remote sensing

MDPI

Article

Three-Dimensional Reconstruction of Structural Surface Model of Heritage Bridges Using UAV-Based Photogrammetric Point Clouds

Yue Pan [1], Yiqing Dong [1], Dalei Wang [1,*], Airong Chen [1] and Zhen Ye [2,†]

[1] College of Civil Engineering, Tongji University, Shanghai 200092, China; 1410001@tongji.edu.cn (Y.P.); dongyiqing@tongji.edu.cn (Y.D.); a.chen@tongji.edu.cn (A.C.)

[2] College of Surveying and Geo-Informatics, Tongji University, Shanghai 200092, China; 89_yezhen@tongji.edu.cn or z.ye@tum.de

* Correspondence: wangdalei@tongji.edu.cn; Tel.: +86-021-6598-1871

† Current address: Photogrammetry and Remote Sensing, Technische Universität München, Munich 80333, Germany

Received: 11 April 2019; Accepted: 16 May 2019; Published: 21 May 2019

Abstract: Three-dimensional (3D) digital technology is essential to the maintenance and monitoring of cultural heritage sites. In the field of bridge engineering, 3D models generated from point clouds of existing bridges is drawing increasing attention. Currently, the widespread use of the unmanned aerial vehicle (UAV) provides a practical solution for generating 3D point clouds as well as models, which can drastically reduce the manual effort and cost involved. In this study, we present a semi-automated framework for generating structural surface models of heritage bridges. To be specific, we propose to tackle this challenge via a novel top-down method for segmenting main bridge components, combined with rule-based classification, to produce labeled 3D models from UAV photogrammetric point clouds. The point clouds of the heritage bridge are generated from the captured UAV images through the structure-from-motion workflow. A segmentation method is developed based on the supervoxel structure and global graph optimization, which can effectively separate bridge components based on geometric features. Then, recognition by the use of a classification tree and bridge geometry is utilized to recognize different structural elements from the obtained segments. Finally, surface modeling is conducted to generate surface models of the recognized elements. Experiments using two bridges in China demonstrate the potential of the presented structural model reconstruction method using UAV photogrammetry and point cloud processing in 3D digital documentation of heritage bridges. By using given markers, the reconstruction error of point clouds can be as small as 0.4%. Moreover, the precision and recall of segmentation results using testing date are better than 0.8, and a recognition accuracy better than 0.8 is achieved.

Keywords: point cloud; heritage bridge; segmentation; recognition; unmanned aerial vehicle

1. Introduction

Ancient bridges in China are an essential part of the world's material and cultural heritage, but due to their age, archival information on ancient bridges is seriously scarce, and the number is sharply decreasing due to natural and human-made disasters. Therefore, each ancient bridge needs a fast, efficient, and accurate record for permanent preservation, particularly for endangered bridges.

Detailed three-dimensional (3D) reconstruction of ancient bridges provides a new way to solve this problem. This kind of work can be achieved using either range-based techniques, such as terrestrial laser scanners (TLS), or image-based techniques, mainly photogrammetry including structure-from-motion (SfM) [1,2]. Compared to the range-based approach, the image-based approach has several advantages, including easy acquisition and low costs. Recently, UAVs have been widely used to provide a general view for earth sensing reconnaissance and scientific data collection purposes [3,4]. Compared to terrestrial acquisition and the classic manned aerial acquisition, UAV imaging possesses the merits of flexibility and limited costs. These merits make these flying platforms an attracting choice for data acquisition in diverse applications such as 3D modeling [5,6], urban planning [7,8], natural environment mapping [9–11], and in general, several scene inspection and monitoring applications [12]. UAVs can acquire high-resolution and close-range images that can subsequently be processed using SfM and multiview stereopsis workflows (MVS) to generate detailed 3D point clouds and surface models. The emergence of ready-to-use consumer-grade drones as well as the development of photogrammetry and computer vision techniques further lead to significant research interest in 3D modeling and 3D mapping using UAV images in various applications, such as heritage building documentation [1], landslide monitoring [13], glacial geomorphology [14], building modeling [15] and change detection [16]. Among all these applications, the UAV-based 3D reconstruction. Currently, benefiting from photogrammetric triangulation/SfM techniques, 3D point clouds can be reconstructed from UAV images with satisfactory accuracy. In recent works like [17], with the help of bundle adjustment-based exterior orientation parameters, the accuracy of reconstructed points with consumer-grade drones can reach as high as 0.05 m, which is sufficient for the majority of engineering projects. However, to achieve such a high accuracy of reconstruction, two vital points should be considered and investigated. The first one is the relative orientation of images [18], which is essential to the rectification of the image. The quality of relative orientation will directly influence the success rate of extracting sparse corresponding points and the further rectification of images [19]. The second point is the dense matching technique [2,20], which is a critical step to generate dense point clouds from the orientated images. Only when problems involving these two points are solved can qualified point clouds be obtained with high accuracy for further applications. In the field of bridge engineering, several studies have successfully demonstrated the power of UAV photogrammetry for bridge inspection and maintenance. Khaloo et al. [21] and Morgenthal et al. [22] produced a 3D model of a bridge using UAV-captured images and a SfM algorithm to perform structural condition assessment. Chen et al. [23] proposed a bridge inspection process using point clouds generated from UAV images and compared this with TLS-based inspection. However, few published works in this area focus on the reconstruction of a structural 3D model of bridges. As the semantic and structural information is essential for heritage bridge documentation, automatic recognition of structural elements from the UAV photogrammetric bridge models using advanced point cloud processing methods is necessary.

However, to use 3D point clouds for further applications, we need to identify and separate individual objects from the entire scene [24]. To achieve this, segmentation of the point cloud is usually required before the recognition [25]. The segmentation of point clouds, aggregating 3D points into multiple homogeneous groups with common characteristics [26], has been explored for decades. Conventional segmentation methods like region growing or clustering examine points in the vicinity of initial seeds or origins and check if they belong to the same group or not according to given criteria. Euclidean distance [27], density [28,29], normal vector deviation [30,31], smoothness of surfaces [32], and curvatures [33] of points are representative criteria. Besides, the segmentation can also be conducted in the feature space. Geometric features or RGB colorinformation with distinctiveness are also introduced as segmentation criteria [34]. However, all these segmentation methods are easily influenced by noise and outliers in the dataset, resulting in over- or under-segmentation with different granularities of the obtained segments. Besides, complex segmentation criteria will significantly increase the computational cost. All these drawbacks should be solved if we want to achieve a good partition of objects from the entire scene.

In this paper, we attempt to bridge the gap between the engineering techniques of photogrammetry and remote sensing and applications from bridge engineering. To this end, an automatic framework for reconstructing structural surface models of heritage bridges is developed using UAV photogrammetry and point cloud processing. Detailed point clouds of a heritage bridge are generated from the captured UAV images through the SfM and MVS workflows. A novel segmentation method based on supervoxel structure and global graph optimization are proposed to separate the entire point cloud into consistent segments. Sequentially, the different structural elements are clustered using a classification based on a classification tree and bridge geometry, and the structural surface model is finally created by a Poisson surface reconstruction algorithm. To be specific, we utilize the graph-based segmentation method for the application of bridge engineering. The reason for using the graph-based method is that graph structure, which is a statistical context model, is commonly used for modeling the geospatial relationship between neighboring 2D/3D points. Since the graph structure has a natural relation with 3D topology, compared with other data structures (e.g., regular 3D grids for structured point clouds), a graphical model can encode not only the features of points in the local context but also the interactions between the point and its surrounding neighbors when constructing the weighted edges of the graph. The weights of edges encapsulate the dependency and affinity between connecting nodes. Using the optimization (e.g., partition) of the graphical model of the point cloud, the segmentation of the bridge point cloud can be easily achieved.The performance of the presented framework is validated via experiments with two bridges in China.

2. Methodology

The framework of 3D structural model generation is illustrated in Figure 1, which consists of five main steps, i.e., UAV flight path planning and image acquisition, image rectification, 3D reconstruction, point cloud segmentation, and structural elements recognition. The point cloud segmentation and structural elements recognition, which involves classifying different structural components of the bridge from the obtained point clouds, are the main contributions of our structural model generation framework. Details of these five steps are briefly presented in the following.

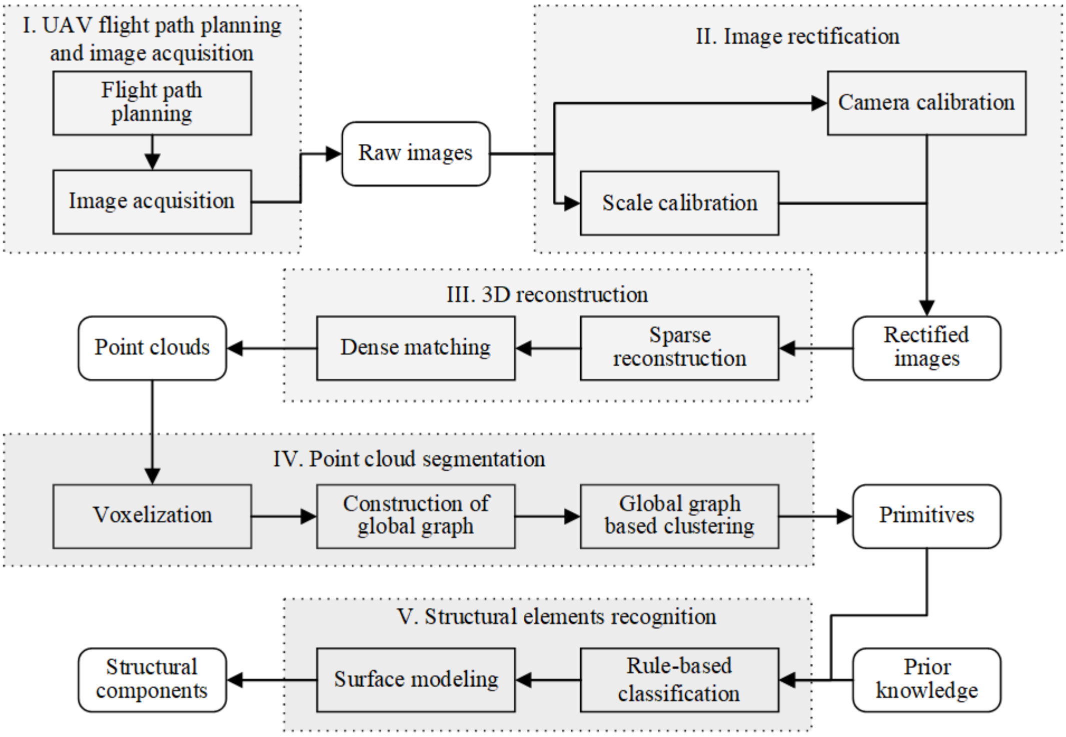


Figure 1. The entire workflow of the presented structural model generation framework.

2.1. UAV Flight Path Planning and Image Acquisition

Planning a flight path requires an awareness of multiple factors such as distance from target, speed, overlap, and pattern. According to the principle of photogrammetry and 3D reconstruction, higher information redundancy can improve the quality of the solution. In order to obtain better reconstruction results, the flight path should be properly planned before take-off to ensure a high overlap rate between images. According to the recommendation in [21], the following rules are initially determined:

- Shooting according to a serpentine route;
- The overlap ratio of the heading should be greater than 60%, and 90% is recommended;
- The side overlap ratio should be greater than 30%, and 60% is recommended.

The distance between the UAV and the bridge is determined by factors including camera field of view, sensor resolution, and safety. In this study, the distances from the bridge are divided into three scales: near, middle, and far, which are about 2 m, 5 m and 8 m, respectively. Taking the 2 m scale as an example, the flight path planning is shown in Figure 2.

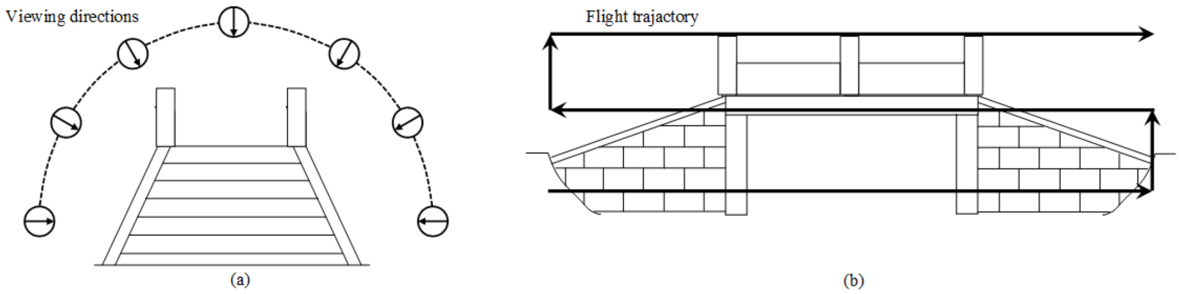


Figure 2. Planning graphs for the flight path. (**a**) Viewing directions along the flight direction. (**b**) Sketch of flight trajectory (side view).

2.2. Image Rectification

This step includes scale estimation and camera calibration. Since the 3D model normally obtained by SfM is initially generated in an arbitrary reference system, it is necessary to transform this initial arbitrary datum into a predefined coordinate reference system. For the purpose of heritage bridge digitization, the absolute position and orientation are not critical, but the absolute scale is indispensable. In this study, three pairs of mark points with the known distance are arranged on the bridge. As shown in Figure 3, Mark-1 and Mark-2 are located on the bridge deck, and Mark-3 is located on the bridge side railing. According to the ratio of the model distance and the actual distance between the mark points, the whole model can be synchronously scaled.

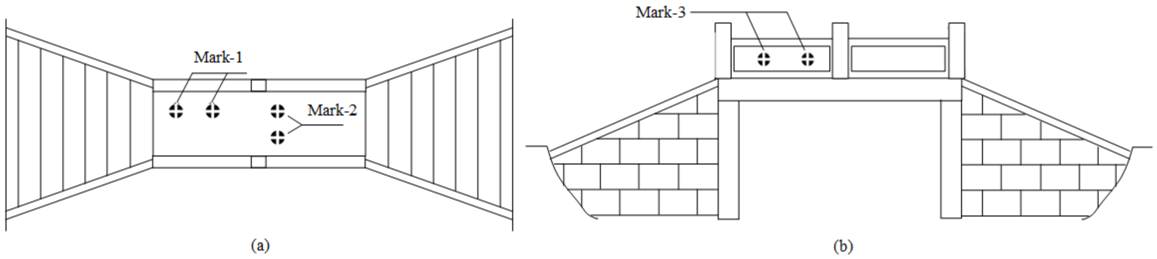


Figure 3. Locations of markers on the bridge.

Camera calibration is the process of estimating the internal geometry of a camera and has the most significant influence on the accuracy and reliability of photogrammetric measurements. Typically, there are two general strategies for calibration, i.e., pre-calibration and self-calibration. The influence of the calibration method on the quality of point clouds derived using UAV photogrammetry has been investigated in [35]. In this study, we pre-calibrate the camera shortly before the flight, by modeling the focal length, principal point coordinates, and distortion coefficients using Brown's distortion model [36]. The estimated interior camera parameters are used to undistort and thus correct the raw images prior to the following process. To be specific, the camera was calibrated immediately prior to capturing the surface of interest. The pre-calibration was realized using a plane-based calibration method implemented by EOS PhotoModeler Scanner software. The planar calibration pattern with circle targets and coded targets is supplied by PhotoModeler Scanner software, as show in Figure 4. It is a flexible, robust and low-cost method to obtain stable camera parameters.

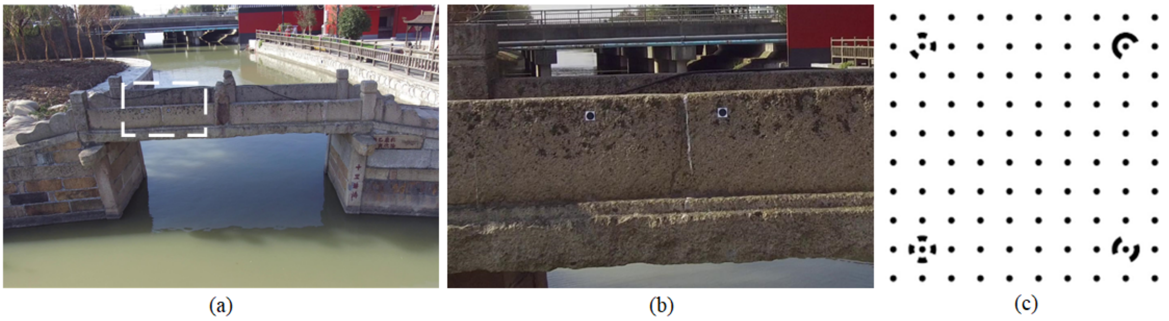


Figure 4. Camera calibration and markers on the UAV image. (**a**) Markers on the UAV image. (**b**) Zoomed in view of white box area in (**a**). (**c**) Calibration pattern used in PhotoModeler.

2.3. 3D Reconstruction

The 3D reconstruction to generate the photogrammetric point clouds is achieved via SfM and MVS workflows [13]. To be specific, this process includes two major steps, i.e., sparse reconstruction and dense matching [37]. The sparse reconstruction involves the identification and matching of homologous feature points as well as the reconstruction of the geometric image acquisition configuration and generation of the sparse point clouds. Dense matching is then conducted to increase the density of the 3D point cloud using the projection relationship recovered by sparse reconstruction. In this study, all processes are implemented with Agisoft PhotoScan software. A description of the SfM and MVS workflows in PhotoScan and commonly used parameters is provided in [38]. Moreover, a statistical outlier removal filtering (SOR) [39] is applied to these point clouds before the point cloud processing. The SOR filter we used is provided by the PCLlibrary. It computes first the average distance of each point to its neighbors, considering k-nearest neighbors for each. Then, it rejects the points that are farther than the average distance considering the standard deviation.

2.4. Point Cloud Segmentation

The segmentation step groups all the points into multiple consistent regions having one or several common characteristics [26], playing a vital role for separating the entire point cloud into meaningful geometric primitives in the presented framework. A novel segmentation method for point clouds based on the supervoxel structure and global graph optimization is proposed. The workflow of the proposed segmentation method is shown in Figure 5 and can be divided into three steps: supervoxelization, global graph construction, and graph-based clustering.

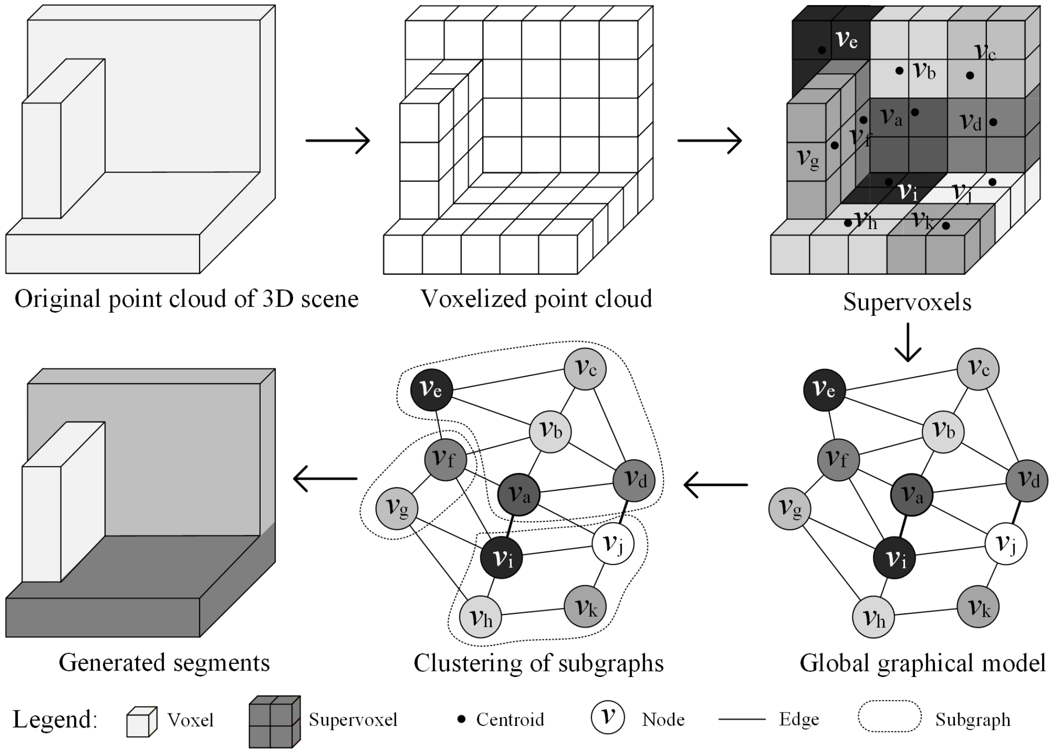


Figure 5. Workflow of the proposed segmentation method.

2.4.1. Supervoxelization of Point Clouds

For the supervoxelization of point clouds, we adopt the octree-based data structure to discretize the entire point cloud with 3D voxels, which allows the indexing of the unorganized point cloud with octree structure and simplifies the dataset with a grid-based representation [30]. As stated in [40], the selection of the voxel size should be carefully considered since we need to have a good balance between the processing time and the quality of preserved details.

Based on the voxel structure, supervoxels are further generated using the VCCS method [41], which groups candidate voxels according to their distance to seed points within a feature space comprising centroid positions, normal vectors, geometrical features, and RGB colors. The major advantage of VCCS is its ability to preserve boundaries, through which we can obtain supervoxels whose boundaries coincide with the edges of objects in the scene. For the performance of VCCS, parameters like the voxel size and the seed resolution will significantly influence the quality of segments. It is noteworthy that in this study, when generating the supervoxel, no color information is considered (i.e., only the spatial distance and normal vectors are used). The parameters are empirically set according to densities of points and distances between the sensor and objects within the scene.

2.4.2. Construction of Global Graph

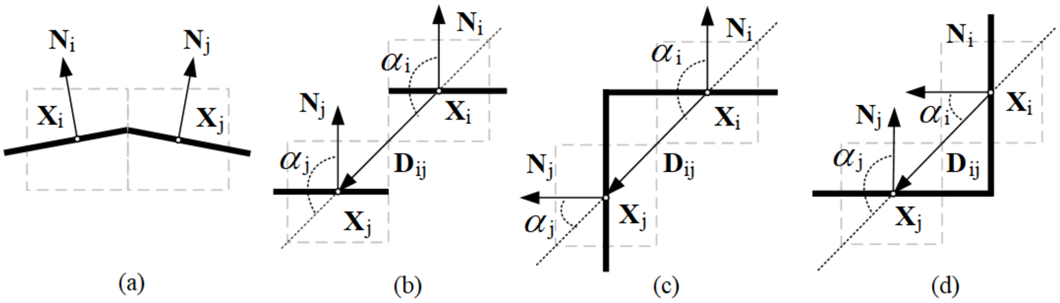
We use a supervoxel graph-based method of point cloud segmentation. The supervoxel structure of a point cloud can be represented by an undirected k-nearest neighbors graph $G = (V, E)$, in which the nodes represent the supervoxels and the edges encode their relationship. Each edge $(v_i, v_j) \in E$ has a corresponding weight w_{ij}, which is a non-negative measure of the similarity between neighboring supervoxels v_i and v_j. To identify the weight of the edge, each node v is assigned with an attribute vector representing geometric attributes formed by all points within v, relating to three groups of attributes: spatial positions, geometric features, and normal vectors [42]. To be specific, the spatial positions are the spatial coordinates $\mathbf{X}$ of the centroid p of supervoxel v. The geometric features are eigenvalue-based features representing the 3D distribution of points inside v, including linearity L_e, planarity P_e, scattering S_e, and change in curvature C_e [43]. The normal vector $\mathbf{N}$ is estimated using points within v. Based on the attributes of nodes, three types of geometric cues (i.e., proximity, similarity, continuity) representing binary features between neighboring supervoxels v_{i} and v_{j} are estimated. The proximity W_{ij}^{p} relates to the spatial distance between v_{i} and v_{j}. The similarity W_{ij}^{s} measures the consistency between geometric features of v_{i} and v_{j}. The continuity W_{ij}^{c} combines the smoothness W_{ij}^{m} and the convexity W_{ij}^{o} of supervoxel surfaces, and is estimated by normal vectors and direction vectors between v_{i} and v_{j} [40,44]. To judge the continuity, we assume four types of connected surfaces between supervoxels: smooth, "stair-like", convex, and concave. Sketches of these four types of connections are shown in Figure 6. The smoothness W_{ij}^{m} relates to the difference in angles between normal vectors $\mathbf{N}_{\mathrm{i}}$ and $\mathbf{N}_{\mathrm{j}}$. The convexity W_{ij}^{o} relies on the local configuration of the surfaces of adjacent supervoxels. If the local configuration is convex, adjacent supervoxels are considered to be highly connective. The local configuration (i.e., convex or concave) is judged by angles α_{i} and α_{j} (see Figure 6c,d), and is calculated by the angle between $\mathbf{N}_{\mathrm{i}}$ and $\mathbf{N}_{\mathrm{j}}$ and the vector $\mathbf{D}_{\mathrm{ij}}$ linking $\mathbf{X}_{\mathrm{i}}$ and $\mathbf{X}_{\mathrm{j}}$, where $\mathbf{D}_{\mathrm{ij}} = (\mathbf{X}_{\mathrm{j}} - \mathbf{X}_{\mathrm{i}})/||\mathbf{X}_{\mathrm{i}} - \mathbf{X}_{\mathrm{j}}||_2$. As stated in [44], if $\alpha_{\mathrm{i}} - \alpha_{\mathrm{j}} > \theta$, the surface connectivity is defined as a convex connection, and vice versa. Here θ is the threshold for judging convexity, which is a given threshold. The surface continuity W_{ij}^{c} is calculated according to Equation (1) [42], giving a higher continuity to convex or smooth connected surfaces:

$$W_{\mathrm{ij}}^{c} = \begin{cases} (\alpha_{\mathrm{i}} - \alpha_{\mathrm{j}})^2 + \pi^2 & , \ \alpha_{\mathrm{i}} > \alpha_{\mathrm{j}} + \theta \\ (\alpha_{\mathrm{i}} - \alpha_{\mathrm{j}})^2 + (\alpha_{\mathrm{i}} + \alpha_{\mathrm{j}} - \pi)^2 & , \ \alpha_{\mathrm{i}} \leq \alpha_{\mathrm{j}} + \theta. \end{cases} \tag{1}$$

Finally, the weight w_{ij} can be defined by considering all $W_{\mathrm{ij}}^{k}, k \in [\mathrm{p}, \mathrm{s}, \mathrm{c}]$ as:

$$w_{\mathrm{ij}} = \prod_{k \in [\mathrm{p,s,c}]} \exp\left(-\frac{(W_{\mathrm{ij}}^{k})^2}{2\lambda^2}\right), \tag{2}$$

where λ controls the weight of the spatial distance, the similarity, and the continuity. In this work, all of these three lambda parameters are set to one.

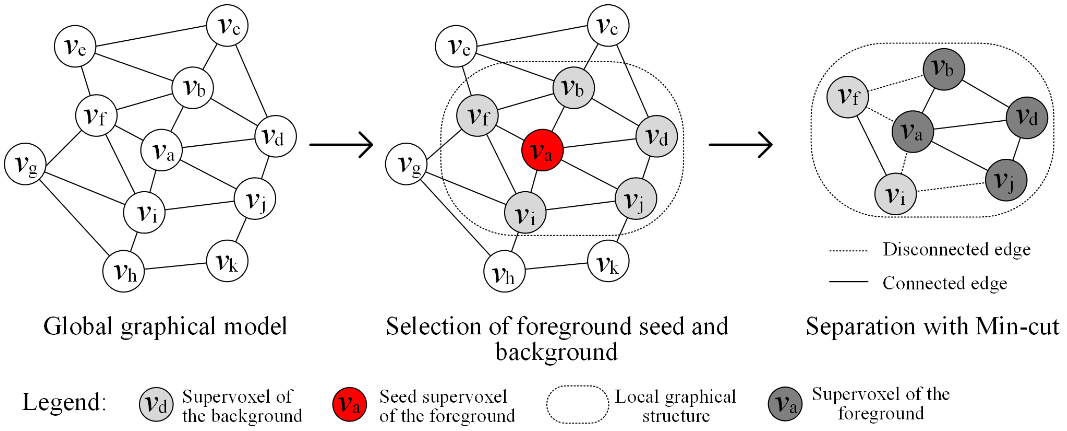


$\mathbf{N}_{\mathrm{i}}$
$\mathbf{N}_{\mathrm{j}}$
$\mathbf{X}_{\mathrm{i}}$
$\mathbf{X}_{\mathrm{j}}$
α_{i}
α_{j}
$\mathbf{D}_{\mathrm{ij}}$
(a)
(b)
(c)
(d)

Figure 6. Connection types between neighboring supervoxels [45]. (**a**) Smooth, (**b**) "stair-like", (**c**) convex, and (**d**) concave connections.

2.4.3. Global Graph-Based Clustering

To aggregate supervoxels into an entire segment, we formulate the aggregation process into a clustering work based on the graphical model. As stated in [46], the graphical model can explicitly represent 3D points with a mathematically sound structure and employs contextual information to deduce hidden information from given observations [47]. Thus, by constructing and partitioning the graphical model, we can obtain the connection information between supervoxels. To be specific, graph-based clustering aims to divide a dataset into disjointed subsets with members similar to each other from the affinity matrix. In [42], the use of the local graph structure for the description of the 3D geometry with the supervoxel structure was tested. The use of the local graph model can make the clustering process quite efficient and available for parallel computing when combined with a region-growing strategy. However, the local graph structure can merely encode the local geometry information, which can hardly represent the optimal in the global scale, so that over-segmentation frequently occurs when dealing with surfaces with irregular geometric shapes (e.g., points of vegetation). To tackle the drawbacks of the local graph model, we developed global graph-based clustering, which constructs a global graph model to describe the regional characteristics of 3D scenes with different complexities, and details of objects are preserved among the clustered nodes. Specifically, once the global graph of all supervoxels is constructed, we can optimize the connection between supervoxels by partitioning the global graph into several subgraphs. To solve this graphical model, we utilize the method introduced in [48], in which the Min-cut algorithm achieves a foreground–background separation. Different from the original work of [48], in our solution each supervoxel is regarded as the seed of the foreground and is compared with the background consisting of the local neighborhood. The foreground and background of the supervoxel builds a local graphical structure that is solved by the Min-cut algorithm. In Figure 7, we illustrate the built local graphical structure in the entire graphical model. Once the Min-cut algorithm is applied, the local graphical structure is separated into the foreground and background, and the edges linking these two parts are regarded as disconnected. When all the supervoxels have been checked, the entire global graphical model is separated into disconnected subgraphs by considering the connection of edges. For the supervoxels V in the same subgraph C, these are merged into a single segment S of points.



v_e
v_c
v_b
v_f
v_d
v_a
v_g
v_j
v_i
v_h
v_k
Disconnected edge
Connected edge
Legend:
Supervoxel of the background
Seed supervoxel of the foreground
Local graphical structure
Supervoxel of the foreground

Global graphical model Selection of foreground seed and background Separation with Min-cut

Figure 7. Node clustering in the graphical model.

2.5. Structural Element Recognition

Once the segments are achieved, a rule-based classification method is applied to recognize different structural elements from segments. Here, the recognition step consists of the refinement of segments, the recognition of elements, and surface modeling.

2.5.1. Refinement of Segments

The refinement of obtained segments is needed for the supervoxel-based segmentation, the results of which always suffer from the "zig-zag" effect since the basic element of segments is the cubic-shaped voxel [49]. To overcome this drawback, we propose a boundary refinement of the achieved segment, consisting of two major steps: the detection of points of boundary supervoxels and the refinement of these boundary points. In the first step, for each segment containing several supervoxels, if one of the supervoxels is adjacent to the supervoxels of other segments, this supervoxel will be identified as the boundary one, and all of its points will be regarded as boundary points. Then, in the second step, the normal vectors of the boundary points and the supervoxel at the boundary are estimated. Based on the estimated normal vectors, a local k-mean clustering is conducted between the boundary point and the centers of neighboring supervoxels (see Figure 8).

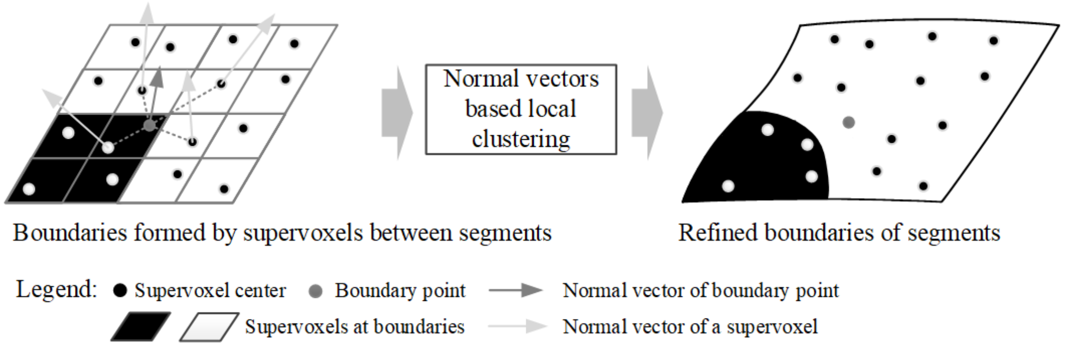

Normal vectors based local clustering
Boundaries formed by supervoxels between segments
Refined boundaries of segments
Legend:
● Supervoxel center
● Boundary point
Normal vector of boundary point
Supervoxels at boundaries
Normal vector of a supervoxel

Figure 8. Refinement of boundary points of segments.

Here, the clustering is governed by a distance measure calculated in the feature space, considering the normal vectors **N** and the spatial distance between centroids **X**:

$$D = \sqrt{w_\mathrm{n}\frac{\mathbf{N}_\mathrm{i}\cdot\mathbf{N}_\mathrm{b}}{|\mathbf{N}_\mathrm{i}||\mathbf{N}_\mathrm{b}|} + w_\mathrm{d}(|\mathbf{X}_\mathrm{i}| - |\mathbf{X}_\mathrm{b}|)}, \tag{3}$$

where w_n and w_d are weight factors controlling the contribution of normal vectors and spatial distances.

2.5.2. Recognition of Elements

For recognizing structural elements from the segments, a rule-based classification method is introduced that utilizes the saliency of segments to determine the label of segments (i.e., the type of object that it belongs to) [50]. The saliency of the segment is defined by its geometric properties, which include the following aspects:

- The height S_h indicating the spatial position;
- The angle S_v between the horizontal direction and the normal vector of the segment;
- The size S_s of segments relating to the spatial length, width, and height.

To be specific, the saliency for a given segment S_sal is a vector that is computed as follows:

$$S_\mathrm{sal} = [\frac{S_\mathrm{h}}{\max_{k=1,\ldots,n}(S_\mathrm{h}^k)}, (1 - \frac{S_\mathrm{s}}{\max_{k=1,\ldots,n}(S_\mathrm{s}^k)}), \frac{S_\mathrm{v}}{\pi/2}], \tag{4}$$

where n is the total number of segments in the point clouds. The saliency of each segment is ranked in decreasing order as the input to the decision for recognizing structural elements. In this work, three types of structural elements are considered, namely decks, fences, and walls of the bases. Once the saliency of all of the segments is calculated, a sequential classification is applied with given thresholds in a classification tree (see Figure 9). The segments with saliency values falling into particular branches are recognized as particular structural elements.

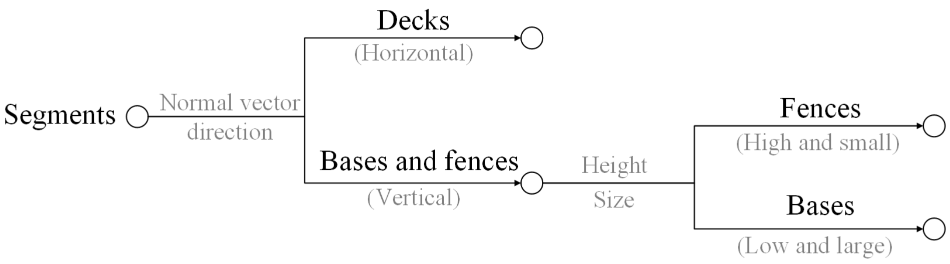


Figure 9. Classification tree for recognition.

2.5.3. Surface Modeling

A Poisson surface reconstruction algorithm [51] is adopted for surface modeling. The input for Poisson surface reconstruction is the point cloud and its normal vector. The point represents the position of the object surface, and its normal vector stands for the direction. Surface reconstruction is achieved using an implicit function framework, which computes a 3D indicator function and considers all of the points at once. Considering a region M with its boundary δM and an indicator function $\mathcal{X}_M$, for any point $p \in \delta M$, we define $\vec{N}_{\delta M}(p)$ as the inward surface normal vector. $\tilde{F}$ is a Gaussian smoothing filter, and $\tilde{F}_p(q) = \tilde{F}(q-p)$ is a translation of $\tilde{F}$ to the point p. As $\mathcal{X}_M$ is generally not well-derived, the available gradient of $\mathcal{X}_M * \tilde{F}$ is approximated as

$$\bigtriangledown (\mathcal{X}_M * \tilde{F})(q_0) = \int_{\delta M} \tilde{F}_p(q_0)\vec{N}_{\delta M}(p)dp. \tag{5}$$

The indicator function can be calculated by Equation (5), and the reconstructed surface M is obtained by extracting an appropriate isosurface. Finally, a 3D mesh model with vertices and faces is achieved.

3. Results and Discussion

In the experiments section, we first introduce our bridge testing data, with two bridge field sites introduced. The generation of point clouds for these two bridges is also stated. Then, experiments using benchmarks are carried out to assess the performance of our proposed segmentation methods. Field tests using point clouds of the two bridges mentioned above are conducted, and the corresponding results are analyzed and discussed.

3.1. Testing Data of Bridges

To test the performance of the presented framework, two bridges were selected as testing sites. The first one is the Hongde Bridge, located in Hongsan Village, Tang Town, Shanghai, China. It was built in the Qianlong period of the Qing dynasty and has been around for more than 250 years. It is a single-span stone bridge with a span of 4.9 m, a bridge length of 14.5 m, and a bridge deck width of 2 m. The second one, the Tongxin Bridge, is located at Tongji University, Shanghai, China. It is a brick double-arch bridge, that was built over 50 years ago, with a length of 9.5 m and a bridge deck width of 5 m. In Figure 10, views of the testing bridges are shown.

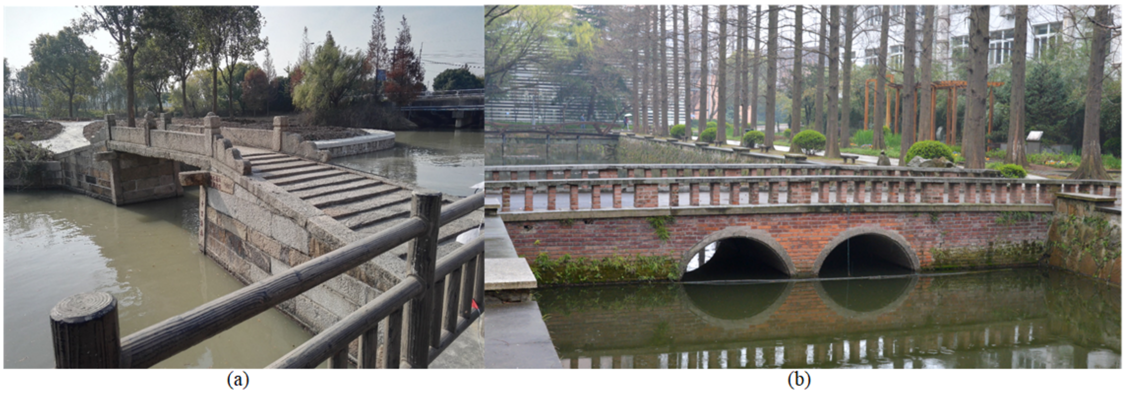

(a) (b)

Figure 10. Testing bridges. (**a**) Hongde Bridge. (**b**) Tongxin Bridge.

The UAV used was a DJI Phantom 3 Professional, which is a consumer-grade quadcopter. A total of 763 and 575 images, respectively, were collected with a $1/2.3''$ complementary metal oxide semiconductor sensor with 12 megapixels, for each of the two bridges. The camera was mounted on a three-axis stabilization gimbal. Although more advanced UAVs and cameras are available, the purpose of this flight was to demonstrate the presented framework for bridge digitization in a real scenario.

3.2. Generated Point Clouds

According to the matching results of the feature points and the orientation parameters, point clouds of the Hongde Bridge and Tongxin Bridge were generated, as shown in Figure 11a–d. It is clear that the sparse point cloud reflects the main contour of the Hongde Bridge, including the outer edges of the bridge deck, the pier, the stone lion, and so on. The number of sparse points was about 300,000. Then, using the results of MVS, all of the pixels were re-projected to obtain the dense point cloud model shown in Figure 11b,d. It can be seen that the overall structure and texture of the bridges was fully restored. The point number of the final point cloud is around 18 million. The point cloud of the Tongxin Bridge was generated in the same way. The number of points in the sparse point cloud was 37,000, which is much smaller than that of the Hongde Bridge. For the dense point cloud, there was a total of 2.5 million points generated. The root mean square (RMS) of reprojection errors was 0.715 pixels and 1.010 pixels for the point clouds of the Hongde Bridge and the Tongxin Bridge, respectively.

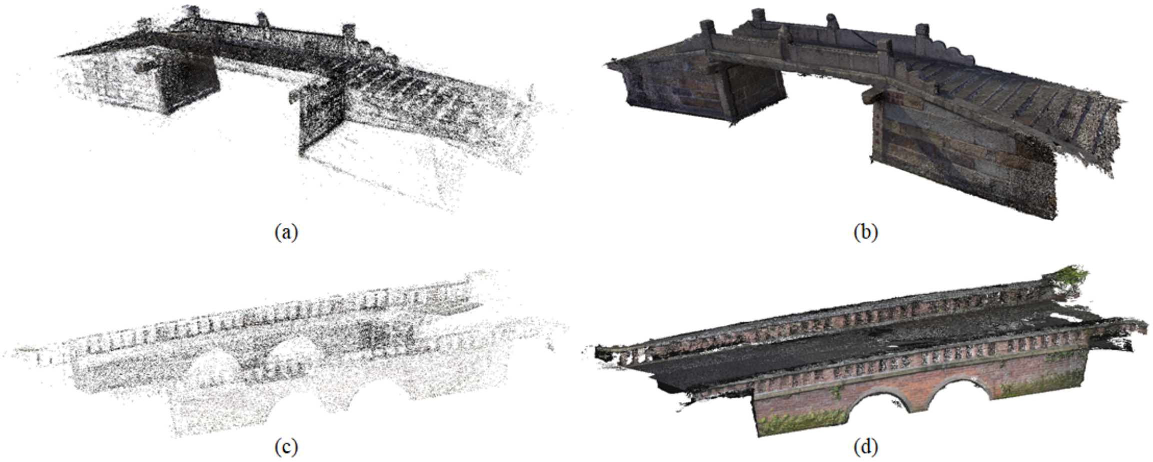


Figure 11. Generated point clouds of the Hongde Bridge and the Tongxin Bridge. (**a**,**c**) Sparse point cloud. (**b**,**d**) Dense point cloud.

To evaluate the quality of the point cloud reconstructions, in the point clouds of the Hongde Bridge, we measured the distance between three pairs of marker and compared them with the ground truth distance. The result is given in Table 1. As seen from the table, we find that when using all the UAV images from different scales, the error of reconstruction can be as small as 0.4%, which could be regarded as sufficient for the documentation of heritage bridges.

Table 1. Reconstruction accuracy of different scales.

Scale	UAV Distance to Bridge (m)	Measured Distance (m)	Ground Truth Distance (m)	Error
Small	2.0	0.504	0.5	0.8%
Middle	5.0	0.515	0.5	3.0%
Large	8.0	0.523	0.5	4.6%
All	–	0.502	0.5	0.4%

3.3. Quality of Segmentation

The performance of the segmentation plays an essential role in the recognition of structural elements. For this reason, we assessed the segmentation method by conducting a comparison between the achieved segments and the manually generated segments from reference data. For the reference dataset, we utilized the one used in [42], which is a manually segmented point cloud of a single building. The results generated by three point- or voxel-based segmentation algorithms, namely the smoothness-based Region Growing (RG) [32], Supervoxel- and Graph-based Segmentation (SVGS) [42], and Locally Convex Connected Patches (LCCP) [44] algorithms, were used as baselines for further comparison.

In these experiments, the voxel size (used in our method, SVGS, and LCCP) and the neighborhood size (for estimating normal vectors) of RG were set to 0.15 m, ensuring that the normal vectors used in all the methods were calculated by almost the same number of points. For the supervoxelization process used in our segmentation method, SVGS, and LCCP, the seed resolution of VCCS was set to 0.5 m. The threshold for tolerating the angle difference between normal vectors was set to 0.26 rad, empirically. For SVGS, the threshold for efficient graph-based segmentation was 0.75.

The *precision* and *recall* were selected as the basic evaluation metrics for assessing the performance of our method, which are calculated via Equations (6) and (7), by the use of true positive (TP), true negative (TN), false positive (FP), and false negative (FN). Here, *precision* is used to estimate the ratio of correctly segmented points in the segmentation results, while *recall* is used to assess the ration of correctly segmented points in the reference data [30].

$$precision = \frac{|TP|}{|TP| + |FP|}. \tag{6}$$

$$recall = \frac{|TP|}{|TP| + |FN|}. \tag{7}$$

In Figure 12, the segmentation results using different baseline algorithms is given. It is noted that to handle outliers and isolated points, we removed the supervoxels that had too few points, namely, those that did not satisfy the minimum number (at least three points) of required points when estimating the normal vector. As seen from the figure, it is clear that our segmentation method can generate outstanding results. To be specific, not only are planar surfaces (e.g., roofs and walls) well separated, but irregular shapes (e.g., bushes and fences) are also segmented. By contrast, the results of LCCP are more like over-segmentation, with small patches appearing at edges. With RG, the areas of planes are well segmented, but for curved surfaces and irregular objects, they are more likely to be over-segmented as well. As for the results of SVGS, irregular objects like bushes are well segmented, but it seems that under-segmentation frequently happens for nearby objects.

To thoroughly investigate the potential of our segmentation method, we created the precision–recall (PR) curves (see Figure 13) for all baseline algorithms by changing thresholds based on the reference data. The tendencies of the PR curves show that our method can achieve better segmentation and a good compromise between accuracy and recall. Specifically, according to the PR curve, we find that when the recall value is greater than 0.7, the proposed method has better performance than other methods. However, since the RG method is a point-based method that tends to generate over-segmentation, the results of the RG method can achieve better precision values than our method, but with smaller recall values. For the results of the LCCP method, since the smoothness and convexity criteria used in the LCCP method are more suitable to segment planes and box-shaped structures [40], when dealing with rough and irregular-shaped surfaces (e.g., roofs) and linear structures (e.g., fences), we can observe a similar over-segmentation like that of RG, showing discontinuous patches of segments. In this work, the size of voxels is determined according to the demands of application, namely the subdivision of the octree is stopped according to the divided size of voxels. In each voxel, there should remain at least three points to estimate the eigenvectors. In Figure 14, we illustrate the relation between the voxel sizes and the segmentation performance. Here, the segmentation result is evaluated via the F_1 score. In this test, with a fixed threshold for partitioning the graphical model, the voxel size ranges from 0.1 m to 1.0 m, and the seed resolution of corresponding supervoxels is three times larger than the voxel size. The test results shows that with appropriate voxel sizes (i.e., 0.2 m–0.4 m), our segmentation method can achieve good results, with F_1 scores ranging from around 0.8 to 0.67.

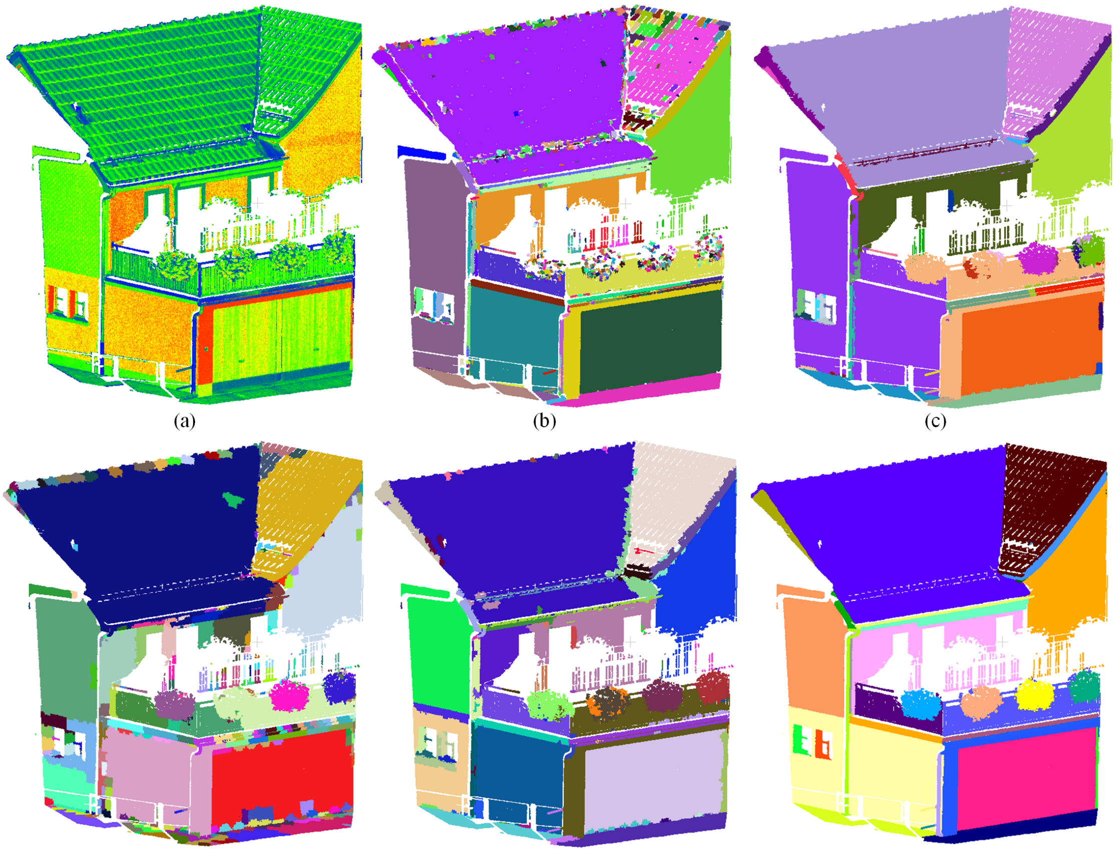

(a) (b) (c)
(d) (e) (f)

Figure 12. Segmentation results using the reference data. (**a**) Original point cloud. (**b**) Result of the Region Growing (RG) algorithm. (**c**) Result of the Supervoxel- and Graph-based Segmentation (SVGS) algorithm. (**d**) Result of the Locally Convex Connected Patches (LCCP) algorithm. (**e**) Result using our method. (**f**) Ground truth.

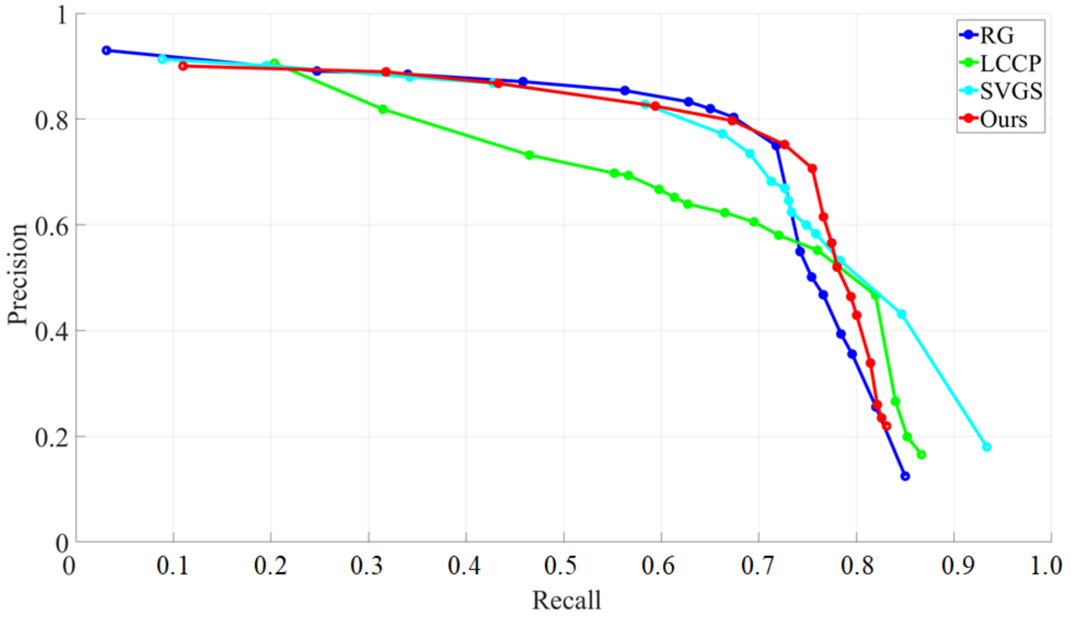


Figure 13. Precision–recall (PR) curves of segmentation results.

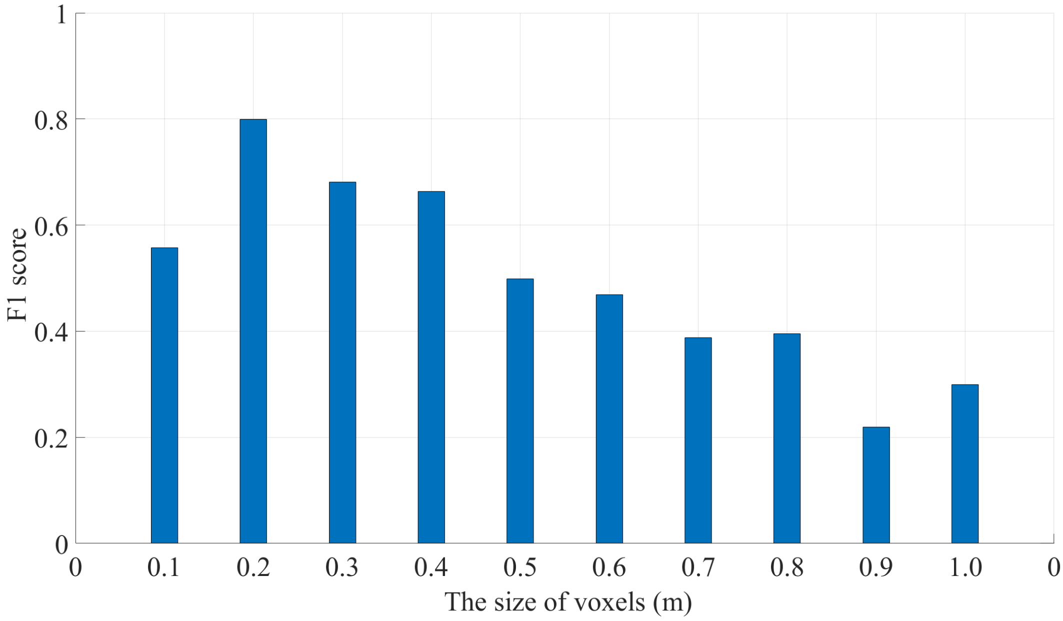


Figure 14. The influence of different voxel sizes on the segmentation results.

The segmentation results of point clouds of real bridges are given in Figures 15 and 16, while the number of generated segments is given in Table 2. We find that the major components of the bridges are well separated from the entire point cloud. However, for the boundaries of the obtained segments, there are still some errors, especially for the area without a good quality of points (e.g., the points are too sparse or include too many outliers). In some area, over-segmentation also happens, for example, in Figure 15b, the wall on the base of the bridge is over-segmented into two parts. Moreover, for some connection areas between the decks and fences, the edges are not clear and incorrect. However, for the majority of the segments, the edges are satisfactory.

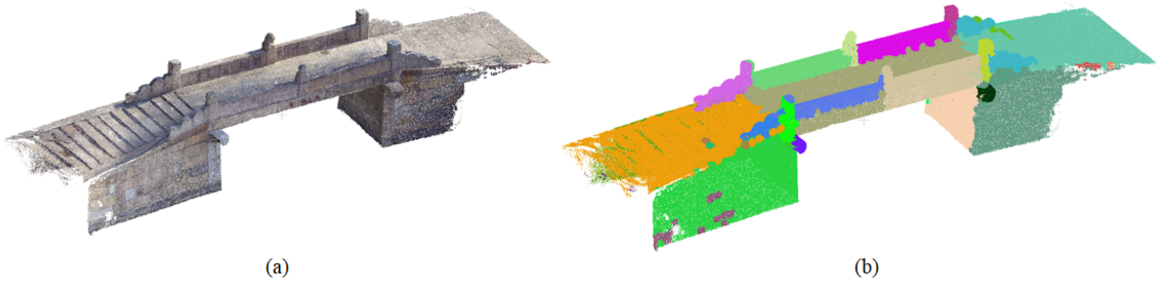

(a) (b)

Figure 15. Segmentation results for the Hongde Bridge. (**a**) Original point cloud. (**b**) Segmentation results.

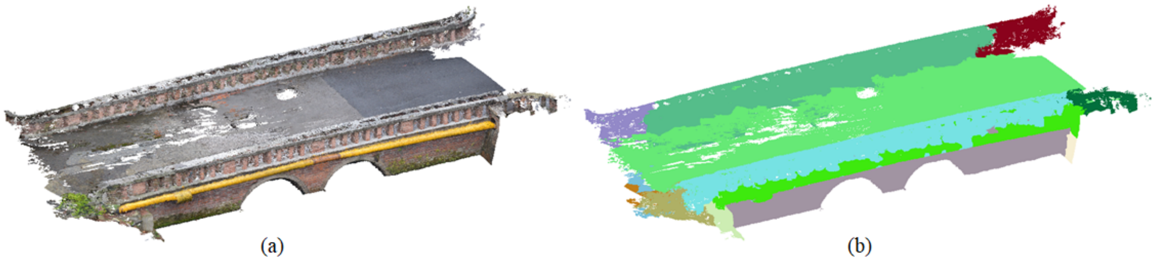

(a) (b)

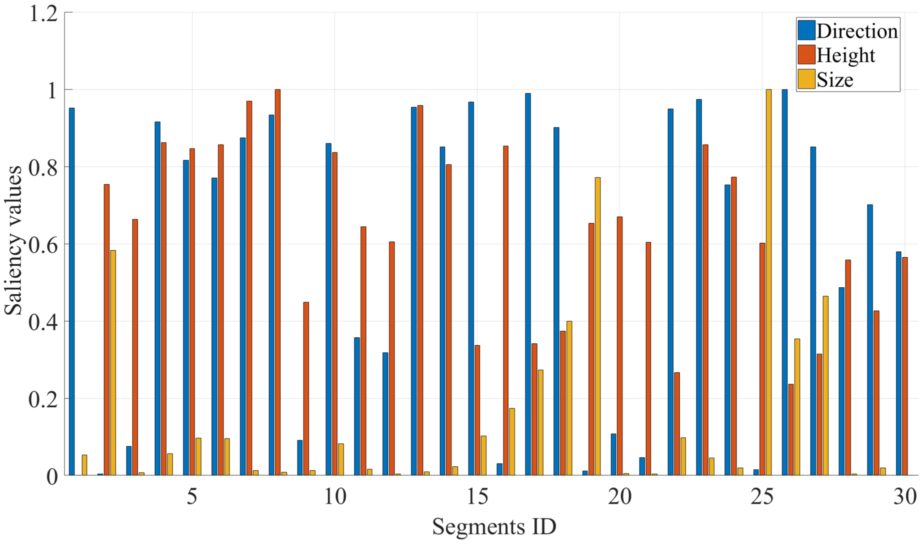
Figure 16. Segmentation results for the Tongxin Bridge. (**a**) Original point cloud. (**b**) Segmentation results.

Table 2. Segmentation results for the Hongde Bridge and Tongxin Bridge.

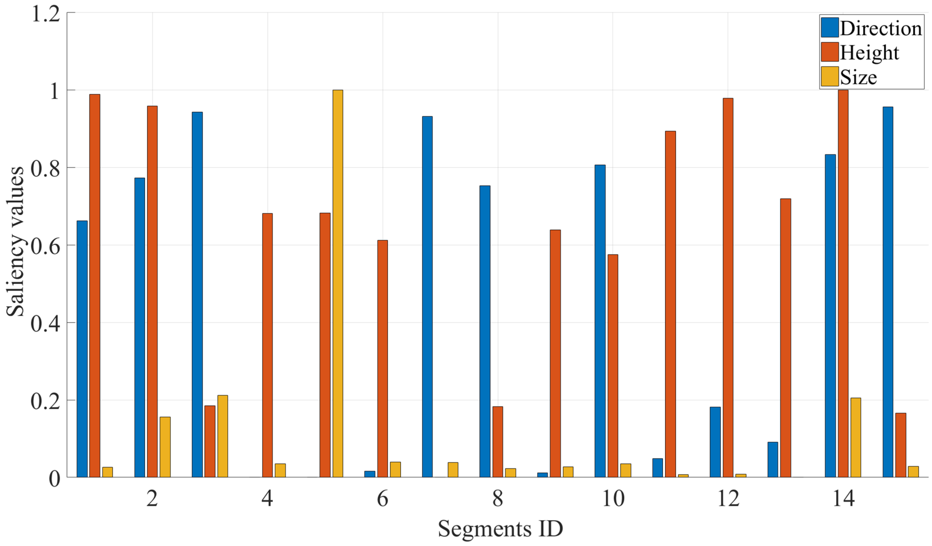
Name	Number of Points	Number of Segments
Hongde Bridge	18,114,743	30
Tongxin Bridge	5,613,422	15

3.4. Recognition and Modeling of Structural Elements

Once the segmentation of the point cloud is achieved. The segments are classified into three groups by the saliency values. In Figure 17, the saliency values of segments from two different datasets are given. As seen from these saliency values, it is evident that segments of different structural elements will generate significantly different saliency values; for example, the decks and bases of the bridge will have totally different height and direction values.



(a)



(b)

Figure 17. Segment saliency of the (**a**) Hongde Bridge and (**b**) Tongxin Bridge.

Based on these saliency values, the classification tree mentioned above was applied to group the segments. The results are shown in Table 3 and the corresponding saliency values of each segment are shown in Figure 18. Concerning the saliency values, we find that the height and direction values dominate the distinguishing process. As seen from the recognition results, it is clear that the majority of the major structures have been correctly recognized, and especially large elements. However, small patches belonging to the decorations are occasionally recognized incorrectly. To be specific, for Hongde Bridge, there are eight segments recognized as decks, while for Tongxin Bridge, only two segments are recognized as decks. For the bases, there are eight and four segments recognized as bases of the two bridges, respectively. For the rest of the segments, they are all regarded as parts of fences. Nevertheless, there are still some segments of the fences that are recognized incorrectly as parts of the base, which can be seen in Figure 19b. In Table 4, we also give an evaluation of the recognition results compared with the ground truth from manual recognition. The evaluation result reveals that our recognition method can reach an overall accuracy (OA) greater than 0.8 for both bridges.

Table 3. Recognition results for Hongde Bridge and Tongxin Bridge.

Name	Total Segments	Segments of Decks	Segments of Bases	Segments of Fences
Hongde Bridge	30	8	8	14
Tongxin Bridge	15	2	4	9

Table 4. Evaluation of the recognition result.

	Hongde Bridge			Tongxin Bridge		
Truth \ Predict	Decks	Bases	Fences	Decks	Bases	Fences
Decks	6	0	2	2	0	0
Bases	1	8	1	0	3	2
Fences	1	0	11	0	1	7
Overall accuracy (OA)		0.83			0.8	

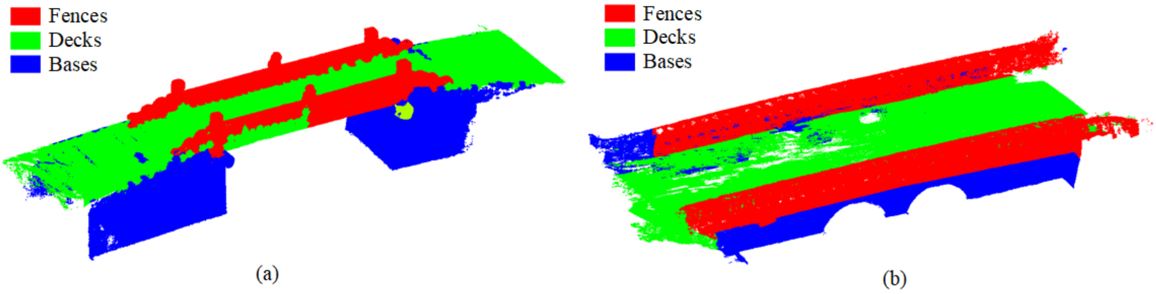


Figure 18. Recognition results for (**a**) Hongde Bridge and (**b**) Tongxin Bridge.

In Figure 19, we illustrate the modeling results of the recognized structural elements. As seen from the figure, it is apparent that the quality of segments significantly influences the modeling quality. For segments with clear and accurate boundaries, the surfaces are well reconstructed. The generated surface models provide a comprehensive documentation of these historical bridges, with the size, the orientation, and the geometric properties estimated and recorded. However, we should also note that in this work, we only categorize the structural elements of the bridge into three types, namely decks, fences, and bases, which is not sufficient to fully describe the 3D information of historical bridges.

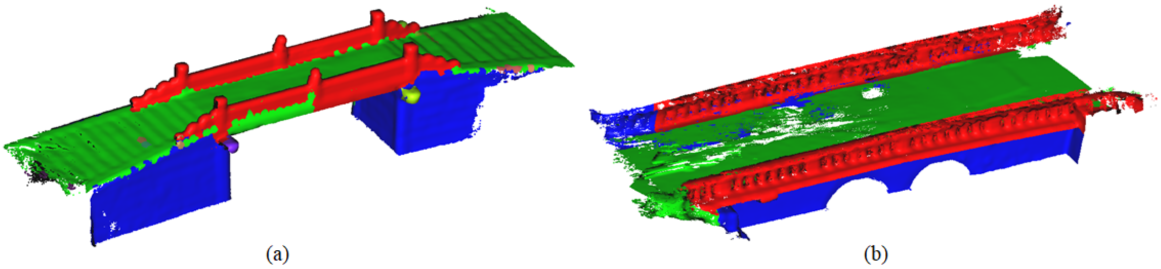


Figure 19. Surface modeling results for (**a**) Hongde Bridge and (**b**) Tongxin Bridge.

4. Conclusions

In this work, we present an automatic method for generating surface models of structural elements of heritage bridges. To be specific, we tackle this challenge via a novel top-down method for segmenting main bridge components, combined with rule-based classification, to generate labeled models from point clouds. This study focuses on using UAV photogrammetric point clouds associated with heritage bridges in China. The segmentation method is developed based on voxel structure and global graph optimization, which can effectively separate bridge components based on geometric features. Then, classification based on a classification tree and bridge geometry is utilized to recognize different structural elements from the obtained segments. Finally, surface modeling is conducted to generate surface models of the recognized segments. Experiments using Hongde Bridge and Tongxin Bridge demonstrate the potential for using UAV photogrammetry in 3D digital documentation of cultural heritage sites, with promising results achieved. To be specific, primary structural elements are recognized from 30 and 15 segments of the two bridges, respectively. Surface models are also created based on the recognized segments, which can provide solid documentation for the further preservation of these bridges. Moreover, experimental results using benchmark datasets also reveal that our proposed segmentation algorithm is promising for the separation of structural elements in the field of civil engineering. In the future, we will focus on optimizing 3D reconstruction algorithms for more complicated bridge structures and developing better noise removal techniques. In addition, the improvement of the quality of UAV-based point clouds should also be considered, which could be helped by acquiring better position and orientation information based on internal (GPS) and inertial measurement units systems.

Author Contributions: Conceptualization, Y.P.; Data curation, Y.D.; Formal analysis, Z.Y.; Funding acquisition, D.W.; Investigation, Y.P.; Methodology, Y.P.; Project administration, D.W. and A.C.; Software, Y.D. and Z.Y.; Supervision, D.W. and A.C.; Visualization, Y.D.; Writing—original draft, Y.P. and Z.Y.

Funding: This research was funded by the National Natural Science Foundation of China (Grant No.:51778472) and the Jiangsu Provincial Department of Transport (Grant No.: GCJS2018-89).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Murtiyoso, A.; Grussenmeyer, P. Documentation of heritage buildings using close-range UAV images: Dense matching issues, comparison and case studies. *Photogramm. Rec.* **2017**, *32*, 206–229. [CrossRef]
2. Jiang, S.; Jiang, W. Efficient sfm for oblique uav images: From match pair selection to geometrical verification. *Remote. Sens.* **2018**, *10*, 1246. [CrossRef]
3. Watts, A.C.; Ambrosia, V.G.; Hinkley, E.A. Unmanned aircraft systems in remote sensing and scientific research: Classification and considerations of use. *Remote. Sens.* **2012**, *4*, 1671–1692. [CrossRef]
4. Colomina, I.; Molina, P. Unmanned aerial systems for photogrammetry and remote sensing: A review. *ISPRS J. Photogramm. Remote. Sens.* **2014**, *92*, 79–97. [CrossRef]
5. Nex, F.; Remondino, F. UAV for 3D mapping applications: A review. *Appl. Geomat.* **2014**, *6*, 1–15. [CrossRef]

6. Fernández-Hernandez, J.; González-Aguilera, D.; Rodríguez-Gonzálvez, P.; Mancera-Taboada, J. Image-based modelling from unmanned aerial vehicle (UAV) photogrammetry: An effective, low-cost tool for archaeological applications. *Archaeometry* **2015**, *57*, 128–145. [CrossRef]
7. Liu, S.; Tong, X.; Chen, J.; Liu, X.; Sun, W.; Xie, H.; Chen, P.; Jin, Y.; Ye, Z. A Linear Feature-Based Approach for the Registration of Unmanned Aerial Vehicle Remotely-Sensed Images and Airborne LiDAR Data. *Remote. Sens.* **2016**, *8*, 82. [CrossRef]
8. Gevaert, C.; Sliuzas, R.; Persello, C.; Vosselman, G. Evaluating the societal impact of using drones to support urban upgrading projects. *ISPRS Int. J. -Geo-Inf.* **2018**, *7*, 91. [CrossRef]
9. Bhardwaj, A.; Sam, L.; Martín-Torres, F.J.; Kumar, R. UAVs as remote sensing platform in glaciology: Present applications and future prospects. *Remote. Sens. Environ.* **2016**, *175*, 196–204. [CrossRef]
10. Lottes, P.; Khanna, R.; Pfeifer, J.; Siegwart, R.; Stachniss, C. UAV-based crop and weed classification for smart farming. In Proceedings of the 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 29 May–3 June 2017; pp. 3024–3031.
11. Manfreda, S.; McCabe, M.; Miller, P.; Lucas, R.; Pajuelo Madrigal, V.; Mallinis, G.; Ben Dor, E.; Helman, D.; Estes, L.; Ciraolo, G.; et al. On the use of unmanned aerial systems for environmental monitoring. *Remote. Sens.* **2018**, *10*, 641. [CrossRef]
12. Chen, P.; Dang, Y.; Liang, R.; Zhu, W.; He, X. Real-time object tracking on a drone with multi-inertial sensing data. *IEEE Trans. Intell. Transp. Syst.* **2018**, *19*, 131–139. [CrossRef]
13. Lucieer, A.; Jong, S.M.d.; Turner, D. Mapping landslide displacements using Structure from Motion (SfM) and image correlation of multi-temporal UAV photography. *Prog. Phys. Geogr.* **2014**, *38*, 97–116. [CrossRef]
14. Ewertowski, M.; Tomczyk, A.; Evans, D.; Roberts, D.; Ewertowski, W. Operational Framework for Rapid, Very-high Resolution Mapping of Glacial Geomorphology Using Low-cost Unmanned Aerial Vehicles and Structure-from-Motion Approach. *Remote. Sens.* **2019**, *11*, 65. [CrossRef]
15. Rakha, T.; Gorodetsky, A. Review of Unmanned Aerial System (UAS) applications in the built environment: Towards automated building inspection procedures using drones. *Autom. Constr.* **2018**, *93*, 252–264. [CrossRef]
16. Kohv, M.; Sepp, E.; Vammus, L. Assessing multitemporal water-level changes with uav-based photogrammetry. *Photogramm. Rec.* **2017**, *32*, 424–442. [CrossRef]
17. He, F.; Zhou, T.; Xiong, W.; Hasheminnasab, S.; Habib, A. Automated aerial triangulation for UAV-based mapping. *Remote. Sens.* **2018**, *10*, 1952. [CrossRef]
18. He, F.; Habib, A. Automated relative orientation of UAV-based imagery in the presence of prior information for the flight trajectory. *Photogramm. Eng. Remote. Sens.* **2016**, *82*, 879–891. [CrossRef]
19. Habib, A.; Xiong, W.; He, F.; Yang, H.L.; Crawford, M. Improving orthorectification of UAV-based push-broom scanner imagery using derived orthophotos from frame cameras. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* **2017**, *10*, 262–276. [CrossRef]
20. Ye, Z.; Tong, X.; Zheng, S.; Guo, C.; Gao, S.; Liu, S.; Xu, X.; Jin, Y.; Xie, H.; Liu, S.; et al. Illumination-Robust Subpixel Fourier-Based Image Correlation Methods Based on Phase Congruency. *IEEE Trans. Geosci. Remote. Sens.* **2018**, 1995–2008. [CrossRef]
21. Khaloo, A.; Lattanzi, D.; Cunningham, K.; Dell'Andrea, R.; Riley, M. Unmanned aerial vehicle inspection of the Placer River Trail Bridge through image-based 3D modelling. *Struct. Infrastruct. Eng.* **2018**, *14*, 124–136. [CrossRef]
22. Morgenthal, G.; Hallermann, N.; Kersten, J.; Taraben, J.; Debus, P.; Helmrich, M.; Rodehorst, V. Framework for automated UAS-based structural condition assessment of bridges. *Autom. Constr.* **2019**, *97*, 77–95. [CrossRef]
23. Chen, S.; Laefer, D.F.; Mangina, E.; Zolanvari, S.I.; Byrne, J. UAV Bridge Inspection through Evaluated 3D Reconstructions. *J. Bridge Eng.* **2019**, *24*, 05019001. [CrossRef]
24. Xu, Y.; Tuttas, S.; Heogner, L.; Stilla, U. Reconstruction of scaffolds from a photogrammetric point cloud of construction sites using a novel 3D local feature descriptor. *Autom. Constr.* **2018**, *85*. doi:10.1016/j.autcon.2017.09.014. [CrossRef]
25. Xu, Y.; Tuttas, S.; Hoegner, L.; Stilla, U. Geometric Primitive Extraction From Point Clouds of Construction Sites Using VGS. *IEEE Geosci. Remote. Sens. Lett.* **2017**, *14*, 424–428. doi:10.1109/LGRS.2017.2647816. [CrossRef]

26. Grilli, E.; Menna, F.; Remondino, F. A review of point clouds segmentation and classification algorithms. *ISPRS Int. Arch. Photogramm. Remote. Sens. Spat. Inf. Sci.* **2017**, *XLII-2/W3*, 339–344. doi:10.5194/isprs-archives-xlii-2-w3-339-2017. [CrossRef]
27. Aldoma, A.; Marton, Z.C.; Tombari, F.; Wohlkinger, W.; Potthast, C.; Zeisl, B.; Rusu, R.; Gedikli, S.; Vincze, M. Tutorial: Point cloud library: Three-dimensional object recognition and 6 dof pose estimation. *IEEE Robot. Autom. Mag.* **2012**, *19*, 80–91. doi:10.1109/mra.2012.2206675. [CrossRef]
28. Lu, X.; Yao, J.; Tu, J.; Li, K.; Li, L.; Liu, Y. Pairwise linkage for point cloud segmentation. *ISPRS Ann. Photogramm. Remote. Sens. Spat. Inf. Sci.* **2016**, *III-3*, 201–208. doi:10.5194/isprs-annals-iii-3-201-2016. [CrossRef]
29. Aljumaily, H.; Laefer, D.F.; Cuadra, D. Urban point cloud mining based on density clustering and mapreduce. *J. Comput. Civ. Eng.* **2017**, *31*, 04017021. doi:10.1061/(asce)cp.1943-5487.0000674. [CrossRef]
30. Vo, A.V.; Truong-Hong, L.; Laefer, D.F.; Bertolotto, M. Octree-based region growing for point cloud segmentation. *ISPRS J. Photogramm. Remote. Sens.* **2015**, *104*, 88–100. doi:10.1016/j.isprsjprs.2015.01.011. [CrossRef]
31. Tóvári, D.; Pfeifer, N. Segmentation based robust interpolation-a new approach to laser data filtering. *ISPRS Int. Arch. Photogramm. Remote. Sens. Spat. Inf. Sci.* **2005**, *36*, 79–84.
32. Rabbani, T.; Van Den Heuvel, F.; Vosselmann, G. Segmentation of point clouds using smoothness constraint. *ISPRS Int. Arch. Photogramm. Remote. Sens. Spat. Inf. Sci.* **2006**, *36*, 248–253.
33. Besl, P.; Jain, R. Segmentation through variable-order surface fitting. *IEEE Trans. Pattern Anal. Mach. Intell.* **1988**, *10*, 167–192. doi:10.1109/34.3881. [CrossRef]
34. Nurunnabi, A.; Belton, D.; West, G. Robust segmentation for large volumes of laser scanning three-dimensional point cloud data. *IEEE Trans. Geosci. Remote. Sens.* **2016**, *54*, 4790–4805. doi:10.1109/tgrs.2016.2551546. [CrossRef]
35. Harwin, S.; Lucieer, A.; Osborn, J. The impact of the calibration method on the accuracy of point clouds derived using unmanned aerial vehicle multi-view stereopsis. *Remote. Sens.* **2015**, *7*, 11933–11953. [CrossRef]
36. Brown, D.C. Close-range camera calibration. *Photogramm. Eng.* **1971**, *37*, 855–866.
37. Eltner, A.; Kaiser, A.; Castillo, C.; Rock, G.; Neugirg, F.; Abellán, A. Image-based surface reconstruction in geomorphometry–merits, limits and developments. *Earth Surf. Dyn.* **2016**, *4*, 359–389. [CrossRef]
38. Verhoeven, G. Taking computer vision aloft–archaeological three-dimensional reconstructions from aerial photographs with photoscan. *Archaeol. Prospect.* **2011**, *18*, 67–73. [CrossRef]
39. Rusu, R.B.; Cousins, S. 3D is here: Point cloud library (pcl). In Proceedings of the IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 1–4, doi:10.1109/ICRA.2011.5980567. [CrossRef]
40. Xu, Y.; Yao, W.; Tuttas, S.; Hoegner, L.; Stilla, U. Unsupervised segmentation of point clouds from buildings using hierarchical clustering based on gestalt principles. *IEEE J. Sel. Top. Appl. Earth Obs. Remote. Sens.* **2018**, pp. 1–17. doi:10.1109/jstars.2018.2817227. [CrossRef]
41. Papon, J.; Abramov, A.; Schoeler, M.; Worgotter, F. Voxel cloud connectivity segmentation-supervoxels for point clouds. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2027–2034. doi:10.1109/cvpr.2013.264. [CrossRef]
42. Xu, Y.; Heogner, L.; Tuttas, S.; Stilla, U. A voxel- and graph-based strategy for segmenting man-made infrastructures using perceptual grouping laws: comparison and evaluation. *Photogramm. Eng. Remote. Sens.* **2018**. doi:10.14358/PERS.84.6.37. [CrossRef]
43. Weinmann, M.; Jutzi, B.; Hinz, S.; Mallet, C. Semantic point cloud interpretation based on optimal neighborhoods, relevant features and efficient classifiers. *ISPRS J. Photogramm. Remote. Sens.* **2015**, *105*, 286–304. doi:10.1016/j.isprsjprs.2015.01.016. [CrossRef]
44. Stein, S.C.; Schoeler, M.; Papon, J.; Worgotter, F. Object partitioning using local convexity. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 304–311. doi:10.1109/cvpr.2014.46. [CrossRef]
45. Xu, Y.; Yao, W.; Hoegner, L.; Stilla, U. Segmentation of Building Roofs from Airborne LiDAR Point Clouds Using Robust Voxel-Based Region Growing. *Remote. Sens. Lett.* **2017**, *8*, 1062–1071. [CrossRef]
46. Peng, B.; Zhang, L.; Zhang, D. A survey of graph theoretical approaches to image segmentation. *Pattern Recognit.* **2013**, *46*, 1020–1038. doi:10.1016/j.patcog.2012.09.015. [CrossRef]

47. Yao, W.; Hinz, S.; Stilla, U. Automatic vehicle extraction from airborne LiDAR data of urban areas aided by geodesic morphology. *Pattern Recognit. Lett.* **2010**, *31*, 1100–1108. doi:10.1016/j.patrec.2010.02.006. [CrossRef]
48. Funkhouser, T.; Golovinsky, A. Min-cut based segmentation of point clouds. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops (ICCV Workshops), Kyoto, Japan, 27 September–4 October 2009; pp. 39–46.
49. Sun, Z.; Xu, Y.; Hoegner, L.; Stilla, U. Classification of MLS point clouds in urban scenes using detrended geometric features from supervoxel-based local context. *ISPRS Ann. Photogramm. Remote. Sens. Spat. Inf. Sci.* **2018**, *IV-2*, 271–278. doi:10.5194/isprs-annals-iv-2-271-2018. [CrossRef]
50. Yang, B.; Dong, Z.; Zhao, G.; Dai, W. Hierarchical extraction of urban objects from mobile laser scanning data. *ISPRS J. Photogramm. Remote. Sens.* **2015**, *99*, 45–57. [CrossRef]
51. Kazhdan, M.; Bolitho, M.; Hoppe, H. Poisson surface reconstruction. In Proceedings of the Fourth Eurographics Symposium on Geometry Processing, Sardinia, Italy, 26–28 June 2006; Volume 7.