



Title	Automatic registration of MLS point clouds and SfM meshes of urban area
Author(s)	Yoshimura, Reiji; Date, Hiroaki; Kanai, Satoshi; Honma, Ryohei; Oda, Kazuo; Ikeda, Tatsuya
Citation	Geo-spatial Information Science, 19(3), 171-181 <a href="https://doi.org/10.1080/10095020.2016.1212517">https://doi.org/10.1080/10095020.2016.1212517</a>
Issue Date	2016
Doc URL	<a href="http://hdl.handle.net/2115/67322">http://hdl.handle.net/2115/67322</a>
Rights	This is an Accepted Manuscript of an article published by Taylor & Francis in Geo-spatial Information Science on Geo-spatial information science, 2016 Vol.19, no. 3, 171–181, available online: <a href="http://www.tandfonline.com/10.1080/10095020.2016.1212517">http://www.tandfonline.com/10.1080/10095020.2016.1212517</a> .
Rights(URL)	<a href="http://creativecommons.org/licenses/by/4.0/">http://creativecommons.org/licenses/by/4.0/</a>
Type	article
File Information	Automatic registration of MLS point clouds and SfM meshes of urban area.pdf



[Instructions for use](#)

## Automatic registration of MLS point clouds and SfM meshes of urban area

Reiji Yoshimura<sup>a</sup>, Hiroaki Date<sup>a</sup>, Satoshi Kanai<sup>a</sup>, Ryohei Honma<sup>b</sup>, Kazuo Oda<sup>b</sup> and Tatsuya Ikeda<sup>b</sup>

<sup>a</sup>Graduate School of Information Science and Technology, Hokkaido University, Sapporo, Japan; <sup>b</sup>Asia Air Survey Co., Ltd, Kawasaki, Japan

### ABSTRACT

Recent advances in 3D scanning technologies allow us to acquire accurate and dense 3D scan data of large-scale environments efficiently. Currently, there are various methods for acquiring large-scale 3D scan data, such as Mobile Laser Scanning (MLS), Airborne Laser Scanning, Terrestrial Laser Scanning, photogrammetry and Structure from Motion (SfM). Especially, MLS is useful to acquire dense point clouds of road and road-side objects, and SfM is a powerful technique to reconstruct meshes with textures from a set of digital images. In this research, a registration method of point clouds from vehicle-based MLS (MLS point cloud), and textured meshes from the SfM of aerial photographs (SfM mesh), is proposed for creating high-quality surface models of urban areas by combining them. In general, SfM mesh has non-scale information; therefore, scale, position, and orientation of the SfM mesh are adjusted in the registration process. In our method, first, 2D feature points are extracted from both SfM mesh and MLS point cloud. This process consists of ground- and building-plane extraction by region growing, random sample consensus and least square method, vertical edge extraction by detecting intersections between the planes, and feature point extraction by intersection tests between the ground plane and the edges. Then, the corresponding feature points between the MLS point cloud and the SfM mesh are searched efficiently, using similarity invariant features and hashing. Next, the coordinate transformation is applied to the SfM mesh so that the ground planes and corresponding feature points are adjusted. Finally, scaling Iterative Closest Point algorithm is applied for accurate registration. Experimental results for three data-sets show that our method is effective for the registration of SfM mesh and MLS point cloud of urban areas including buildings.

### ARTICLE HISTORY

Received 12 April 2016  
Accepted 17 June 2016

### KEYWORDS

Registration; MLS point clouds; SfM mesh; urban area; hash; similarity invariant feature

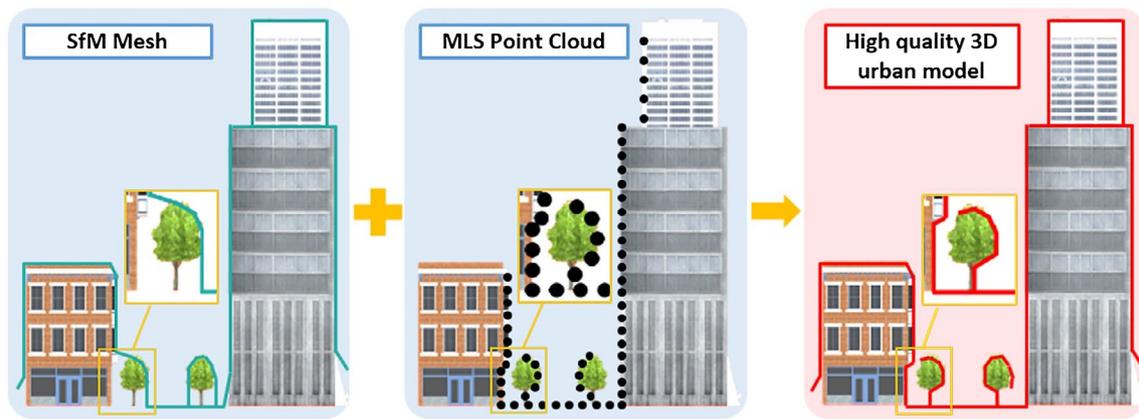
### 1. Introduction

High-quality 3D urban models are required for navigation and web mapping services, and 3D scan data is useful to create them. Currently, there are various ways for acquiring the 3D scan data of large-scale environments, such as laser scanning technology including Airborne Laser Scanning (ALS), Terrestrial Laser Scanning (TLS), and Mobile Laser Scanning (MLS), and image-based 3D reconstruction technology including photogrammetry and Structure from Motion (SfM) which estimates the 3D geometry and the position and orientation of the camera, simultaneously. Especially, MLS is useful to acquire high-density point clouds of the urban roads and the objects around them, efficiently. On the other hand, SfM is a powerful technique to reconstruct the textured meshes of a wide range of urban areas. Therefore, it is possible to reconstruct the high-quality 3D urban models by combining MLS point clouds and SfM meshes.

To combine the 3D scan data acquired from different positions or systems, a registration process is required as the pre-processing. A number of registration methods have been proposed. Generally, registration of 3D scan

data consists of two steps: rough registration and accurate registration. For the automatic rough registration, features extracted from the scan data, e.g. Fast Point Feature Histograms (Rusu, Blodow, and Beetz 2009) and 3D Scale Invariant Feature Transform (SIFT) (Theiler, Wegner, and Schindler 2013), or randomized sampling approach, e.g. 4-points congruent sets (4PCS) (Aiger, Mitra, and Cohen-Or 2008) are used. For the accurate registration, Iterative Closest Point (ICP) algorithms and its variants are often used (Besl and McKay 1992; Rusinkiewicz and Levoy 2001). However, there is some room for improvement regarding the robustness for the variations of the scanned objects and the environments, and the computational efficiency.

In this research, we propose a registration method of the SfM mesh and the MLS point cloud for constructing the high-quality 3D urban model (Figure 1). This research targets the registration of the scan data of the urban area including the building. Moreover, the SfM meshes handled in this method have non-scale information; therefore, the scale of the SfM mesh is adjusted to the MLS point cloud in addition to the position and orientation.



**Figure 1.** Constructing high-quality 3D urban model.

Our method consists of automatic efficient rough registration using the 2D feature points and the accurate registration by the scaling ICP (SICP) algorithm (Zinßer, Schmidt, and Niemann 2005) which is the extension method of ICP algorithm. The rough registration is done by the similarity transformation invariant feature extraction from the 2D feature points which are obtained from the building's vertical edges and the ground plane, the efficient feature point matching using hashing, and the coordinate transformations. After the rough registration, the accurate registration is performed by SICP.

The feature of our method is the efficiency of the registration, with scale adjustment, and robustness against the different densities and scanning noises of the 3D scan data in the urban areas. By comparing the results of the rough registration and accurate registration, the accuracy is evaluated using three data-sets of the SfM mesh and the MLS point cloud.

The rest of the paper is organized as follows. Section 2 discusses the related works. The detail of the proposed registration method of the SfM mesh and the MLS point cloud is described in Section 3. In Section 4, experimental results for the three data-sets are shown in order to evaluate the effectiveness of our method. Section 5 shows conclusions of the paper.

## 2. Related work

For the rough registration of the 3D scan data of the large-scale environments, various methods based on features extracted from the point clouds and randomized approach have been proposed. Kang et al. (2009) proposed a registration method between the point clouds by using panoramic reflectance images and SIFT, and Nüchter et al. (2011) proposed a registration method between the point clouds by using the skylines from the range images. These methods require panoramic images obtained from a fixed scan position; therefore, the methods are applicable only to the TLS point clouds. As a registration method using more general features of point clouds, Rusu, Blodow, and Beetz (2009) proposed a

registration method of point clouds by using the feature histogram generated from the set of the point normals in the local point clouds. The method can find the correspondences between two unrelated point clouds, but it is difficult to apply the method to the data with a large density difference. Moreover, the method requires relatively correct normals for generating histograms, but robust normal estimation of the complex urban scene including trees or wired objects is still one of the challenging problems. AL-Durgham, Habib, and Kwak (2013) proposed a registration method of the point clouds by random sample consensus (RANSAC) for straight line edges extracted from the planes in the scene. They concluded that the improvement of the computational time is required. The 4PCS is one of the effective methods for registration of the 3D point clouds with a large number of outliers and less overlaps. They used coplanar 4-points as the key, and internal ratio of intersection points of the diagonals of the 4-points is used for efficient search of congruent set in each data. However, the low robustness and high computational cost of the method for the point clouds of large-scale environments are concluded by Theiler, Wegner, and Schindler (2013). These methods have some limitations for the application to the registration of SfM mesh and MLS point cloud. Moreover, these methods cannot adjust the scale of the model.

Some registration methods which can adjust the scale of the 3D scan data of large-scale environments have been proposed as follows. Kochi (2013) proposed a registration method for interpolating the TLS point cloud by the mesh obtained from photogrammetry. In this method, the vertical edges of the objects in the scene are extracted from both data, and they are used in the matching of the two data. However, the method requires prior knowledge about the orientation of each data. Novák and Schindler (2013) proposed a registration method of the photogrammetric point cloud and the laser-scanned point cloud by using height maps. This method uses overlapping height maps with distinctive features, and the effectiveness of the method has been shown through several experiments using different types of scanned objects

and environments. However, the MLS point cloud and the SfM mesh generally have quite low overlapping, and useful height maps for matching may not be obtained because the commonly scanned surfaces in both data are only road surfaces and facades of the building.

In our research, the intersection points between the building's vertical edges and ground plane are used as feature points of the registration for solving the above problems. The features are extracted efficiently and stably from the 3D scan data including the buildings in the urban area. Furthermore, the *similarity invariant features*, which are invariant under similarity transformations, are extracted from the feature points for registration with scale adjustment. Simple feature representation from the buildings is useful for registration of scan data of urban environments. Recently, registration methods using building outlines on the grounds have been proposed for aligning the point clouds acquired from ALS and TLS (Yang et al. 2015) or sequent images from UAV (Yang and Chen 2015). In addition, similar to our method, Yang et al. (2016) uses intersection points between grounds and vertical lines from pole-like objects and vertical planes for the registration of TLS point clouds of large-scale urban scene. Based on the geometric constraints, spectral graphs, and triangulations, these methods provide useful matching methods of the building outlines and feature points on the ground for rough registration. However, scale adjustment is not considered, and the registration performance may depend on the overlap and scanned area of the object surfaces.

For the accurate registration, ICP algorithm (Besl and McKay 1992) is usually used. The ICP algorithm provides an accurate registration between two point clouds by iteratively minimizing registration error function which is the sum of squared distances between corresponding points or planes in each point cloud. The ICP requires correct initial position and orientation to avoid incorrect results caused by local minima of the error function. Therefore, it is better to apply rough registration before the ICP, if the original position and orientations of given point clouds differ greatly. Most variants of the ICP algorithm (Rusinkiewicz and Levoy 2001) derive a rigid body transformations for registration. Some accurate registration methods capable of scaling based on the ICP algorithm have been proposed. Du et al. (2007) and Zinßer, Schmidt, and Niemann (2005) extend the ICP algorithm by adding the scale factor. In our research, the SICP algorithm proposed by Zinßer, Schmidt, and Niemann (2005) is used for the accurate registration. The SICP algorithm provides simple calculation of the scale factor under the traditional computational framework of the ICP algorithm.

### 3. Registration method

#### 3.1. Overview of the algorithm

Our registration method is applied to the SfM mesh and MLS point cloud of the urban area including the buildings. In general, the SfM meshes have non-scale

information. Therefore, scale, position, and orientation of the SfM mesh are adjusted to the MLS point cloud in our registration method.

Several geometric features of buildings, such as the surfaces, corners, edges, etc., may be useful for the registration of the SfM mesh and the MLS point cloud. In this research, we use the vertical edges of the building, which can be extracted stably from both data. By assuming that the ground planes are identical in both data, we reduce the 3D registration problem to a 2D problem consisting of 2DOF for translations and 1DOF for rotation. To solve the 2D registration problem, 2D feature points, which are intersections between the ground plane and the building's vertical edges, are used. For the registration with scaling of the SfM mesh, feature point matching is efficiently performed using the similarity invariant features and a hash table.

Figure 2 shows the overview of our method. Our method consists of the following three steps.

*Step 1: Feature Point Extraction* (A1 and A2 in Figure 2). First, planar regions are extracted by region growing, and the ground plane is detected based on the number of points or triangles in the regions. Then, intersection lines between neighboring planar regions are calculated, and vertical intersection lines are extracted. Finally, intersection points between the vertical lines and the ground plane are extracted as 2D feature points. These processes are applied to both SfM mesh and MLS point cloud.

*Step 2: Feature Point Matching* (A3 in Figure 2). The corresponding feature points between the SfM mesh and the MLS point cloud are efficiently found using similarity invariant features obtained from each of the feature points and the hash table. Then, the coordinate transformation is applied to the SfM meshes so that the ground planes and the corresponding feature points can be adjusted.

*Step 3: Accurate Registration* (A4 in Figure 2). The SICP algorithm (Zinßer, Schmidt, and Niemann 2005) is applied to both data. In this process, the source (movable data) is the SfM mesh after rough registration and the target (fixed data) is the MLS point cloud.

In the following sections, the details of Steps 1 and 2 are described.

#### 3.2. Feature point extraction

##### 3.2.1. Feature points extraction for SfM mesh

###### (1) Planar region extraction

The connected sets of triangles which have a similar normal direction are extracted by region growing. Region growing is performed in the following procedure.

*Step 1:* The seed triangle  $j$  is selected randomly from the unprocessed triangles, and the seed triangle's normal is set to the base normal of the region.

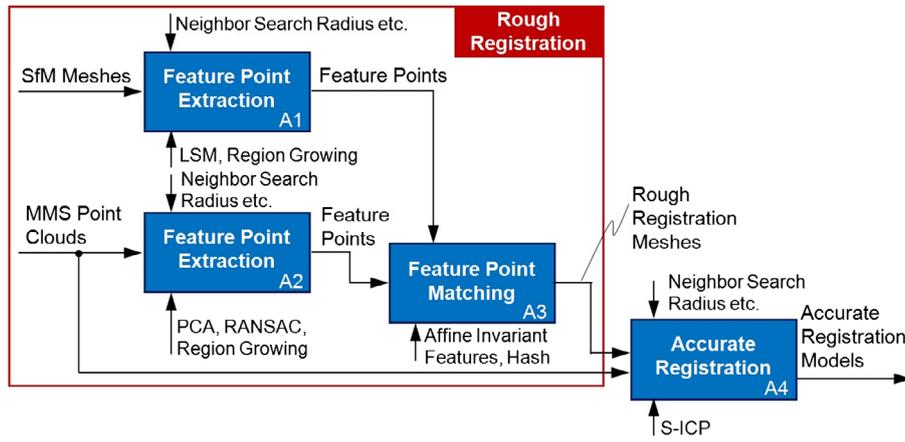


Figure 2. Algorithm of our registration method.

*Step 2:* If the adjacent triangles of triangle  $j$  satisfy the region growing conditions, they are added to the region of triangle  $j$ . The region growing conditions are that triangle  $i$  is unprocessed and the angle between the normal of triangle  $i$  and the base normal is smaller than the threshold.

*Step 3:* If the adjacent triangles are added to the region in *Step 2*, replace the added triangle with triangle  $j$ , and return to *Step 2*. At this time, if a certain number of triangles are added to the region, the base normal is updated to the average of the triangle normals in the region.

*Step 4:* If unprocessed triangles exist, return to *Step 1*.

Following this, plane fitting by the least square method is applied to the region with a certain number of triangles. The ground plane is defined as one of the regions with the highest number of triangles in the obtained planar regions.

### (2) Vertical edge extraction and feature points extraction

First, if an edge is shared by two triangles belonging to different vertical planar regions, the regions are recognized as neighboring regions. For all neighboring regions, the intersecting lines between their planes are calculated. Then, the intersecting lines which are nearly vertical to the ground plane are extracted. Finally, intersection points between the extracted vertical lines and the ground plane are calculated as 2D feature points.

#### 3.2.2. Feature points extraction for MLS point clouds

##### (1) Planar region extraction

Planar regions are extracted from the MLS point cloud by using region growing similar to that of the SfM mesh. The normal vector of each point is calculated by PCA (Principal Component Analysis). A detailed process of the region growing is same as that of the SfM mesh. After planar region extraction, plane fitting by RANSAC (Fischler and Bolles 1981; Schnabel, Wahl, and Klein 2007) is applied to the each region for obtaining the plane of the region.

##### (2) Vertical edge extraction and feature points extraction

If a point in region  $r_a$  has neighbor points in the other region  $r_b$ , these regions  $r_a$  and  $r_b$  are recognized as neighboring regions. Similar to the process for SfM mesh, vertical intersection lines are extracted from the neighboring regions, and the intersection points between them and the ground plane are calculated.

#### 3.3. Feature point matching

In this research, to find the correspondence of the feature points independently from the scale of the SfM mesh, similar triangles consisting of three feature points are used. The similar triangle pair is found from both data and the coordinate transformation is calculated so that they are adjusted to each other. This process is applied repeatedly to some similar triangle pairs, and the optimal coordinate transformation is obtained. The optimal coordinate transformation maximizes the number of feature points of the SfM mesh corresponding to that of the MLS point cloud.

The similar triangles consisting of three feature points have the same ratio of the distances from a feature point to the other two points and the angle between the two difference vectors. In this paper, the three points are defined as  $(i, j, k)$ , where  $i$  is a reference point and  $\text{dist}(i, k) \geq \text{dist}(i, j)$ ,  $\text{dist}(x, y)$  is the distance between point  $x$  and  $y$ . The ratio  $L_{i,j,k}$  is defined by  $\text{dist}(i, j)/\text{dist}(i, k)$ . The angle  $\theta_{i,j,k}$  is defined as the angle between the vector of points  $i$  and  $j$  and that of points  $i$  and  $k$ . In this research, pairs of these two measures are used as similarity invariant features, and similar triangles (i.e. three points have similar similarity invariant features) are searched efficiently using the hash table created by using the two measures.

The hash table generation and the search process for efficiently finding the corresponding three feature points are described as follows.

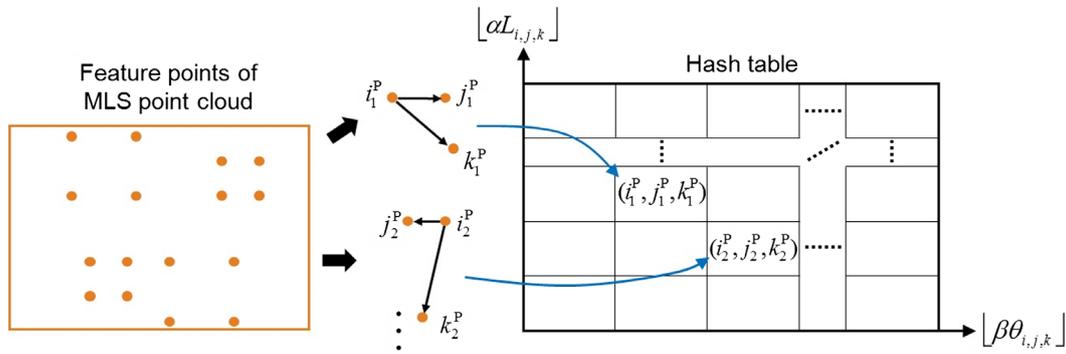


Figure 3. Hash table generation of rough registration.

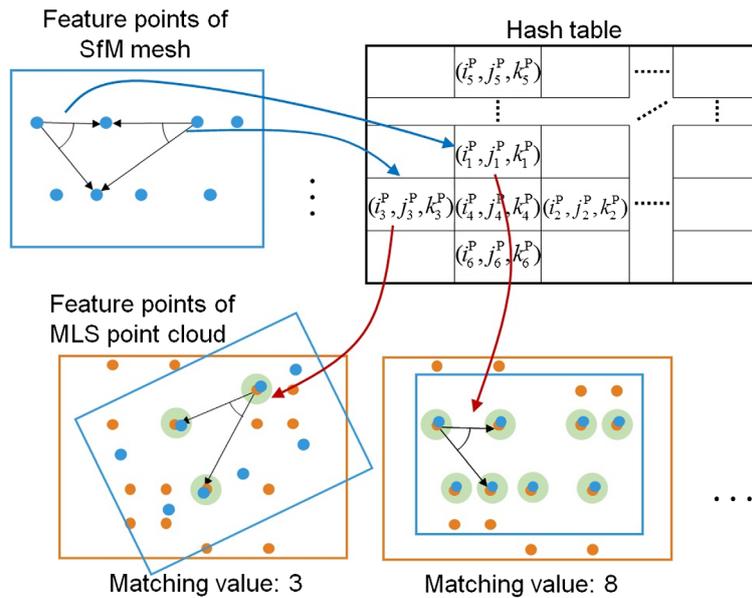


Figure 4. Search process and matching evaluation of rough registration.

(1) Hash table generation (Figure 3)

The similarity invariant features are calculated for all 3-tuples  $\{(i^p, j^p, k^p)\}$  of the feature points of MLS point cloud. Each tuple is stored in the 2D hash table whose keys are  $L_{i,j,k}$  and  $\theta_{i,j,k}$  as shown in Figure 3.

(2) Search process and matching evaluation (Figure 4)

First, a 3-tuple consisting of feature points  $(i^M, j^M, k^M)$  is selected randomly from the feature points of the SfM mesh, and the similarity invariant features of  $(i^M, j^M, k^M)$  are calculated. Next, by searching the hash table based on the similarity invariant features, the set of tuples  $\{(i^p, j^p, k^p)\}$  of the feature points of the MLS point cloud, which have similar similarity invariant features, are obtained. Then, for each tuple  $(i^p, j^p, k^p)$ , the coordinate transformation is calculated so that  $i^p, i^M$  and  $j^p, j^M$  are adjusted, and the coordinate transformation is applied to the feature points of the SfM mesh. Finally, “the matching value” is calculated. The matching value is the number of the feature points of the SfM mesh, which have the feature points of the MLS point clouds within a constant distance.

The above process is repeated  $N$  times, and the coordinate transform matrix which gives the maximum matching value is obtained.  $N$  is calculated by the following equation (Fischler and Bolles 1981):

$$N = \frac{\log(1-p)}{\log(1-w^3)} \quad (1)$$

where  $p$  is the user-specified success probability, and  $w$  is the ratio of the number of feature points of the SfM mesh, which correspond to those of the MLS point cloud, to the one of all feature points of the SfM mesh.

After the feature point matching, rigid body transformation is applied to the SfM mesh so that the ground planes of both data are adjusted, and then similarity transformation is applied so that the feature points of both data are adjusted.

### 3.4. Accurate registration

In the SICP algorithm (Zinßer, Schmidt, and Niemann 2005), ICP with scale adjustment is achieved by adding the scale factor  $s$  to the standard ICP algorithm (Besl

and McKay 1992; Rusinkiewicz and Levoy 2001). The scale factor  $s$ , the rotation matrix  $\mathbf{R}$ , and the translation vector  $\mathbf{t}$  for adjusting position, orientation and scale of the SfM meshes to the ones of the MLS point clouds are obtained from the following minimization:

$$\underset{s, \mathbf{R}, \mathbf{t}}{\text{minimize}} \sum_{i=1}^N \|\mathbf{q}_i - s\mathbf{R}\mathbf{p}_i - \mathbf{t}\|^2 \quad (2)$$

where  $\mathbf{p}_i$  is a vertex of the SfM mesh,  $\mathbf{q}_i$  a point of the MLS point cloud corresponding to the  $\mathbf{p}_i$  and  $N$  the number of the sampled vertices of the SfM mesh.  $\mathbf{R}$  and  $\mathbf{t}$  in Equation (2) are calculated by using the conventional methods such as the singular value decomposition or quaternions. The scale factor  $s$  is obtained by the following equation:

$$s = \frac{\sum_{i=1}^N \mathbf{R}(\mathbf{p}_i - \bar{\mathbf{p}}) \cdot (\mathbf{q}_i - \bar{\mathbf{q}})}{\sum_{i=1}^N (\mathbf{p}_i - \bar{\mathbf{p}}) \cdot (\mathbf{p}_i - \bar{\mathbf{p}})} \quad (3)$$

where  $\bar{\mathbf{p}}$  and  $\bar{\mathbf{q}}$  are the barycentre of the vertices and points used in the Equation (2), respectively. Finally, the similarity transformation defined by the  $s$ ,  $\mathbf{R}$  and  $\mathbf{t}$  is applied to the SfM mesh.

## 4. Results

The proposed method was applied to three data-sets which are pairs of the SfM mesh and the MLS point cloud acquired from the same regions. The three data-sets have different characteristics, i.e. size, height, and number of buildings, and the overlap data. The test site is the area around the National Stadium Japan. The size of each SfM mesh is 250 m square area and the SfM mesh covers the entire area. On the other hand, each MLS point cloud represents the roads, road-side objects, and buildings in a part of the area of corresponding to the SfM mesh. The MLS system which was used for acquiring point clouds was Riegl VQ-250, and each SfM mesh was created by the commercial software Smart3DCapture (Acute3D n.d.). In our data-sets, the SfM meshes have the correct scale because they are generated by using the camera's position and orientation. In order to verify the effectiveness of our method, we changed the scale, position, and orientation of the SfM meshes, and used them in the experiments. The specification of the PC used in the experiments was: CPU of Intel Core-i7 3.50 GHz, RAM with 32 GB, and GPU of GeForce GTX680.

### 4.1. Data-set 1

Figure 5(a) shows the SfM mesh (#vertices: 489,625) and the MLS point cloud (#points: 29,334,832). Figure 5(b) shows the initial position and orientation of each data. Data-set 1 includes large tall buildings. The vertical edges and feature points extracted by our method are shown in Figure 5(c). In the figure, only the extracted

planar regions are shown, and the colored triangles and points correspond to the extracted the planar regions. The building edges on the road side could be extracted in the MLS point cloud. On the other hand, in the SfM mesh, more building edges including the back side of the buildings could also be extracted. The number of feature points of the SfM mesh and the MLS point cloud were 25 and 9, respectively. Figure 5(d) shows the results of the rough and accurate registration. From the left figure, it was confirmed that the rough registration succeeded. The right figure shows that the position and orientation of the SfM mesh was modified slightly by the accurate registration. The processing time of the rough registration and the accurate registration were 11 and 48 s; The scale-error rates of the rough registration and the accurate registration were about 0.26 and 0.11%, respectively.

Figure 5(e) shows the registration errors after the rough registration and accurate registration. The distance from the vertex of the SfM mesh to its nearest point of the MLS point cloud is calculated as the error. The average error after the rough registration and accurate registration were 0.44 and 0.37 m, respectively. The average error is calculated using the points with error within 1 m to avoid the influence of the non-overlapping regions. By accurate registration, the error of the road and building surface decreases.

These results show that our method is useful for roughly aligning the scale, position, and orientation of the SfM mesh and the MLS point cloud of urban areas with large buildings, and can solve the rough registration problem before the accurate registration by SICIP.

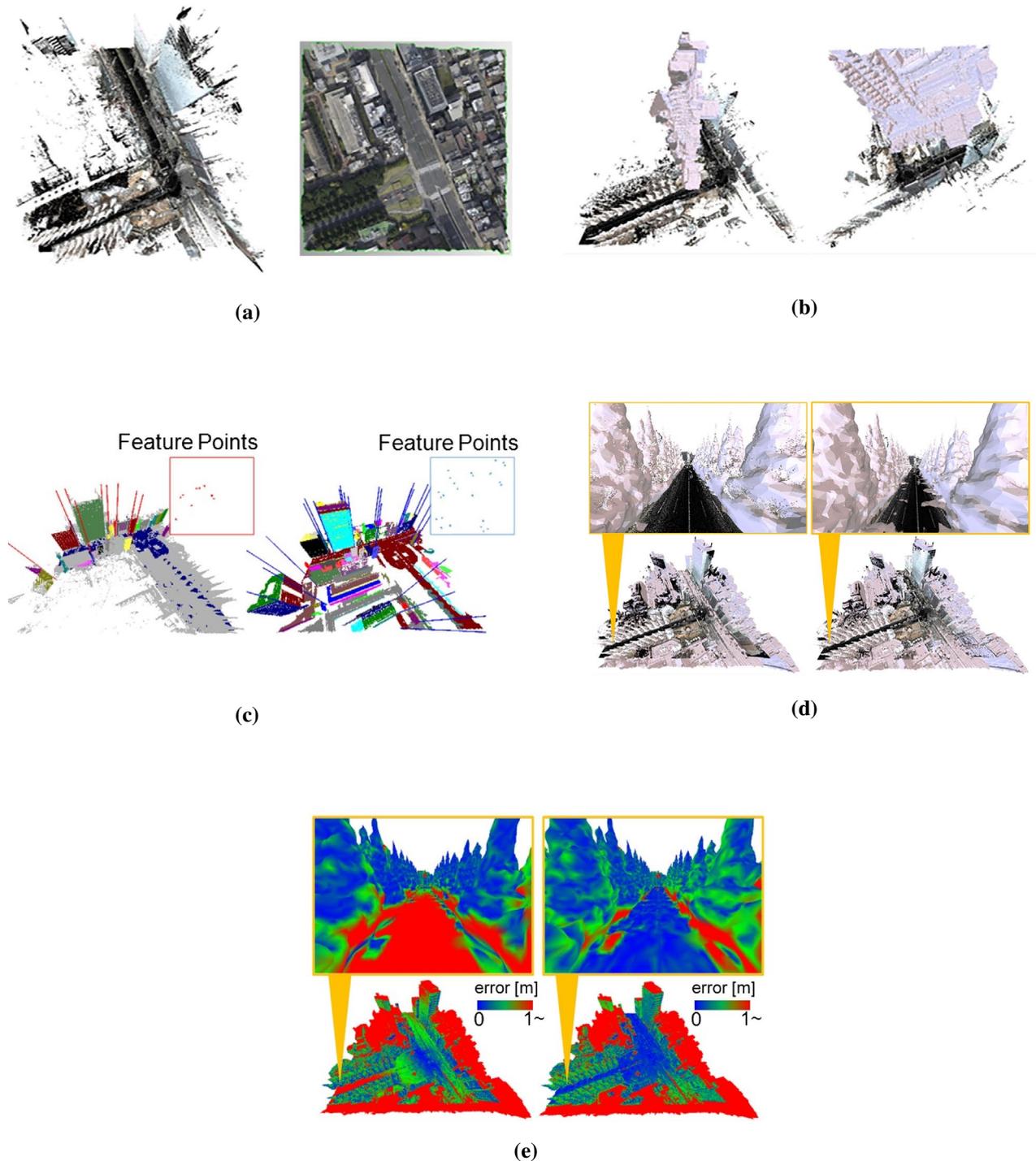
### 4.2. Data-set 2

Data-set 2 is scanned in the urban area with relatively low buildings. Input data, initial position and orientation, and registration results of Data-set 2 are shown in Figure 6 (similar to that of Data-set 1).

The number of the vertices of the SfM mesh and the points of the MLS point cloud are 474,792 and 34,163,957, respectively. The numbers of the feature points of the SfM mesh and the MLS point cloud are 58 and 51; The processing time of the rough registration and accurate registration are 11 and 51 s; The scale-error rates of the rough registration and accurate registration are about 0.41 and 0.18%; The average error after the rough registration and accurate registration is 0.48 and 0.42 m, respectively. From these experiments, it was observed that our method could work well for the scan data, which included a lot of relatively low buildings.

### 4.3. Data-set 3

Figure 7 shows the input data, initial position and orientation, and registration results of Data-set 3. In Data-set 3, overlapping of the SfM mesh and the MLS point cloud is relatively low, compared with the previous two



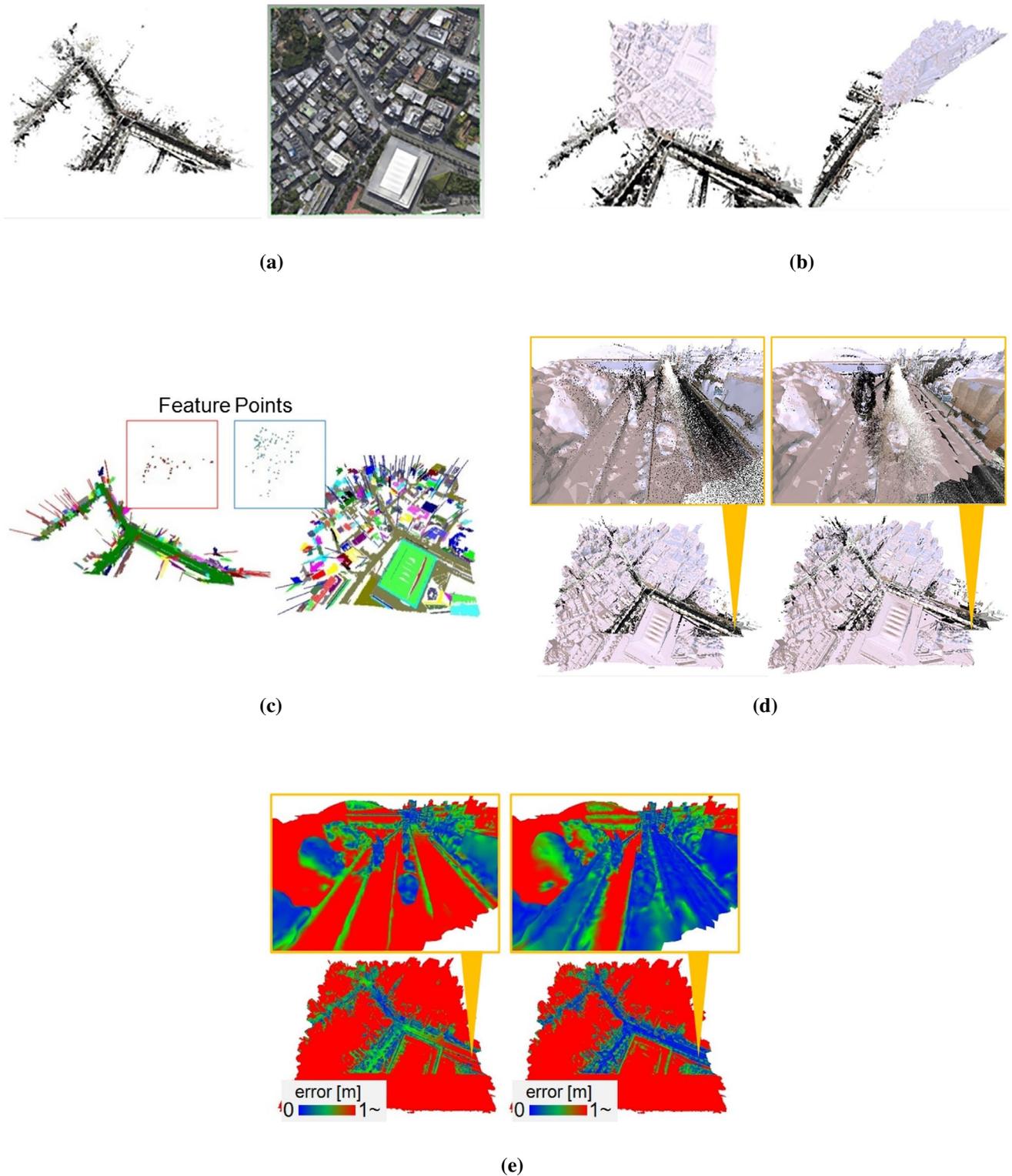
**Figure 5.** Experimental results of Data-set 1. (a) Aerial view of the MLS point cloud (left) and SfM mesh (right). (b) Input data from two viewpoints (pink: SfM mesh, colored points: MLS point cloud). (c) The vertical edges and extracted feature points (left: MLS point cloud, right: SfM mesh). (d) Rough registration result (left) and accurate registration result (right). (e) The distance from the vertex of the SfM mesh to its nearest point of the MLS point clouds is colored (left: rough registration, right: accurate registration).

data-sets, and most buildings exist in the area far from the road scanned by the MLS system. In spite of the small overlap of the SfM mesh and the MLS point cloud, our method could produce a correct rough registration result when enough feature points are able to be extracted.

The numbers of the feature points of the SfM mesh (#vertices: 426,256) and the MLS point cloud (#points: 14,339,237) are 87 and 27, respectively. The processing time of the rough registration and the accurate

registration are 9 and 22 s; The scale error rates of the rough registration and the accurate registration are about 0.49 and 0.02%; The average error after the rough registration and accurate registration is 0.53 and 0.48 m, respectively.

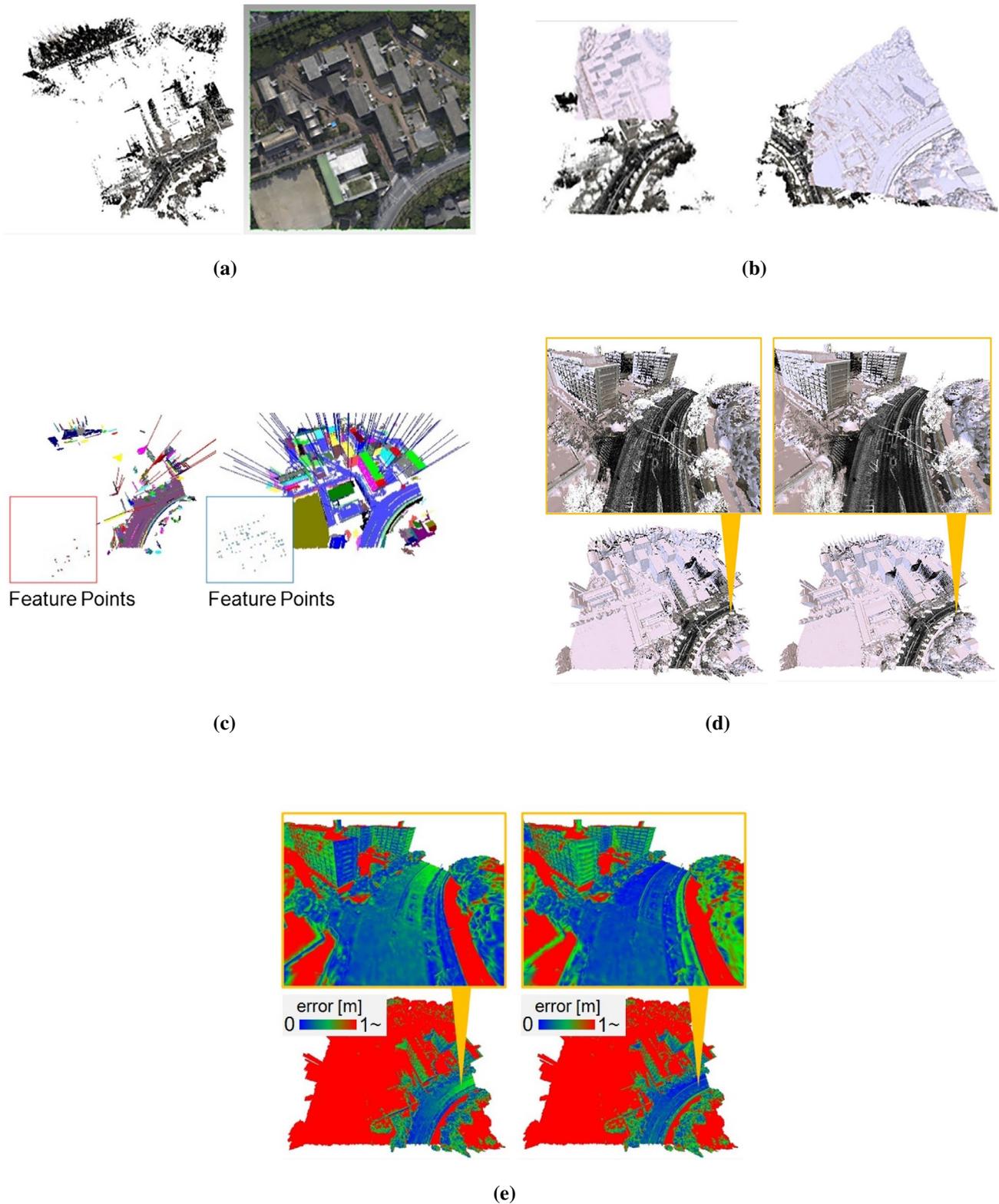
The three experimental results show that our method is useful for efficient automatic rough registration of SfM meshes from aerial photographs and MMS point clouds of urban areas with different complexities, i.e. different



**Figure 6.** Experimental results of Data-set 2. (a) Aerial view of the MLS point cloud (left) and SfM mesh (right). (b) Input data from two viewpoints (pink: SfM mesh, colored points: MLS point cloud). (c) The vertical edges and extracted feature points (left: MLS point cloud, right: SfM mesh). (d) Rough registration result (left) and accurate registration result (right). (e) The distance from the vertex of the SfM mesh to its nearest point of the MLS point clouds is colored (left: rough registration; right: accurate registration)

numbers and sizes of the buildings. The processing times for rough and accurate registration were less than 11 and 51 s, respectively, for the SfM meshes with about 0.4 million vertices and the MMS point clouds with tens of millions of points. This may be acceptable in many

applications of 3D scan data of urban areas. After the rough registration, the distance error defined by the average distances between closest points was 0.53 m at a maximum, and the maximum scale-error rate was 0.49%. Noticeable difference of the errors was not



**Figure 7.** Experimental results of Data-set 3. (a) Aerial view of the MLS point cloud (left) and SfM mesh (right). (b) Input data from two viewpoints (pink: SfM mesh, colored points: MLS point cloud). (c) The vertical edges and extracted feature points (left: MLS point cloud, right: SfM mesh). (d) Rough registration result (left) and accurate registration result (right). (e) The distance from the vertex of the SfM mesh to its nearest point of the MLS point clouds is colored (left: rough registration, right: accurate registration).

observed in three data-sets. After the SICP, both errors were improved and correct registration results were obtained. The resulting registered data will be useful for high-quality urban model construction by combining them, which will be included in our future work.

## 5. Conclusions

In this study, we proposed a registration method for SfM mesh and MLS point cloud for generating high-quality surface models of urban areas. In our method, first, the vertical edges of the buildings are extracted, and the

intersection points of the vertical edges and the ground are obtained as the feature points. Then, similarity invariant features are estimated from the feature points, and the feature is used for finding correspondences between the two data by hashing. Finally, the accurate registration by the SICP algorithm is conducted for the MLS point cloud and the SfM mesh after the rough registration.

In order to verify the effectiveness of our method, experiments using the three data-sets are carried out. The scale-error rates of the rough registration and the accurate registration were within 0.49 and 0.18%, respectively. The average error of the rough registration and the accurate registration were 0.53 and 0.48 m at maximum, respectively. The results of the experiments show that our method could work well for data-sets with different properties related to the size of buildings and overlapping of the data. The experimental results show that our method is useful for registration of the SfM meshes from aerial photographs and the MLS point clouds of urban areas.

## Funding

This work was partially supported by JSPS KAKENHI [grant number 26420073].

## Notes on contributors

**Reiji Yoshimura** is a master course student of Graduate School of Information Science and Technology, Hokkaido University, Japan. His research interests include segmentation and registration of point clouds from laser scanning and digital images.

**Hiroaki Date** is an associate professor of Graduate School of Information Science and Technology at Hokkaido University. He received his doctoral degree of engineering from Hokkaido University in 2003. His research interests include point cloud data processing, geometric modeling, and finite element mesh generation.

**Satoshi Kanai** is a full professor of Systems Science and Informatics Division in Graduate School of Information Science and Technology at Hokkaido University. He was an assistant professor in Hokkaido University from 1987 to 1989, and an associate professor in Tokyo Institute of Technology from 1989–1997. He is directing the projects of “Digital Ergonomics” and “Massive Digital Geometry Processing” research as a lab chair. His research interests include digital prototyping, virtual reality, and 3D geometric modeling. He is an editorial board member of *International Journal of Interactive Design and Manufacturing*, *Journal of Computational Design and Engineering*, and *Advances in Computational Design*. He is a member of JSPE, JSME, JSDE, and IEEE.

**Ryohei Honma** is an engineer of Defense and Security Department at Asia Air Survey Co., Ltd. He received his master’s degree of engineering from Hosei University in 2006. His research interests include computer graphics, 3D measurement and modeling, and point cloud processing.

**Kazuo Oda** is a general manager of Research and Development Institute at Asia Air Survey Co., Ltd. He received his doctoral degree of engineering from Tokyo University in 2005. His research interests include point cloud data processing, photogrammetry, and remote sensing.

**Tatsuya Ikeda** is a general manager of Defense and Security Department at Asia Air Survey Co., Ltd. His research interests include computer graphics, 3D measurement and modeling, and point cloud processing. He owns some patents on 3D viewing systems, omnidirectional image viewing systems, 3D processing algorithms and cloud computing.

## References

- Acute3D n.d. *Smart3DCapture*. Accessed October 11, 2015. <http://www.acute3d.com/>
- Aiger, D., N. J. Mitra, and D. Cohen-Or. 2008. “4-Points Congruent Sets for Robust Surface Registration.” *ACM Transactions on Graphics* 27 (3): 1–10. doi:10.1145/1399504.1360684.
- AL-Durgham, K., A. Habib, and E. Kwak. 2013. “RANSAC Approach for Automated Registration of Terrestrial Laser Scans Using Linear Features.” *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences II-5/W2*: 13–18. doi:10.5194/isprsannals-II-5-W2-13-2013.
- Besl, P. J., and N. D. McKay. 1992. “A Method for Registration of 3-D Shapes.” *IEEE Transactions on Pattern Analysis and Machine Intelligence* 14 (2): 239–256. doi:10.1109/34.121791.
- Du, S., N. Zheng, S. Ying, Q. You, and Y. Wu. 2007. “An Extension of the ICP Algorithm Considering Scale Factor.” *Proceedings of IEEE International Conference on Image Processing*, San Antonio, TX, 5: 193–196. doi:10.1109/ICIP.2007.4379798.
- Fischler, M. A., and R. C. Bolles. 1981. “Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography.” *Communications of the ACM* 24 (6): 381–395. doi:10.1145/358669.358692.
- Kang, Z., J. Li, L. Zhang, Q. Zhao, and S. Zlatanova. 2009. “Automatic Registration of Terrestrial Laser Scanning Point Clouds Using Panoramic Reflectance Images.” *Sensors* 9 (4): 2621–2646. doi:10.3390/s90402621.
- Kochi, N. 2013. “3D Measuring and Model Integration of Point Cloud and 2D Images through Robust Feature Detection, and Their Application of Architecture and Archaeological Ruins.” PhD diss., Hokkaido University.
- Novák, D., and K. Schindler. 2013. “Approximate Registration of Point Clouds with Large Scale Differences.” *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences II-5/W2*: 211–216. doi:10.5194/isprsannals-II-5-W2-211-2013.
- Nüchter, A., S. Gütlev, D. Borrmann, and J. Elseberg. 2011. “Skyline-based Registration of 3D Laser Scans.” *Geo-Spatial Information Science* 14 (2): 85–90. doi:10.1007/s11806-011-0449-4.
- Rusinkiewicz, S., and M. Levoy. 2001. “Efficient Variants of the ICP Algorithm.” *Proceedings of Third International Conference on 3-D Digital Imaging and Modeling*, Quebec, 145–152. doi:10.1109/IM.2001.924423.
- Rusu, R. B., N. Blodow, and M. Beetz. 2009. “Fast Point Feature Histograms (FPFH) for 3d Registration.” *Proceedings of IEEE International Conference on Robotics*

- and Automation*, Kobe, 3212–3217. doi:[10.1109/ROBOT.2009.5152473](https://doi.org/10.1109/ROBOT.2009.5152473).
- Schnabel, R., R. Wahl, and R. Klein. 2007. “Efficient RANSAC for Point-Cloud Shape Detection.” *Computer Graphics Forum* 26 (2): 214–226. doi:[10.1111/j.1467-8659.2007.01016.x](https://doi.org/10.1111/j.1467-8659.2007.01016.x).
- Theiler, P.W., J.D. Wegner, and K. Schindler, 2013. “Markerless Point Cloud Registration with Keypoint-based 4-Points Congruent Sets.” *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences II-5/W2*: 283–288. doi:[10.5194/isprsannals-II-5-W2-283-2013](https://doi.org/10.5194/isprsannals-II-5-W2-283-2013).
- Yang, B., and C. Chen. 2015. “Automatic Registration of UAV-borne Sequent Images and LiDAR Data.” *ISPRS Journal of Photogrammetry and Remote Sensing* 101: 262–274. doi:[10.1016/j.isprsjprs.2014.12.025](https://doi.org/10.1016/j.isprsjprs.2014.12.025).
- Yang, B., Z. Dong, F. Liang, and Y. Liu. 2016. “Automatic Registration of Large-scale Urban Scene Point Clouds Based on Semantic Feature Points.” *ISPRS Journal of Photogrammetry and Remote Sensing* 113: 43–58. doi:[10.1016/j.isprsjprs.2015.10.005](https://doi.org/10.1016/j.isprsjprs.2015.10.005).
- Yang, B., Y. Zang, Z. Dong, and R. Huang. 2015. “An Automated Method to Register Airborne and Terrestrial Laser Scanning Point Clouds.” *ISPRS Journal of Photogrammetry and Remote Sensing* 109: 62–76. doi:[10.1016/j.isprsjprs.2015.08.006](https://doi.org/10.1016/j.isprsjprs.2015.08.006).
- Zinßer, T., J. Schmidt, and H. Niemann, 2005. “Point Set Registration with Integrated Scale Estimation.” In *Proceedings of International Conference on Pattern Recognition and Image Processing*, Estoril, 116–119.