RoofIt: 2D Parametric Sketch Extraction from Rough Models for High-quality Roof Reconstruction

Anonymous cvm submission

Paper ID 636

Abstract

3D building roof reconstruction remains a challenging task in urban modeling, particularly when processing rough roof models containing noise and structural imperfections. While existing methods have achieved significant progress, they often fail to maintain reconstruction quality when handling such inputs. This paper presents a unified reconstruction pipeline that transforms rough roof models into high-quality reconstructions by integrating 2D sketch optimization. Our approach begins by applying statistical priors to segment the input rough roof model into planar primitives with constrained orientations. We then extract 2D roof sketches and establish precise correspondences between sketch elements and planar segments. A structure-aware optimization process enforces geometric constraints to refine these sketches. The optimized parametric sketches guide the correction of planar primitives, which are subsequently fused into an enhanced 3D roof surface representation. Experimental evaluations demonstrate that our method outperforms current baseline approaches, achieving superior reconstruction accuracy while effectively handling noise and structural irregularities inherent in rough roof models. The proposed pipeline proves particularly effective in recovering sharp features and maintaining surface regularity from challenging input data.

Keywords: ***roof reconstruction, 3D shape analysis, sketch analysis, planar primitives***

1. Introduction

Three-dimensional building reconstruction serves as a fundamental component in digital city applications, ranging from urban planning to virtual reality systems. Among various architectural elements, roof structures present particularly challenging reconstruction targets due to their complex geometric configurations and frequent imperfections in input data. While existing reconstruction techniques have made considerable progress, the accurate recovery of roof geometry from rough 3D models remains an open research problem with significant practical implications.

Current roof reconstruction approaches generally fall into three categories: Footprint-based methods [29, 28] leverage 2D contours extracted from aerial imagery, but their effectiveness diminishes with complex roof structures and noisy data conditions. Learning-based techniques [4, 25] employing implicit neural representations demonstrate impressive reconstruction fidelity, yet often fail to maintain crucial geometric regularities in practical scenarios. Plane-based approaches [10, 16] produce regular surfaces but suffer from error accumulation during plane detection and intersection operations.

The fundamental challenge lies in simultaneously addressing three critical yet often conflicting requirements in roof reconstruction. First, robustness to imperfect inputs must handle real-world scanning artifacts including substantial noise and irregular sampling patterns that collectively corrupt geometric features. Second, precise preservation of architectural features demands sub-degree accuracy for characteristic roof elements like ridges and valleys, which typically form 30°-90° angles. Third, maintaining surface regularity requires high-level flatness in planar components coupled with watertight junctions. As evidenced in Fig. 6, current approaches struggle with these simultaneous demands - learning methods handle noise at the expense of regularity, while plane-based techniques ensure flatness but struggle under noisy conditions, particularly with real-world scan data where all three challenges interact complexly.

We present a novel reconstruction pipeline that bridges 2D sketch optimization with 3D geometry refinement to overcome these limitations. Our key insight recognizes that while 3D roof data may contain significant noise, their 2D projections often preserve recognizable structural patterns. By establishing bidirectional connections between 2D sketches and 3D planar primitives, our method achieves robust reconstruction while maintaining geometric regularity. Specifically, our method addresses these limitations through a hybrid 2D-3D optimization framework that combines three key innovations. (1) The multi-level sketches

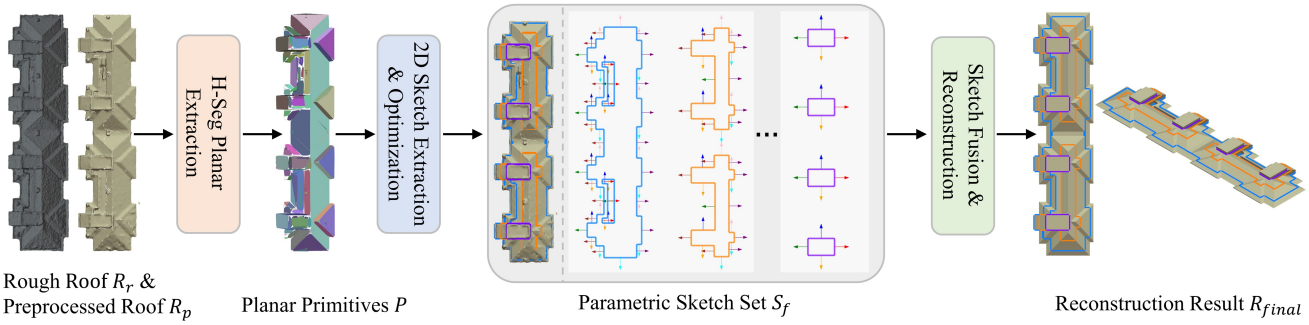

Figure 1. Overview of the proposed method. Our method first preprocesses the input rough roof model R_r, including smoothing and removal of irrelevant structures. Next, we perform **Hierarchical-Segmentation-Based Planar Extraction(H-Seg Planar Extraction)** on the preprocessed roof model R_p to obtain the initial planar primitives P. Then, with **2D Sketch Extraction and Optimization**, we extract and optimize leveled 2D sketches along the z-axis direction from the point cloud associated with P, and after filtering, obtain the sketch set S_f. Finally, with **Sketch Fusion and Reconstruction**, we fuse S_f, transfer the 2D sketch correction data to 3D, convert it into planar primitives, and use a high-quality reconstruction method to transform the planar primitives into a high-quality roof model R_{final}, achieving reconstruction from rough roof to high-quality roof.

extraction process projects 3D noise into multiple 2D domains where structural patterns remain statistically distinguishable. (2) Bidirectional constraint propagation then enforces 3D geometric regularity while preserving 2D-extracted features for 2D sketch optimization. (3) An adaptive fusion mechanism dynamically weights 2D parametric sketch contributions for high-quality 3D roof reconstruction.

Our experimental results demonstrate significant improvements over existing methods, particularly in handling noisy inputs while preserving sharp features. The pipeline achieves higher accuracy than current plane-based approaches and maintains better feature preservation compared to learning-based methods, as measured on a highly challenging benchmark.

The main contributions of our work include:

- A unified reconstruction framework that integrates 2D sketch optimization with 3D geometry refinement, demonstrating superior performance on rough roof models compared to existing methods
- An effective parametric sketch extraction and labeling strategy that preserves structural relationships between 2D elements and 3D planar segments, even with noisy input data
- A novel optimization scheme that transfers geometric corrections from 2D parametric sketches to 3D planar primitives, enabling high-fidelity 3D roof reconstruction

2. Related Works

2.1. Roof Reconstruction

As a core task in urban reconstruction, the automatic generation of 3D models from physically measured data heavily relies on accurate roof reconstruction. Numerous studies have explored roof reconstruction based on traditional algorithmic optimization. [2] proposed a three-step approach involving primitive clustering, boundary representation, and geometric modeling to reconstruct building models from airborne LiDARpointclouds. [7] introduced an automated top-down pipeline that leverages a predefined library of basic roof shapes to generate models conforming to the input data. [12] represented building geometries using a set of well-aligned boxes and selected the optimal subset to approximate the structure. [16] utilized binary programming to achieve polygonal surface reconstruction for buildings. [31] detected locally fitted planar primitives and global regularity from 2.5D point clouds to achieve high-quality building reconstruction. [18] employed a robust clustering algorithm to partition data into segments, followed by boundary extraction to reconstruct polygonal building models. [19] encoded roof topology using graph structures to enable 3D roof reconstruction from single images.

Alternatively, some studies adopted data-driven or model-driven approaches. [6] applied supervised machine learning to identify the most probable roof models from point clouds based on typical roof features. [11] combined deep learning with an end-to-end trainable network to reconstruct 3D roof wireframes from airborne LiDAR point clouds. [15] introduced a frequency-domain feature learning strategy to extract comprehensive geometric features, improving roof topology inference. [20] employed neural

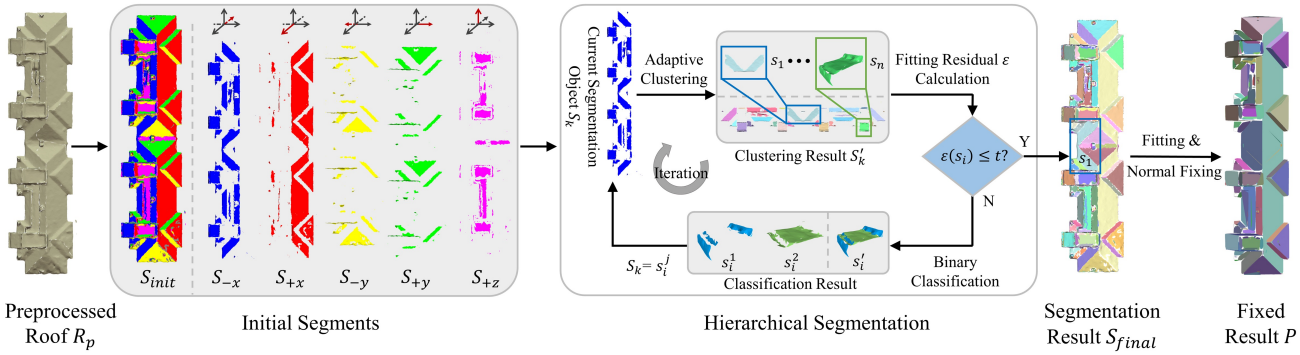

Figure 2. H-seg Planar Extraction: Hierarchical-Segmentation-Based Planar Extraction

networks to detect buildings and roof planes in images before reconstructing models using vectorized roof surfaces. [29] leveraged relational neural network architectures to learn outdoor building reconstruction from vector graphics. [27] achieved LoD1 building reconstruction from multi-view aerial images using deep learning. To address the data scarcity in urban modeling, [24] introduced Building3D, a large-scale benchmark containing over 160,000 buildings with point clouds, meshes, and wireframes. This benchmark has facilitated the development of numerous deep learning-based methods [14, 9, 13] for roof reconstruction in recent years.

However, these methods are not well-suited for our task, as real-world roof data often contain substantial noise, and existing reconstruction approaches lack sufficient robustness to effectively handle such interference. Although [14, 9, 13] achieves the conversion from scanned roof point clouds to 3D roof wireframes, their application and performance have been demonstrated only on roofs of relatively simple structure. Our work addresses this gap by introducing a noise-resistant scheme which can achieve high-fidelity roof reconstruction from structurally complex roof under challenging conditions.

2.2. High-Quality 3D Reconstruction

Existing methods can be broadly categorized into optimization-based traditional approaches and learning-based neural methods, each exhibiting distinct advantages in different application scenarios.

Among traditional methods, early research adopted probabilistic search approaches like RANSAC [21] or variational optimization methods [3, 22] to fit candidate primitive surface patches to input shapes (which are discretized as point clouds or polygon meshes), iterating between primitive fitting and segmentation. [23, 1] focused on polygonal surface reconstruction, forming convex polygonal decompositions through plane arrangement strategies to reconstruct smooth and watertight results.

With the emergence of deep learning methods, data-driven approaches have regained attention for solving high-quality reconstruction problems. [8, 26] trained point cloud-based neural networks to assign primitive types and parameters to each input point, effectively addressing primitive segmentation and fitting problems by learning from large-scale annotated CAD datasets. [32] seamlessly converted input point clouds into versatile piecewise linear (PWL) curve representations through a learning framework, achieving 3D wireframe reconstruction via spline fitting algorithms.[4] employed gaussian curvature constraints to obtain high-fidelity and smooth reconstruction results. [17] focused on polygonal model reconstruction and employed an autoregressive network to iteratively predict vertices and planes.

Our method achieves high-quality roof reconstruction from coarse 3D inputs through a novel pipeline. The pipeline works by optimizing extracted 2D roof contour sketches, transmitting these corrections to the 3D domain, and finally combining the refined structure with a polygonal surface reconstruction method.

3. Method

Our method mainly consists of three core modules, as shown in Figure 1. We take a rough 3D roof model R_r as input, and obtain R_p through preprocessing before it enters the core modules. First, through the hierarchical-segmentation-based planar extraction(H-seg planar extraction), we perform hierarchical segmentation on R_p and extract the initial set of planar primitives P. Next, we extract multi-level 2D sketches from the sampled points $\mathcal{P}$ of R_p, obtaining the 2D sketch set S. We then denoise and optimize each level of sketches in S, and after filtering, obtain the refined 2D sketch set S_f. Finally, we select a key sketch from S_f and fuse the remaining sketches with it as the foundation. The fusion sketch guide the correction of planar primitive set P, which are subsequently fused into an enhanced 3D roof surface representation.

3.1. Statistical Prior

Roof structures typically exhibit a high degree of geometric regularity. This regularity is primarily reflected in the strong constraints on the spatial orientation of their constituent planes, meaning that different roof surfaces often adhere to a limited set of normal vector directions and frequently exhibit rotational or mirror symmetry with respect to one another.

We defined the positive x-axis direction as the reference starting point (0°) to analyze the distribution of normal vector azimuths in roof structures. The top 10 most frequent normal vectors are highlighted in Figure 3, revealing distinct clustering around 0, $\pm\pi/2$, and $\pm\pi$ orientations. This indicates their 2D projections primarily concentrate at four cardinal directions: $(1, 0)$, $(-1, 0)$, $(0, 1)$, and $(0, -1)$.

Based on statistical analysis of extensive real-world building data, an effective structural prior can be introduced: in a single roof structure, the number of distinct normal vector orientations usually does not exceed nine. This prior constraint not only aligns with the design conventions of most man-made structures but also helps reduce the complexity of the solution space during the reconstruction process, thereby enhancing the rationality and stability of the reconstruction results. To further simplify the problem and leverage the inherent regularities of building structures, we also incorporate the axis-aligned assumption. Under this constraint, the orientation of roof planes is limited to a finite set of fundamental normal vectors. Additionally, we assume that the upward direction of the roof is the positive z-axis.

Building upon this observed concentration of normal directions and the inherent structural regularity, we can formalize the orientation constraint under the axis-aligned assumption. This formulation translates the geometric regularity of roof structures into a highly constrained discrete problem, where the unit normal vector of any roof plane is restricted to a finite set composed of **nine fundamental directional instances**. This discrete direction space $\mathcal{D}$ constitutes a complete set of all possible roof plane orientations, defined mathematically as:

$$\mathcal{D} = \mathcal{D}_{\text{vertical}} \cup \mathcal{D}_{\text{horizontal}} \cup \mathcal{D}_{\text{inclined}} \tag{1}$$

This set consists of the following three mutually exclusive subsets, corresponding to three basic geometric types:

1. **Vertical Direction Subset**: This subset contains all planar orientations whose normal vectors are parallel to the ground (i.e., $n_z = 0$). Such planes typically correspond to building gables or vertical facades.

   $$\mathcal{D}_{\text{vertical}} = \{(1, 0, 0),\ (-1, 0, 0),\ (0, 1, 0),\ (0, -1, 0)\}$$

   Number of instances: 4. These represent the normal vectors aligned with the positive and negative directions of the X-axis and Y-axis in the Cartesian coordinate system.

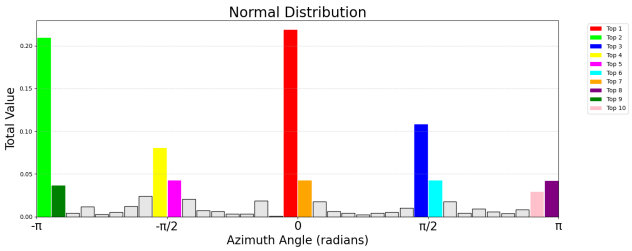

Figure 3. Normal distribution in roof.

2. **Horizontal Direction Subset**: This subset contains only the planar orientation whose normal vector is perpendicular to the ground (i.e., parallel to the Z-axis).

   $$\mathcal{D}_{\text{horizontal}} = \{(0, 0, 1)\}$$

   Number of instances: 1. Represents the horizontal roof surface with a vertically upward normal vector.

3. **Inclined Direction Subset**: This subset contains all orientations of sloped roof surfaces. Their normal vectors lie in the X-Z or Y-Z plane and form a fixed inclination angle with the Z-axis. Here, the parameters $a > 0$, $b > 0$, $c > 0$, and satisfy the unit vector constraint $a^2 + c^2 = 1$ or $b^2 + c^2 = 1$.

   $$\mathcal{D}_{\text{inclined}} = \{(a, 0, c),\ (-a, 0, c),\ (0, a, c),\ (0, -a, c)\}$$

   Number of instances: 4. These represent the sloped roof surfaces inclined along the positive or negative directions of the X-axis or Y-axis.

3.2. Preprocess

For rough 3D roof mesh models with significant noise, preprocessing is required as they are difficult to be segmented directly. The first preprocessing step is smoothing. Additionally, fence-like structures often appear around the roof's bottom edges due to data acquisition artifacts. These structures do not belong to the roof itself and introduce segmentation noise, so we remove them. This preprocessing ensures cleaner input data for subsequent hierarchical segmentation and 2D contour extraction.

3.3. H-seg planar extraction

As shown in Figure 2, we extract planar primitives based on Hierarchical Segmentation from R_p. Through statistical analysis, we have found that the orientations of roof surfaces typically exhibit no more than 9 principal directions. To leverage this structural prior and simplify the classification task, we adopt an axis-aligned assumption as the initial categorization criterion. This allows us to project the

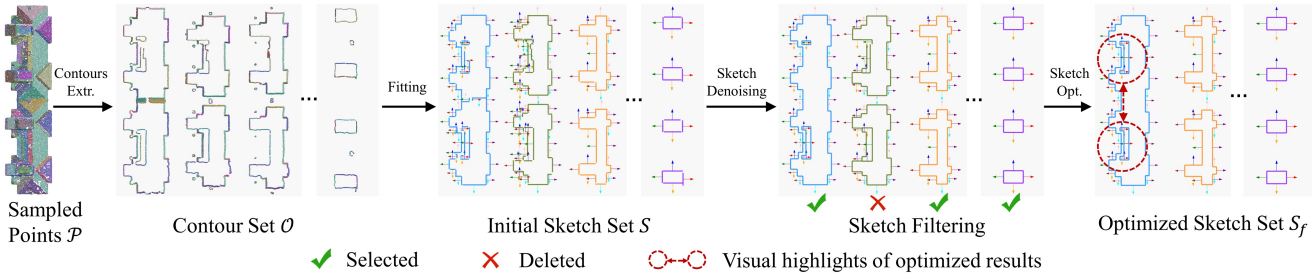

Figure 4. 2D Sketch Extraction and Optimization

continuous orientation space onto a discrete set of fundamental directions. Therefore, we initially classify the triangular faces f_i in R_p. We define 6 categories for roof surface orientations: positive x-axis, negative x-axis, positive y-axis, negative y-axis, positive z-axis, and no orientation (null), denoted as $C = \{+x, -x, +y, -y, +z, null\}$. The corresponding reference direction vectors are defined as: $V = \{\vec{v}_{+x}, \vec{v}_{-x}, \vec{v}_{+y}, \vec{v}_{-y}, \vec{v}_{+z}\}$. For each triangular face f_i, we calculate the angle θ_{ij} between its normal vector $\vec{n}_i$ and each reference direction vector $\vec{v}_j \in V$, and classify it according to the following formula:

$$c_i = \begin{cases} C_j, & \text{if } min_j \theta_{ij} < \theta_{th} \\ null, & \text{if } min_j \theta_{ij} \geq \theta_{th} \end{cases} \tag{2}$$

Here, θ_{th} is the preset angle threshold. In the first case, the category is the one that shares the same subscript as the vector $\vec{v}_j$ that minimizes the angle θ_{ij} (for example, if $\vec{v}_{+x}$ yields the smallest angle, the category is $C_j = +x$). If the smallest angle is less than or equal to θ_{th}, the current triangular face is classified into the corresponding orientation category; when the smallest angle exceeds θ_{th}, we classify them into the no-orientation and discard them. At this point, we obtain the initial classification result $S_{init} = \{S_{+x}, S_{-x}, S_{+y}, S_{-y}, S_{+z}\}$. Next, we perform more detailed segmentation on each category of faces in S_{init}. For the current segmentation target S_k, we use Euclidean clustering to segment it, resulting in a subset of faces $S'_k = \{s_i\}_{i=1}^{N}$. For each subset s_i, we need to approximate s_i as a plane s_i^p. We use a simple strategy to approximate s_i as a plane s_i^p. First, we determine the normal vector of s_i^p through a voting strategy. i.e., we count the occurrences of normal vectors of the triangular faces in s_i, treating similar normal vectors as identical during the counting process. The most frequently occurring normal vector is set as the normal vector $\vec{n}_i^p$ of s_i^p:

$$\vec{n}_i^p = \arg\max_n \sum_{k=1}^{N} \mathbb{I}(\mathbf{n}_k \cdot \mathbf{n} \geq \cos\epsilon) \tag{3}$$

where ϵ is the normal vector similarity threshold. We sample points from s_i to obtain a point set $\mathcal{P}_i$, then downsample this set to get $\hat{\mathcal{P}}_i$. Combining $\hat{\mathcal{P}}_i$ with n_i^p, we compute multiple planes using the point-normal form. s_i^p is determined by selecting the point with the minimal residual value. If the minimal residual $\varepsilon(s_i)$ exceeds the threshold t, it indicates that s_i requires further segmentation. At this point, we perform binary classification on the faces in s_i, dividing them into two categories: one with orientations closer to the z-axis and the other with orientations closer to the x- or y-axis. This yields the binary classification result $s'_i = \left\{s_i^j \subset s_i\right\}_{j=1}^{2}$. For each subset s_i^j in s'_i, we iteratively apply the aforementioned process until all segmentation results reach the minimal segmentation requirement. At this stage, we obtain the segmentation result S_{final} for the roof. However, the current plane orientations in the set S_{final} are disordered. Therefore, we further perform consistency correction on these plane orientations, adjusting them to 9 standard directions. Ultimately, we obtain a plane set P with at most 9 distinct orientations. Through this strategy, we effectively extract planar primitives, significantly suppressing the generation of fragmented faces.

3.4. Parametric Sketch Extraction and Optimization

As shown in Figure 4, we extract 2D parametric Sketch from sampled points $\mathcal{P}$. We first voxelize the sampled point set $\mathcal{P}$ from the segmentation process to generate uniform voxel grids $V = \{v_{x,y,z}\}$, obtaining the contour set $\mathcal{O} = \{o_k\}_{k=1}^{M}$, where the contour of each level is defined as follows:

$$o_k = \{v_{x,y,k} \mid x \in [1, X] \cap \mathbb{Z}, y \in [1, Y] \cap \mathbb{Z}\} \tag{4}$$

Here, X and Y represent the resolution of the voxel grid in the x and y directions respectively, and $v_{x,y,k}$ denotes all voxel grids with z-index k. We first project each contour level onto the XOY plane. For each 2D voxel grid, its associated internal points $p_{x,y,k} = \{p_i \in \mathcal{P}\}_{i=1}^{N}$ contain information about the planar primitives. Additionally, we assign an orientation to each voxel grid, defining four 2D orientation categories $D = \{up, down, left, right\}$. We project the normal vector $\vec{n}_{p_i} = (n_x, n_y, n_z)$ of the planar primitive corresponding to p_i onto the XOY plane. Since the 9

standard normal vectors must satisfy $n_x \cdot n_y = 0$, the projected normal vectors can only yield four possible results:

$$\vec{n}_{p_i}^{proj} = \begin{cases} (1,0), & \text{if } n_x > 0 \\ (-1,0), & \text{if } n_x < 0 \\ (0,1), & \text{if } n_y > 0 \\ (0,-1), & \text{if } n_y < 0 \end{cases} \tag{5}$$

These four possible results are mapped to the orientation labels in D as follows:

$$lable(\vec{n}_{p_i}^{proj}) = \begin{cases} right, & \text{if } \vec{n}_{p_i}^{proj} = (1,0) \\ left, & \text{if } \vec{n}_{p_i}^{proj} = (-1,0) \\ up, & \text{if } \vec{n}_{p_i}^{proj} = (0,1) \\ down, & \text{if } \vec{n}_{p_i}^{proj} = (0,-1) \end{cases} \tag{6}$$

We assign the orientation label with the highest votes to each voxel grid through a voting process. Since our contour extraction is performed along the z-axis, normal vectors oriented in the positive z-direction are ignored during voting. In each level of the contour, we segment the contours based on the orientation of the voxel grids and their adjacency relationships, grouping voxel grids with the same orientation and adjacency into a single result. Next, we perform line segment fitting on each segmented result to obtain the parametric 2D sketch set $S = \{\hat{s}_k\}_{k=1}^{M}$. Simultaneously, we transfer the planar primitive information contained in the voxel grids to the line segments with voting strategy. Specifically, we employ a two-stage voting scheme to determine the planar primitive label for each line segment:

1. **Voxel-level Assignment:** For each voxel, we compute the frequency of the planar primitives associated with its contained points $p_{x,y,k}$. The voxel is assigned the label of the planar primitive with the highest frequency.

2. **Segment-level Assignment:** For each line segment, we aggregate the labels from all voxels it associates with. The segment is then assigned the label that achieves a majority vote from these voxels.

After fitting, we merge adjacent and collinear line segments to simplify topological relationships and enhance geometric consistency, while also removing extremely short line segments. Subsequently, we eliminate asymmetric short line segments from the sketch, with the symmetry center being the horizontal center of the sketch's bounding box. Finally, we construct a graph where line segment endpoints serve as nodes and the line segments themselves as edges, then remove nodes with a degree of 1. These two processes primarily aim to eliminate noise from the sketch.

At this stage, we obtain the denoised sketch S_d. We filter S_d based on closure. Sketches located at lower levels but associated with only a small number of planar primitives are considered non-contributory sketches, as they often contain noise structures. Therefore, we remove such sketches from S_d. At this point, we obtain the filtered sketch set S_{sel}. For each selected sketch $s_{sel}^i \in S_{sel}$, we apply collinearity and symmetry constraints to its line segments. Additionally, for line segments belonging to the same planar label but located in different levels, they must strictly lie on the associated plane. We further repair the sketches using symmetry, with the symmetry center still being the horizontal center of the sketch's bounding box. Each connected component in s_{sel}^i should have a symmetric counterpart or be self-symmetric. For connected components missing symmetric counterparts, we add their mirrored versions to complete the structure. At this stage, we obtain the corrected and optimized sketch set S_f.

Through the aforementioned optimization and correction steps, we also recalculate the planes based on the planar orientation labels of the line segments and use the computed planes as corrections to the original planar primitives. We leverage the adjacency relationships between line segments within each sketch to obtain the clipping results for the corrected planes. Given that each line segment l_i has been assigned a planar primitive label, we calculate the initial corresponding plane p_i using its endpoints and the normal vector associated with that primitive. The two adjacent line segments of l_i are denoted as l_i^1 and l_i^2. The planes p_i^1 and p_i^2 are calculated using the normal vectors corresponding to these segments. We use p_i^1 and p_i^2 to perform an initial clipping on plane p_i, resulting in the clipped plane $p_i{}'$. Then, based on the original primitive label of l_i, we use the inliers of the original planar primitive to further clip $p_i{}'$, yielding the corrected plane p_c^i corresponding to l_i. In Vertex Groups, the boundaries of planes are typically constrained by inliers, so we sample p_c^i and use the samples as its inliers. Through this sketch-based approach, we effectively remove noise from the original fitted planes. Additionally, by recalculating planes using the line segments and their corresponding normal vectors, we can further correct the original planes.

3.5. Sketch Fusion and Reconstruction

We fuse the selected sketches to reconstruct the complete roof, as shown in Figure 5. First, we select a key sketch $s_{key} \in S_f$ and iteratively add line segments from its upper-level sketches to it, obtaining the fused sketche s_{fuse}. Each line segment in the sketch has an original planar label, and the selection criterion for the key sketch is to associate as many planar primitives as possible, typically located at lower levels. Each time a line segment l_j is added, the following rules are applied:

1. **Planar Label Check:** Determine whether the plane ρ corresponding to l_j is already included in the fused sketch. If not, add l_j. If yes, check whether the dis-

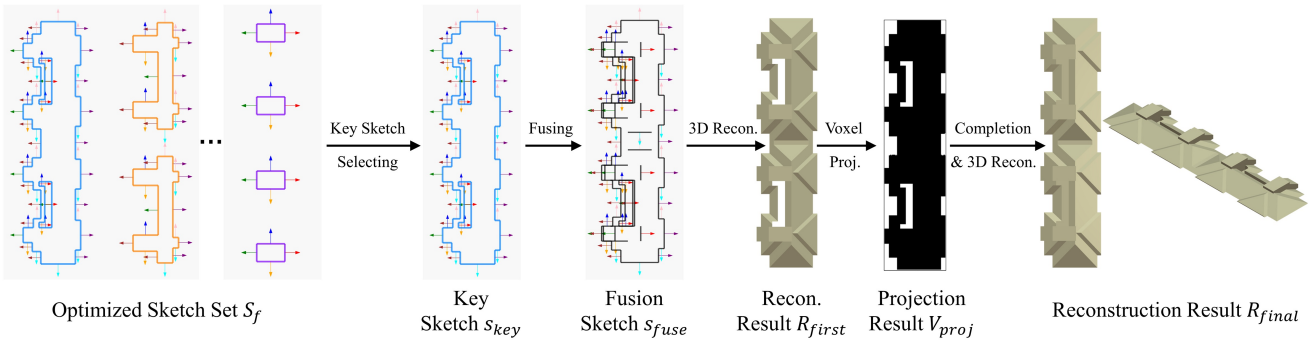

Figure 5. Sketch Fusion and Reconstruction

tribution of inliers I_j^ρ of l_j's plane ρ differs from the distribution of inliers I^ρ of plane ρ in the fused sketch. If inconsistent, add l_j; otherwise, discard it.

2. **Chamfer Distance Check:** Compute the one-way chamfer distance between the corrected surface of l_j and the already fused planes. If the chamfer distance is below a threshold, add l_j; otherwise, discard it.

3. **Inlier Count Check:** If the number of inliers for the plane corresponding to l_j exceeds a threshold, add l_j; otherwise, discard it.

The rationale behind these rules is as follows: For (1), we first ensure that planar primitives are not redundantly added. The distribution consistency check accounts for cases where a single original plane may be split into multiple surfaces during boundary-aware clipping. These surfaces share the same planar label but may have different inlier distributions, so this strategy helps merge them. For (2), surfaces with large chamfer distances are considered missing planes in the current reconstruction, while this also helps filter out noisy surfaces. For (3), facets with too few inliers are treated as noise and discarded.

Then, we convert the sketches into planar primitives set P_f which subsequently used as input for initial reconstruction, yielding the initial reconstruction result R_{first}. At this stage, R_{first} may exhibit some hollow structures, which we subsequently repair. To obtain the bottom clipping surface for these hollow structures, we leverage R_{first} as prior knowledge. First, we voxelize the initial reconstruction result and project the voxels onto the XOY plane to obtain V_{proj}. For the 2D voxel result, we perform binary classification on each voxel grid, assigning a value of 1 to occupied voxel grids and 0 to unoccupied ones. Next, we extract voxel grids with a value of 0 and identify connected components using DFS. Connected components that do not include edge regions are designated as bottom clipping regions. We then add appropriate z-coordinates to the center points of the 2D voxel grids in these regions to convert them into 3D points, which serve as inliers for the bottom clipping surface. Subsequently, we add the bottom clipping surface to P_f, resulting in the updated planar primitive set P_f'. Using P_f' as input, we perform reconstruction algorithm again to obtain the final roof reconstruction result R_{final}.

Table 1. Quantitative comparisons with the state-of-the-art methods NeurCAD[4], PrimFit[10], KSR[1], and Compod[23].

Method	NeurCAD	PrimFit	KSR	Compod	Ours
Times(s)	2394.4	5160.5	462.8	134.8	**132.4**
RMS↓	0.0763	0.0974	0.0424	0.0557	**0.0387**
MM↓	0.0690	0.0587	0.0365	0.0446	**0.0218**

4. Experiments

In this section, we first present the implementation details of our experiments. We then conduct comparative evaluations against both learning-based and plane-based methods on sloped roof datasets. Finally, we perform ablation studies to validate the contribution of key components in our framework. Additional experimental results are provided in the supplementary materials.

4.1. Experimental Setup

4.1.1 Dataset

The dataset we use consists of real-world scanned data of sloped roofs in the form of 3D mesh models without bottom surfaces. These data which comprises 70 distinct roof models contain sufficiently complex roof structures incorporating various styles of sloped roofs, along with substantial noise structures, presenting significant challenges that can comprehensively test the algorithm's robustness, effectiveness, and generalization capability. The dataset is comprehensively demonstrated in the supplementary materials.

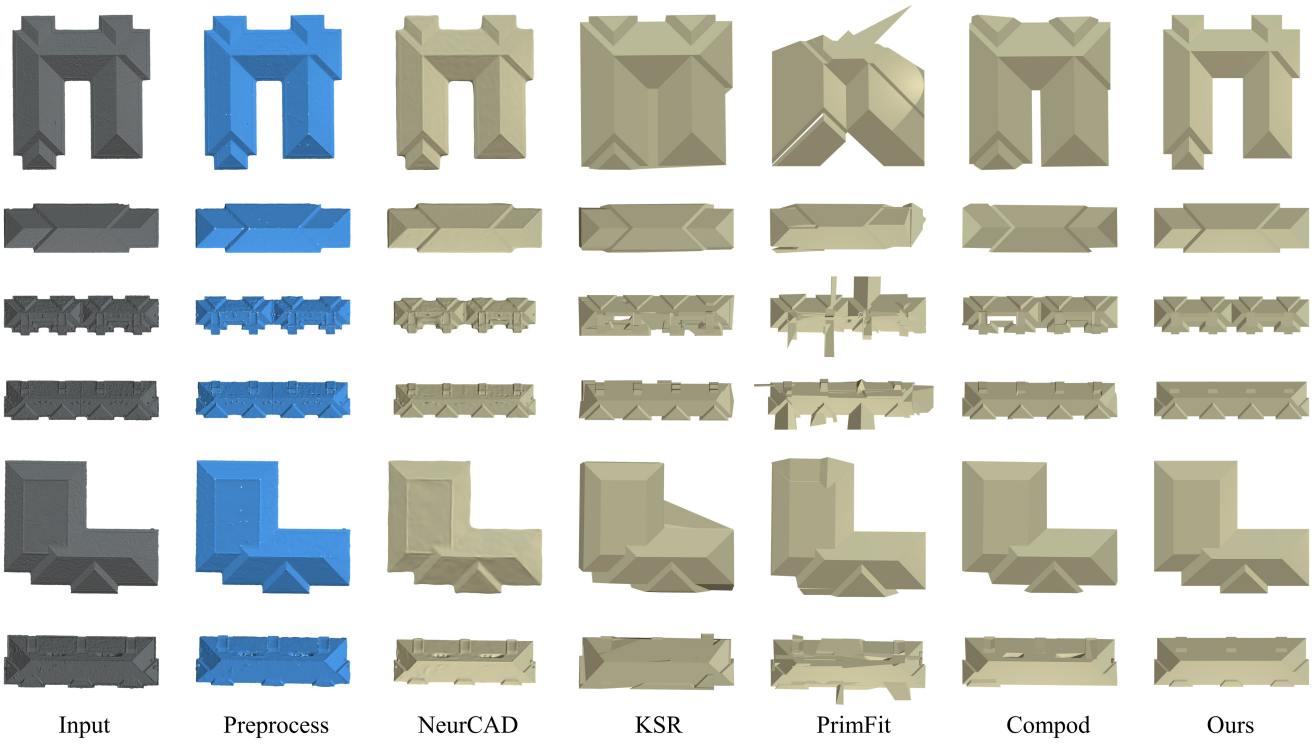

Figure 6. Comparison with state-of-the-art approaches.

4.1.2 Implementation Details

For each roof model, we normalize it to the range $[-1, 1]^3$. During preprocessing, we employ the Dual-DMP [5] method for smoothing. We employ the Compod[23] method for 3D mesh reconstruction. In the segmentation phase, we set the classification angle threshold $\theta_{th} = 10°$, which enables accurate segmentation for most roof data at this setting, with a total sampling point count of 100K.

Following [30], we adopt two complementary metrics to evaluate geometric accuracy: the Root Mean Square Hausdorff Distance (RMS) and the Maximum Mean Hausdorff Distance (MM). The RMS captures the global reconstruction fidelity, whereas the MM pinpoints localized regions of maximum deviation. This dual-metric approach facilitates a balanced assessment of both overall and worst-case performance.

4.2. Comparison

Our method is compared with the learning-based method(NeurCAD[4]) and plane-based approaches(PrimFit[10], KSR[1], Compod[23]). All comparative methods take the preprocessed roof model as input. In the evaluation of computational efficiency, we assess the entire reconstruction pipeline (including both plane detection and 3D mesh reconstruction).

Quantitative comparisons against state-of-the-art methods are summarized in Table 1. The results demonstrate that our method achieves a superior balance between efficiency and accuracy. In terms of computational efficiency, our approach (132.4s) is on par with the fastest method, Compod (134.8s), and is significantly faster than NeurCAD (2394.4s), PrimFit (5160.5s), and KSR (462.8s). More importantly, our method attains the best geometric accuracy across both fidelity metrics, achieving the lowest RMS of 0.0387 and the lowest MM of 0.0218. This indicates that our reconstructions are not only globally the most faithful but also exhibit the smallest localized errors.

As shown in the qualitative results of Figure 6, com-

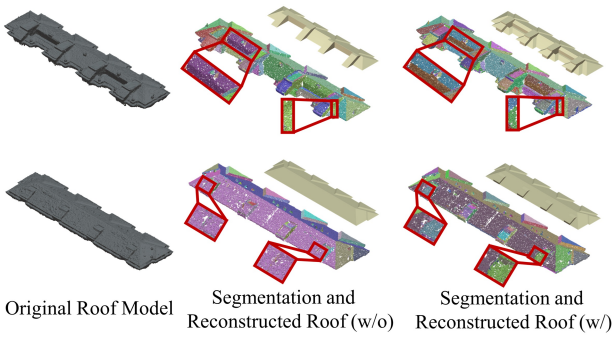

Figure 7. Visualizing the reconstruction result without(w/o) and with(w/) hierarchical segmentation strategy.

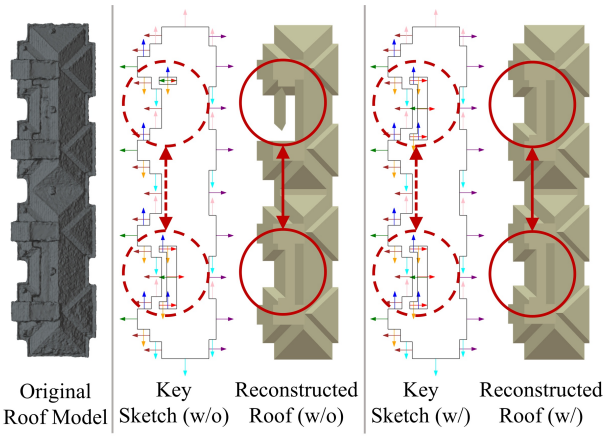

Figure 8. Visualizing the reconstruction result without(w/o) and with(w/) sketch symmetry repair strategy.

pared to these baseline methods, our method demonstrates high-quality reconstruction results. Our approach enforces consistency on the normal vectors of planar primitives during the segmentation stage based on statistical priors, resulting in more regularized roof structures. Additionally, we refine the planar primitives through sketch-based optimization and remove noise structures, leading to superior reconstruction quality. For learning-based method, NeurCAD reconstruct complete roofs without sharp feature and the resulting surfaces are not sufficiently flat. For plane-based methods, both KSR and Compod are affected by noise structures, leading to redundant structures and partial distortion in their reconstruction results. PrimFit performs even worse, as it cannot effectively handle noise structures, resulting in reconstructions with substantial redundant elements. Our method addresses noise structures by propagating the corrections from 2D geometric constraints in sketches and noise removal to optimize the planar primitives, enabling the effective reconstruction of high-quality roof models.

4.3. Ablation Study

4.3.1 Evaluation Of Hierarchical Segmentation

We conducted a comparative experiment by disabling H-Seg, where we only performed clustering on the initial classification results while keeping all other settings unchanged. As shown in the results in Figure 7, the third column displays the segmentation results without hierarchical segmentation, which, when compared to the results with hierarchical segmentation enabled, exhibit errors that lead to missing roof structures. These issues arise because, after clustering, there still exist clusters containing faces oriented in different directions. If these faces are directly approximated as planes, the segmentation results will be incorrect, causing missing coverage in the segmented planar primitives. Therefore, hierarchical segmentation effectively separates faces with different orientations within clusters, significantly improving the accuracy of the segmentation results.

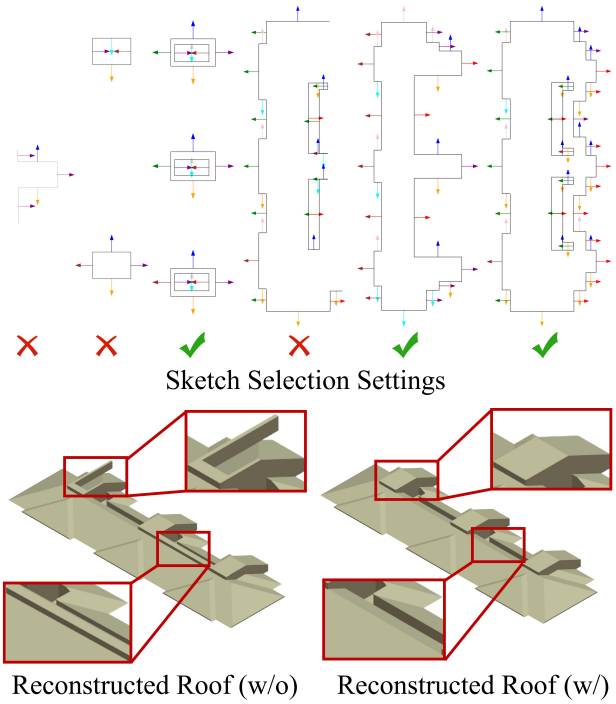

Figure 9. Visualizing the reconstruction result without(w/o) and with(w/) sketch filtering.

4.3.2 Evaluation of Sketch Symmetry Repair

As shown in Figure 8, the reconstruction without symmetric repair clearly shows missing roof structures (red boxes). Since the rough roof contains numerous noise structures, the extracted sketches also include many noise line segments. After denoising, if key roof structures are severely affected by noise, their related line segments are also treated as noise and removed. Without symmetric repair, these major structures would be missing in the final reconstruction. Therefore, symmetric sketch repair can restore those roof structures severely affected by noise, significantly enhancing our algorithm's robustness.

4.3.3 Evaluation of Sketch Filtering

As shown in Figure 9, when all sketches participate in fusion, noise structures appear in the final reconstruction. Moreover, due to the inclusion of noise-related planar primitives, the reconstruction quality deteriorates, resulting in redundant surfaces. These issues occur because disabling sketch filtering allows non-effective planar primitives to

be incorporated into the fusion process, compromising reconstruction quality. Sketch filtering effectively removes sketches containing line segments related to non-effective planar primitives, thereby improving final reconstruction quality.

4.3.4 Evaluation of Key Sketch Selecting

We conducted a comparative experiment on the same model by randomly selecting different sketches as the key sketch while keeping other settings unchanged. As shown in Figure 10, selecting the sketch containing the most planar primitives as the key sketch yields better reconstruction results. This is because the key sketch serves as the foundation for the fusion module - when adding line segments to it, noise segments and duplicate segments are filtered based on this foundation. If the key sketch contains fewer associated planar primitives, noise segments may be confused with some non-noise segments, causing valid segments to be incorrectly filtered out and resulting in missing structures in the reconstruction.

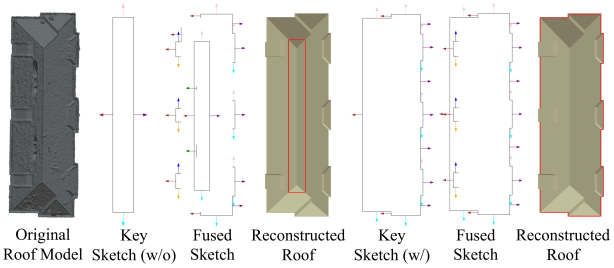

Figure 10. Visualizing the reconstruction result without(w/o) and with(w/) key sketch selecting strategy.

5. Conclusion

This paper introduces a novel and unified computational framework that effectively bridges 2D sketch-based abstraction with 3D geometric refinement for robust roof reconstruction from imperfect data. Departing from methods that operate solely in 3D, our key innovation lies in a cross-domain optimization strategy. The pipeline begins with a hierarchical segmentation of the input rough roof mesh, extracting a set of planar primitives regulated by robust orientation constraints. The core of our method involves the extraction and structural optimization of multi-level 2D roof contour sketches, which provide a topologically clean and compact representation for enforcing regularities. The optimized 2D constraints are then intelligently transferred back to the 3D domain, guiding the geometric correction of the planar primitives. This dual-domain approach allows us to effectively correct errors and impose structural priors that are difficult to enforce directly in 3D. Extensive quantitative and qualitative evaluations on real-world datasets, which encompass a wide spectrum of roof structures from simple to highly complex, demonstrate that our method achieves superior performance, striking an excellent balance between reconstruction fidelity and computational efficiency.

Despite its strengths, the current framework has several limitations that point to fruitful future directions. First, the simplification inherent in our sketching stage leads to the omission of fine-grained details such as small dormers and chimney features. Second, and more critically, our method is primarily designed for geometry correction and regularization rather than geometry completion. It struggles to hallucinate plausible roof structures in cases of significant data loss or large missing sections. Our future work will focus on exploring more advanced frameworks to overcome the current limitations.

References

[1] J.-P. Bauchet and F. Lafarge. Kinetic shape reconstruction. *ACM Transactions on Graphics (TOG)*, 39(5):1–14, 2020. 3, 7, 8

[2] D. Chen, R. Wang, and J. Peethambaran. Topologically aware building rooftop reconstruction from airborne laser scanning point clouds. *IEEE Transactions on Geoscience and Remote Sensing*, 55(12):7032–7052, 2017. 2

[3] D. Cohen-Steiner, P. Alliez, and M. Desbrun. Variational shape approximation. In *ACM SIGGRAPH 2004 Papers*, pages 905–914. 2004. 3

[4] Q. Dong, R. Xu, P. Wang, S. Chen, S. Xin, X. Jia, W. Wang, and C. Tu. Neurcadrecon: Neural representation for reconstructing cad surfaces by enforcing zero gaussian curvature. *arXiv preprint arXiv:2404.13420*, 2024. 1, 3, 7, 8

[5] S. Hattori, T. Yatagawa, Y. Ohtake, and H. Suzuki. Learning self-prior for mesh denoising using dual graph convolutional networks. In *European Conference on Computer Vision*, pages 363–379. Springer, 2022. 8

[6] A. Henn, G. Gröger, V. Stroh, and L. Plümer. Model driven reconstruction of roofs from sparse lidar point clouds. *ISPRS Journal of photogrammetry and remote sensing*, 76:17–29, 2013. 2

[7] H. Huang, C. Brenner, and M. Sester. A generative statistical approach to automatic 3d building roof reconstruction from laser scanning data. *ISPRS Journal of photogrammetry and remote sensing*, 79:29–43, 2013. 2

[8] J. Huang, Y. Zhang, and M. Sun. Primitivenet: Primitive instance segmentation with local primitive embedding under adversarial metric. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 15343–15353, 2021. 3

[9] S. Huang, R. Wang, B. Guo, and H. Yang. Pbwr: Parametric-building-wireframe reconstruction from aerial lidar point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 27778–27787, 2024. 3

[10] J. Jiang, M. Zhao, S. Xin, Y. Yang, H. Wang, X. Jia, and D.-M. Yan. Structure-aware surface reconstruction via primitive assembly. In *Proceedings of the IEEE/CVF International*

Conference on Computer Vision, pages 14171–14180, 2023. 1, 7, 8

[11] L. Li, N. Song, F. Sun, X. Liu, R. Wang, J. Yao, and S. Cao. Point2roof: End-to-end 3d building roof modeling from airborne lidar point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 193:17–28, 2022. 2

[12] M. Li, P. Wonka, and L. Nan. Manhattan-world urban reconstruction from point clouds. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14*, pages 54–69. Springer, 2016. 2

[13] Y. Liu, R. Wang, S. Huang, and G. Cai. Edgediff: Edge-aware diffusion network for building reconstruction from point clouds. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 17008–17018, 2025. 3

[14] Y. Liu, L. Zhu, H. Ye, S. Huang, X. Gao, X. Zheng, and S. Shen. Bwformer: Building wireframe reconstruction from airborne lidar point cloud with transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 22215–22224, 2025. 3

[15] Z. Lu, L. Huan, Q. Ma, and X. Zheng. Holistic geometric feature learning for structured reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 21807–21817, 2023. 2

[16] L. Nan and P. Wonka. Polyfit: Polygonal surface reconstruction from point clouds. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2353–2361, 2017. 1, 2

[17] C. Nash, Y. Ganin, S. A. Eslami, and P. Battaglia. Polygen: An autoregressive generative model of 3d meshes. In *International conference on machine learning*, pages 7220–7229. PMLR, 2020. 3

[18] C. Poullis. Large-scale urban reconstruction with tensor clustering and global boundary refinement. *IEEE transactions on pattern analysis and machine intelligence*, 42(5):1132–1145, 2019. 2

[19] J. Ren, B. Zhang, B. Wu, J. Huang, L. Fan, M. Ovsjanikov, and P. Wonka. Intuitive and efficient roof modeling for reconstruction and synthesis. *arXiv preprint arXiv:2109.07683*, 2021. 2

[20] P. Schuegraf, J. Shan, and K. Bittner. Planes4lod2: Reconstruction of lod-2 building models using a depth attention-based fully convolutional neural network. *ISPRS Journal of Photogrammetry and Remote Sensing*, 211:425–437, 2024. 2

[21] S. Sellán, N. Aigerman, and A. Jacobson. Developability of heightfields via rank minimization. *ACM Trans. Graph.*, 39(4):109, 2020. 3

[22] M. Skrodzki, E. Zimmermann, and K. Polthier. Variational shape approximation of point set surfaces. *Computer Aided Geometric Design*, 80:101875, 2020. 3

[23] R. Sulzer and F. Lafarge. Concise plane arrangements for low-poly surface and volume modelling. In *European Conference on Computer Vision*, pages 357–373. Springer, 2025. 3, 7, 8

[24] R. Wang, S. Huang, and H. Yang. Building3d: An urban-scale dataset and benchmarks for learning roof structures from point clouds. in 2023 ieee. In *CVF International Conference on Computer Vision (ICCV)*, pages 20019–20029, 2023. 3

[25] Z. Wang, Y. Zhang, R. Xu, F. Zhang, P.-S. Wang, S. Chen, S. Xin, W. Wang, and C. Tu. Neural-singular-hessian: Implicit neural representation of unoriented point clouds by enforcing singular hessian. *ACM Transactions on Graphics (TOG)*, 42(6):1–14, 2023. 1

[26] S. Yan, Z. Yang, C. Ma, H. Huang, E. Vouga, and Q. Huang. Hpnet: Deep primitive segmentation using hybrid representations. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2753–2762, 2021. 3

[27] D. Yu, S. Ji, J. Liu, and S. Wei. Automatic 3d building reconstruction from multi-view aerial images with deep learning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 171:155–170, 2021. 3

[28] D. Yu, F. Ye, P. Yue, M. Chen, and F. Biljecki. Rooformer: Reconstructing detailed 3d roof models from high-resolution remote sensing imagery using transformer. *ISPRS Journal of Photogrammetry and Remote Sensing*, 227:745–758, 2025. 1

[29] F. Zhang, N. Nauata, and Y. Furukawa. Conv-mpn: Convolutional message passing neural network for structured outdoor architecture reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2798–2807, 2020. 1, 3

[30] Z. Zhang, M. Zhao, Z. Shen, Y. Wang, X. Jia, and D.-M. Yan. Interactive reverse engineering of cad models. *Computer Aided Geometric Design*, 111:102339, 2024. 8

[31] Q.-Y. Zhou and U. Neumann. 2.5 d building modeling by discovering global regularities. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 326–333. IEEE, 2012. 2

[32] X. Zhu, D. Du, W. Chen, Z. Zhao, Y. Nie, and X. Han. Nerve: Neural volumetric edges for parametric curve extraction from point cloud. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13601–13610, 2023. 3