast, our multiview framework has much more information from building walls, thus achieving more accurate segmentation. Finally, the original drone images usually have high resolution without texture distortion, but orthograph maps may have these drawbacks due to the texture mapping and synthesis on the 3-D meshes.

3) Mask Clustering: The instance mask clustering is crucial to overcome the ambiguities in the multiview instance masks. Many clustering methods are available in the literature, such as Mean-shift [66] and K-means [67], but none of them is suitable for our multiview scenario. These clustering methods require computing the mean of features, which heavily relies on a good feature extractor. For example, the position, size, and mean color of masks cannot accurately describe their features. The other difficulty of these clustering methods is choosing the optimal values of parameters, such as the kernel function for Mean-shift, and the K value for K-means. Note that our clustering method does not require specifying the number of target clusters, which is the core advantage of our method.

What is more important is that it is still unknown how to design comparable features for masks from different views, especially when they have a large variation among different viewpoints. Compared with viewpoint-variant features, the spatial overlap between masks can be calculated accurately. Therefore, we directly calculate a similarity matrix using the spatial overlap between masks, rather than their features.

Based on the similarity matrix, one alternative option for mask clustering is spectral clustering [68]. We have evaluated it with different numbers of target clusters, noted as K, and compared our clustering method with it. For a fair comparison, we used two different values of K for the spectral clustering method on each scene: 1) K_1: the building number of the annotated 3-D scene and 2) K_2: the number of global masks estimated by our clustering method. Note that K_2 is typically larger than K_1, as some of the global masks did not correspond to real 3-D roofs. A more detailed explanation is given in Section III-C. Specifically, K_1 has a value of $185/185, 119/119, 322/322, 266/266$, and K_2 has a value of $385/475, 202/376, 872/714, 646/641$ for Scene #1, #2, #3, and #4 with RGB/RGBH images, respectively.

The advantages of our clustering method over spectral clustering can also be concluded from Tables II and III. Regardless of the 3-D roof or building instance segmentation, using K_1 or K_2, with or without height information, our clustering method always reaches higher precision than spectral clustering. Visual comparison for Scene #1 is shown in Fig. 9. From this comparison, we can observe that spectral clustering has the issue of under-segmentation when $K = K_1$. Since roof instance masks detected from the multiview images are much more than the ground truth and have prediction errors as well, the spectral clustering tends to incorrectly mix some roofs of attached buildings when $K = K_1$. Meanwhile, the spectral clustering has the issue of over-segmentation when $K = K_2$. Since K_2 is larger than the ground-truth roofs in the 3-D scene, it lost the ability to separate the correct and incorrect masks, leaving these masks separated. In contrast, our instance mask clustering successfully segments roof instances precisely without specifying the number of target roofs. More visual comparison results can be found in the supplementary video.

TABLE IV
COMPARISON BETWEEN OUR FRAMEWORK BASED ON MULTIVIEW IMAGES AND THE ONE BASED ON ORTHOPHOTO MAPS USING SWIN TRANSFORMER

Scene	Orthophoto-based			Ours (multi-view)		
	AP	AP50	AP75	AP	AP50	AP75
#1	0.6389	0.8432	0.6541	**0.7130**	**0.8919**	**0.7730**
#2	0.5975	0.7906	0.6059	**0.6655**	**0.8403**	**0.6807**
#3	0.5935	0.9006	0.6429	**0.6671**	**0.9286**	**0.7174**
#4	0.6199	0.8195	0.6391	**0.6726**	**0.8759**	**0.7105**

TABLE V
COMPARISON BETWEEN OUR FRAMEWORK BASED ON MULTIVIEW IMAGES AND THE ONE BASED ON ORTHOPHOTO MAPS USING MASK R-CNN

Scene	Orthophoto-based			Ours (multi-view)		
	AP	AP50	AP75	AP	AP50	AP75
#1	0.6751	**0.8595**	0.7189	**0.7270**	**0.8595**	**0.7730**
#2	0.5969	0.7176	0.5847	**0.6202**	**0.7227**	**0.6471**
#3	0.5475	0.8199	0.5714	**0.6270**	**0.8354**	**0.7019**
#4	0.6079	**0.7895**	0.6278	**0.6117**	**0.7895**	**0.6353**

It is worth pointing out that the last block named "Ours with RGBH" of Table II also shows that we achieve higher AP/AP50/AP75 on the 3-D roof instance segmentation than on the aerial images (values are shown in Section III-A), because our occlusion-aware mask clustering suppresses false prediction from individual images and thus improves the overall segmentation precision.

4) Alternatives for Image Instance Segmentation: The core contribution of our work is the multiview framework with a new instance mask clustering, not the image instance segmentation. Besides Swin Transformer, our framework can incorporate any other image instance segmentation method as well. To demonstrate its compatibility, we have testified it with Mask R-CNN [18], which is another widely used image instance

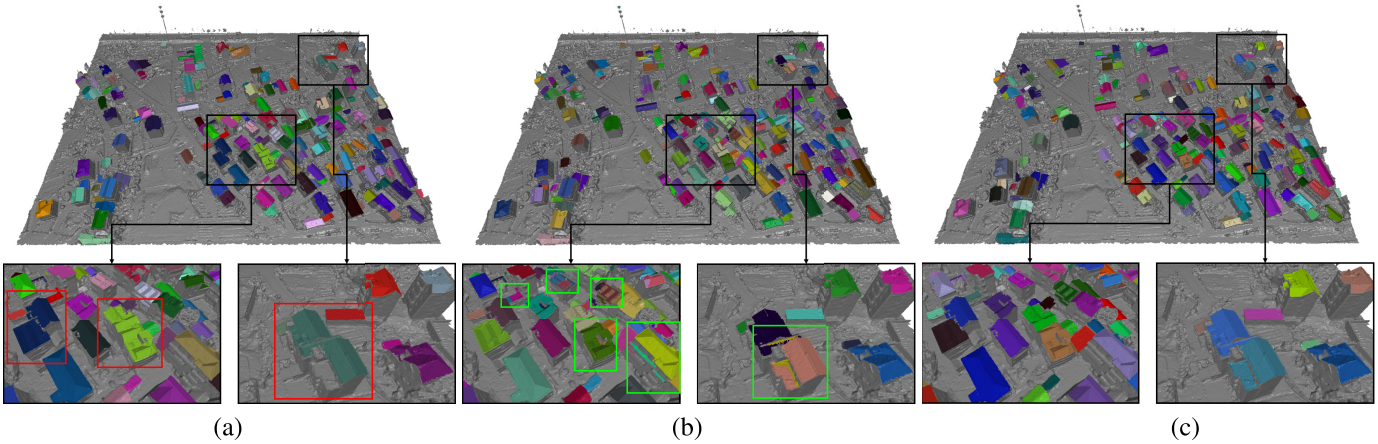


Fig. 9. Comparison of roof mask clustering of three methods on Scene #1. (a) Spectral clustering using the ground-truth number of roof targets K_1 tends to under-segment (red rectangles) some roofs. (b) Spectral clustering using the number of global masks K_2 estimated by our method tends to over-segment (green rectangles) some roofs. (c) Our clustering method achieves more precise roof instance segmentation without specifying the target number.

TABLE VI
COMPARISON OF USING DIFFERENT SEGMENTATION CONSTRAINTS ON 3-D *Building* INSTANCE SEGMENTATION USING SWIN TRANSFORMER

Scene	Without the orientation constraint			Without the distance constraint			With both constraints		
	AP	AP50	AP75	AP	AP50	AP75	AP	AP50	AP75
#1	0.6438	0.8541	0.7189	0.6856	0.8673	0.7537	**0.7130**	**0.8919**	**0.7730**
#2	0.5605	0.8235	0.5966	0.6378	0.8325	0.6722	**0.6655**	**0.8403**	**0.6807**
#3	0.4252	0.8540	0.3882	0.6177	0.9224	0.6770	**0.6671**	**0.9286**	**0.7174**
#4	0.5289	0.8308	0.5301	0.6176	0.8722	0.6466	**0.6726**	**0.8759**	**0.7105**

segmentation model. Table V shows a comparison between our framework based on multiview images and the one based on orthophoto maps using Mask R-CNN. Similar to using Swin Transformer, the advantages of our multiview method over the orthophoto-based method can also be found in this table. This demonstrates the effectiveness of our multiview framework regardless of the chosen image instance segmentation method. Comparisons with the spectral clustering method and different MRF segmentation constraints using Mask R-CNN are shown in the supplementary document. For all these results, our method consistently achieves the highest accuracies. It is worth noting that most of the evaluation results for Mask R-CNN are lower than those of the recently developed Swin Transformer.

5) MRF-Based Building Segmentation: As shown in Fig. 7, the orientation and distance constraints are important to achieve accurate 3-D building segmentation. We have also evaluated the segmentation precision and recall by omitting one of them. The results are reported in Table VI, from which we can conclude that the distance constraint plays a more important role than the orientation constraint, but the best segmentation accuracy can only be achieved if they are both employed.

C. Effects of Parameters

Our method involves a few parameters, among which β in the mask clustering step is the only parameter that we leave tunable for users (while all other parameters are fixed). In this section, we discuss how this parameter affects mask clustering.

Intuitively, the meaning of the β parameter in our work is very similar to the threshold parameter of *IoU* in many existing object detection works where a mask is considered to be correctly predicted when the *IoU* between the detection mask and the ground truth is greater than this threshold. Empirically, this threshold is initially set to 0.5. Similarly, in our mask clustering, if the *IoU* of the two local masks is greater than β, they should be considered to belong to the same group. That is, they represent the same roof instance. In this work, we initially set $\beta = 0.5$ in all of our experiments. To determine the optimal value for β, we experimented with different values in all four scenes. As we can see from Fig. 10, AP, AP50, and AP75 are always close to the highest values when $\beta = 0.5$. Note that slightly increasing/reducing β reduces/increases the confidence values but barely affects the ordering of the masks. This reveals that our mask clustering is tolerant to the β parameter.

D. Running Time

The training took around 83 h with 2000 epochs. The overall running time for segmenting a scene varied from 6 to 8 min, depending on the scene size, image resolution, and the number of images. The MRF optimization takes around 2.5 min on average for each scene. The computational complexity of multiview image generation (only for scenes without drone images), heightmap generation, 3-D vertex projection for the overlapping computation, and back-projection for the clustered instance masks are $O(N * k)$, where N is the number of faces in this 3-D urban model, and k is the number of multiview images. N could be a large number, but these computations are fully accelerated using the GPU, only taking less than 1 min for each urban scene. The instance mask clustering has

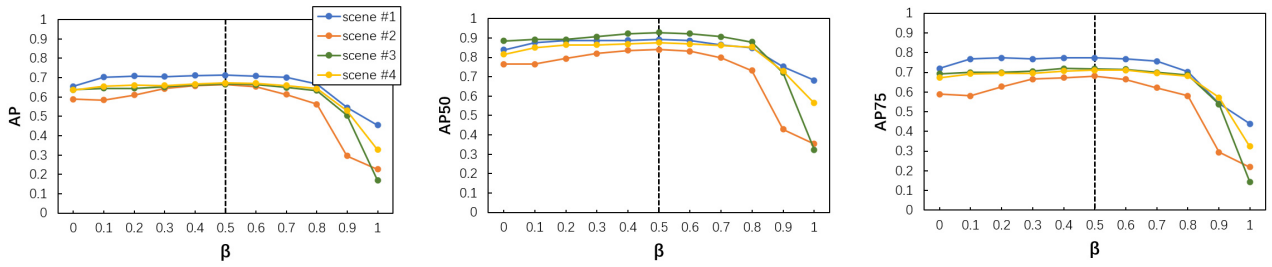


Fig. 10. Evaluation of the segmentation results with different β values using Swin Transformer. AP (left), AP50 (middle), and AP75 (right) of the segmentation result are always close to the highest values when $\beta = 0.5$.

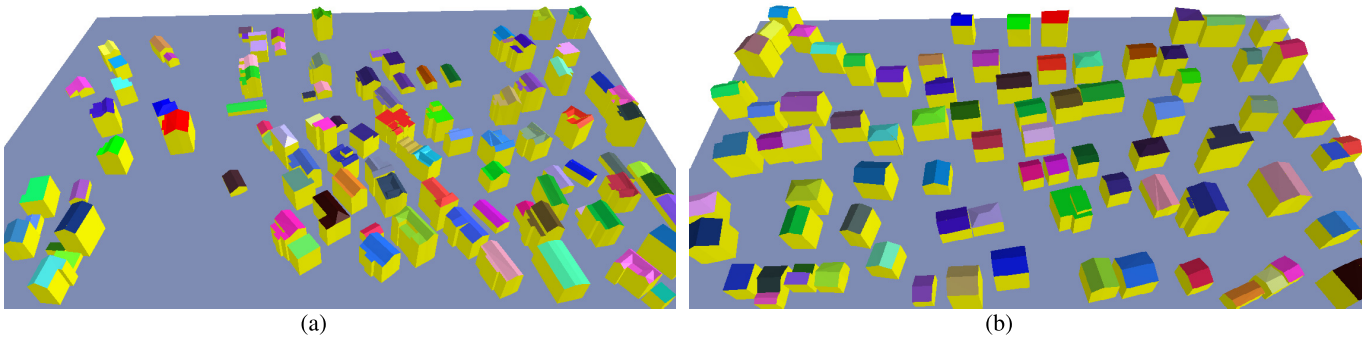


Fig. 11. Automatic simplification of buildings in two large urban scenes. The results are obtained by applying RANSAC for plane extraction followed by plane regularization. (a) Scene #1. (b) Scene #2.

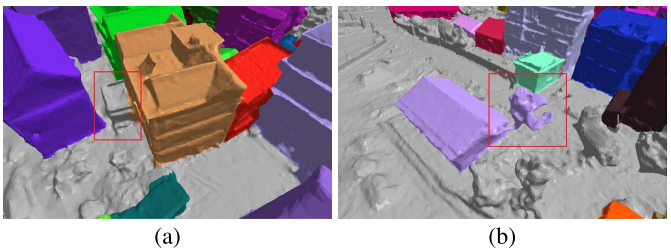


Fig. 12. Two failure cases of our method. (a) Gated shelter was not segmented as part of the roof in the image segmentation stage. (b) Tall tree was segmented as a roof instance because it has a similar height and texture as its nearby building.

a computational complexity of $O(n)$, where n is the number of instance masks, thus it takes less than 1 s in total for each urban scene. It is much faster than many other traditional clustering methods.

E. Discussions

1) Applications: We have implemented a simple building simplification prototype using a RANSAC-based plane fitting [51] and plane regularization. Based on our 3-D building instance segmentation, 3-D buildings in large urban scenes are automatically simplified, as shown in Fig. 11. In such an application, instance segmentation is obligatory, as semantic segmentation is insufficient to separate individual buildings. We believe our 3-D building instance segmentation method can benefit more smart city applications, such as urban planning.

2) Limitations: Since our approach falls into the multiview paradigm, it relies on the quality of the 2-D instance segmentation. Roof types that do not exist in the training dataset may not be precisely segmented. Two such examples are shown in Fig. 12. In Fig. 12(a), a gated shelter was not reliably detected, and in Fig. 12(b), a tall tree was segmented as a part of the nearby building. Enriching the training dataset may partially solve this issue. In addition, developing a neural network dedicated to separating roofs and trees may produce more reliable instance segmentation results.

V. Conclusion and Future Work

We have presented a multiview framework for instance segmentation of 3-D urban buildings. Based on occlusion-aware similarity matrices, a novel instance mask clustering method is proposed to eliminate the mask ambiguities among multiview images. To further improve segmentation accuracy, roofs (instead of buildings) are first segmented, and RGB images are enriched with heightmaps. Our method takes full advantage of the multiview framework to precisely segment 3-D buildings in large urban scenes.

We have collected and annotated an RGBH drone imagery dataset and a 3-D building instance segmentation dataset, named *InstanceBuilding*. We believe that the new dataset could benefit research in 3-D instance segmentation for various urban applications. Since most of the state-of-the-art learning-based 3-D instance segmentation methods focus on indoor scenes, our multiview instance segmentation framework explores a new avenue for large outdoor scenes.

Future directions: Our work focuses on buildings because they are the most important ingredients in the urban environment. One future direction is to extend our multiview 3-D instance segmentation framework to other urban objects and even indoor scenes. Drone images and the reconstructed

3-D models may suffer from various degradation (such as noises), so it is worth investigating the robustness of our method in such degraded scenarios [69]. Finally, applying our method to 3-D point clouds of urban scenes could be an interesting future direction as well.

ACKNOWLEDGMENT

The authors would like to thank Quzhou Southeast Flysee Technology Ltd., for providing drone images, 3-D urban models, and parts of 2-D annotations.

REFERENCES

[1] H.-H. Vu, P. Labatut, J.-P. Pons, and R. Keriven, "High accuracy and visibility-consistent dense multiview stereo," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 889–901, May 2012.

[2] Y. Yao, Z. Luo, S. Li, T. Fang, and L. Quan, "MVSNet: Depth inference for unstructured multi-view stereo," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 785–801.

[3] G. Riegler, A. O. Ulusoy, and A. Geiger, "OctNet: Learning deep 3D representations at high resolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 3577–3586.

[4] D. Maturana and S. Scherer, "VoxNet: A 3D convolutional neural network for real-time object recognition," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2015, pp. 922–928.

[5] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: Deep learning on point sets for 3D classification and segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 652–660.

[6] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 5099–5108.

[7] L. Yi, W. Zhao, H. Wang, M. Sung, and L. J. Guibas, "GSPN: Generative shape proposal network for 3D instance segmentation in point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3947–3956.

[8] W. Wang, R. Yu, Q. Huang, and U. Neumann, "SGPN: Similarity group proposal network for 3D point cloud instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 2569–2578.

[9] J. Hou, A. Dai, and M. Nießner, "3D-SIS: 3D semantic instance segmentation of RGB-D scans," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4421–4430.

[10] X. Wang, S. Liu, X. Shen, C. Shen, and J. Jia, "Associatively segmenting instances and semantics in point clouds," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4096–4105.

[11] Q.-H. Pham, T. Nguyen, B.-S. Hua, G. Roig, and S.-K. Yeung, "JSIS3D: Joint semantic-instance segmentation of 3D point clouds with multi-task pointwise networks and multi-value conditional random fields," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8827–8836.

[12] C. Liu and Y. Furukawa, "MASC: Multi-scale affinity with sparse convolution for 3D instance segmentation," 2019, *arXiv:1902.04478*.

[13] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 580–587.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2014.

[15] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[16] 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.*, 2015, pp. 91–99.

[17] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2117–2125.

[18] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[19] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang, "Mask scoring R-CNN," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 6409–6418.

[20] S. Liu, L. Qi, H. Qin, J. Shi, and J. Jia, "Path aggregation network for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 8759–8768.

[21] K. Chen *et al.*, "Hybrid task cascade for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 4974–4983.

[22] Z. Liu *et al.*, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 10012–10022.

[23] B. De Brabandere, D. Neven, and L. Van Gool, "Semantic instance segmentation with a discriminative loss function," 2017, *arXiv:1708.02551*.

[24] A. Fathi *et al.*, "Semantic instance segmentation via deep metric learning," 2017, *arXiv:1703.10277*.

[25] D. Novotny, S. Albanie, D. Larlus, and A. Vedaldi, "Semi-convolutional operators for instance segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 86–102.

[26] X. Liang, L. Lin, Y. Wei, X. Shen, J. Yang, and S. Yan, "Proposal-free network for instance-level object segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 2978–2991, Dec. 2018.

[27] S. Kong and C. Fowlkes, "Recurrent pixel embedding for instance grouping," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 9018–9028.

[28] M. Bai and R. Urtasun, "Deep watershed transform for instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5221–5229.

[29] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2359–2367.

[30] D. Bolya, C. Zhou, F. Xiao, and Y. J. Lee, "YOLACT: Real-time instance segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 9157–9166.

[31] H. Chen, K. Sun, Z. Tian, C. Shen, Y. Huang, and Y. Yan, "BlendMask: Top-down meets bottom-up for instance segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 8573–8581.

[32] J. A. Montoya-Zegarra, J. D. Wegner, L. Ladický, and K. Schindler, "Semantic segmentation of aerial images in urban areas with class-specific higher-order cliques," *ISPRS Ann. Photogram., Remote Sens. Spatial Inf. Sci.*, vol. 2, no. 3, pp. 127–133, 2015.

[33] Z. Li, J. D. Wegner, and A. Lucchi, "Topological map extraction from overhead images," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1715–1724.

[34] P. W. Battaglia *et al.*, "Relational inductive biases, deep learning, and graph networks," 2018, *arXiv:1806.01261*.

[35] F. Zhang, N. Nauata, and Y. Furukawa, "Conv-MPN: Convolutional message passing neural network for structured outdoor architecture reconstruction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2798–2807.

[36] 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.

[37] D. Hong *et al.*, "More diverse means better: Multimodal deep learning meets remote-sensing imagery classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 5, pp. 4340–4354, May 2021.

[38] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 7, pp. 5966–5978, Jul. 2021.

[39] D. Hong *et al.*, "SpectralFormer: Rethinking hyperspectral image classification with transformers," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, pp. 1–15, 2022.

[40] A. Dai, D. Ritchie, M. Bokeloh, S. Reed, J. Sturm, and M. Nießner, "ScanComplete: Large-scale scene completion and semantic segmentation for 3D scans," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 4578–4587.

[41] Z. Wu *et al.*, "3D ShapeNets: A deep representation for volumetric shapes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1912–1920.

[42] P.-S. Wang, Y. Liu, Y.-X. Guo, C.-Y. Sun, and X. Tong, "O-CNN: Octree-based convolutional neural networks for 3D shape analysis," *ACM Trans. Graph.*, vol. 36, no. 4, pp. 1–11, 2017.

[43] A. Dai, C. R. Qi, and M. Nießner, "Shape completion using 3D-encoder-predictor CNNs and shape synthesis," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5868–5877.

[44] Y. Li, R. Bu, M. Sun, W. Wu, X. Di, and B. Chen, "PointCNN: Convolution on X-transformed points," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 820–830.

[45] L. Zhao and W. Tao, "JSNet: Joint instance and semantic segmentation of 3d point clouds," in *Proc. AAAI*, 2020, pp. 12951–12958.
[46] M. Li, L. Nan, N. Smith, and P. Wonka, "Reconstructing building mass models from UAV images," *Comput. Graph. Forum*, vol. 54, pp. 84–93, Feb. 2016.
[47] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Berlin, Germany: Springer, 2014, pp. 740–755.
[48] A. Kundu, Y. Li, F. Dellaert, F. Li, and J. M. Rehg, "Joint semantic segmentation and 3D reconstruction from monocular video," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Berlin, Germany: Springer, 2014, pp. 703–718.
[49] M. Blaha, C. Vogel, A. Richard, J. D. Wegner, T. Pock, and K. Schindler, "Large-scale semantic 3D reconstruction: An adaptive multi-resolution model for multi-class volumetric labeling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3176–3184.
[50] F. Bernardini and C. L. Bajaj, "Sampling and reconstructing manifolds using alpha-shapes," in *Proc. 9th Can. Conf. Comput. Geometry*, 1997, pp. 193–198.
[51] R. Schnabel, R. Wahl, and R. Klein, "Efficient RANSAC for point-cloud shape detection," *Comput. Graph. Forum*, vol. 26, no. 2, pp. 214–226, 2007.
[52] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "LabelMe: A database and web-based tool for image annotation," *Int. J. Comput. Vis.*, vol. 77, nos. 1–3, pp. 157–173, May 2008, doi: 10.1007/s11263-007-0090-8.
[53] J. Niemeyer, F. Rottensteiner, and U. Soergel, "Contextual classification of LiDAR data and building object detection in urban areas," *ISPRS J. Photogramm. Remote Sens.*, vol. 87, pp. 152–165, Jan. 2014.
[54] G. Can, D. Mantegazza, G. Abbate, S. Chappuis, and A. Giusti, "Semantic segmentation on Swiss3DCities: A benchmark study on aerial photogrammetric 3D pointcloud dataset," *Pattern Recognit. Lett.*, vol. 150, pp. 108–114, Oct. 2021.
[55] M. Kölle *et al.*, "The Hessigheim 3D (H3D) benchmark on semantic segmentation of high-resolution 3D point clouds and textured meshes from UAV LiDAR and multi-view-stereo," *ISPRS Open J. Photogramm. Remote Sens.*, vol. 1, p. 11, Oct. 2021.
[56] W. Gao, L. Nan, B. Boom, and H. Ledoux, "SUM: A benchmark dataset of semantic urban meshes," *ISPRS J. Photogramm. Remote Sens.*, vol. 179, pp. 108–120, Sep. 2021.
[57] Y. Chen, Y. Wang, P. Lu, Y. Chen, and G. Wang, "Large-scale structure from motion with semantic constraints of aerial images," in *Proc. Chin. Conf. Pattern Recognit. Comput. Vis. (PRCV)*. Berlin, Germany: Springer, 2018, pp. 347–359.
[58] Y. Liu, F. Xue, and H. Huang, "UrbanScene3D: A large scale urban scene dataset and simulator," 2021, *arXiv: 2107.04286*.
[59] M. Cordts *et al.*, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.
[60] 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.
[61] A. Van Etten, D. Lindenbaum, and T. M. Bacastow, "SpaceNet: A remote sensing dataset and challenge series," 2018, *arXiv:1807.01232*.
[62] D. Lam *et al.*, "xView: Objects in context in overhead imagery," 2018, *arXiv:1802.07856*.
[63] P. Zhu *et al.*, "Detection and tracking meet drones challenge," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Oct. 14, 2021, doi: 10.1109/TPAMI.2021.3119563.
[64] G.-S. Xia *et al.*, "DOTA: A large-scale dataset for object detection in aerial images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2018, pp. 3974–3983.
[65] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1, Jul. 2001, pp. 105–112.
[66] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002.
[67] T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient K-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 7, pp. 881–892, Jul. 2002.
[68] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.
[69] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Trans. Image Process.*, vol. 28, no. 4, pp. 1923–1938, Apr. 2019.

Jiazhou Chen received the joint Ph.D. degrees in computer science from the Institut National de Recherche en Informatique et en Automatique (INRIA) Bordeaux Sud-Ouest, Talence, France, and the State Key Laboratory of Computer Aided Design (CAD) and Computer Graphics (CG), Zhejiang University, Hangzhou, China, in 2012.

He is currently an Associate Professor with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou. His research interests include computer graphics, computer vision, and visualization.

Yanghui Xu received the M.Sc. degree in software engineering from the Zhejiang University of Technology, Hangzhou, China, in 2019, where he is currently pursuing the master's degree with the College of Computer Science and Technology.

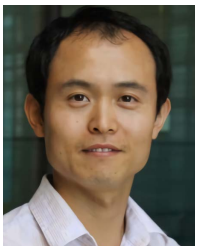
His research interests include computer graphics and computer vision.

Shufang Lu received the B.Sc. degree in software engineering from Wuhan University, Wuhan, China, in 2007, and the Ph.D. degree in computer science and technology from Zhejiang University, Hangzhou, China, in 2013.

She is an Associate Professor with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou. Her research interests include computer graphics and computer vision.

Ronghua Liang (Senior Member, IEEE) received the Ph.D. degree in computer science from Zhejiang University, Hangzhou, China, in 2003.

He was a Research Fellow with the University of Bedfordshire, Luton, U.K., from April 2004 to July 2005 and as a Visiting Scholar with the University of California at Davis, Davis, CA, USA, from March 2010 to March 2011. He is currently a Professor of computer science and the Dean of the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou. He has published more than 80 papers in leading international journals and conferences, including the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, the IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, the IEEE INFORMATION VISUALIZATION, International Joint Conference on Artificial Intelligence (IJCAI), and AAAI. His research interests include image processing, pattern recognition, and visual analytics.



Liangliang Nan received the B.S. degree in material science and engineering from the Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2003, and the Ph.D. degree in mechatronics engineering from the Graduate University of the Chinese Academy of Sciences, Beijing, China, in 2009.

From 2009 to 2013, he was an Assistant and then an Associate Researcher at the Shenzhen Institute of Advanced Technology (SIAT), Chinese Academy of Sciences, Beijing. From 2013 to 2018, he worked at the Visual Computing Center, King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia, as a Research Scientist. He is currently an Assistant Professor with the Delft University of Technology (TU Delft), Delft, The Netherlands, where he is leading the AI Laboratory on 3D Urban Understanding (3DUU). His research interests include computer graphics, computer vision, 3D geoinformation, and machine learning.