Automatic hierarchical registration of aerial and terrestrial image-based point clouds

Amin Baghani, Mohammad Javad Valadan Zoej and Mehdi Mokhtarzade
Faculty of Geodesy and Geomatics, K. N. Toosi University of Technology, Tehran, Iran

ABSTRACT
Point cloud registration has been a major research challenge in recent years. In this paper, a novel hierarchical method is proposed for registering Aerial and Terrestrial image-based Point Clouds (APC & TPC). This registration aims to yield a complete and dense coverage on both top and side faces in urban areas. The main challenges, however, arise from their heterogeneous views and very low overlap as well as the high discrepancies in scale and three-dimensional rotations between the World Coordinate System (WCS) of APC and the Camera Coordinate System (CCS) of TPC. The proposed method begins by an innovative TPC-sufficient method to solve two rotations of TPC around x- and y-axis. After that, in horizontal registration phase, 2D façade lines were extracted from both datasets; and accordingly a novel polar parameterised mathematical model was extended for simultaneous robust matching and parameter estimation. Finally, vertical registration was done through calculating the dominant vertical shift between the points of the ground planes extracted from both datasets. The evaluation results in two different modes, which were conducted on two different urban datasets, showed the efficiency of the proposed method (RMSE error of 0.12 m/0.10 m in horizontal/vertical direction).

Introduction

During the recent years, getting complete, dense and high-resolution 3D information of urban areas has been a very important issue among scientific communities such as: remote sensing, photogrammetry, machine vision, and computer graphic (Musialski et al., 2013). Point clouds from active light detection and ranging (LiDAR) and passive imaging technologies have been major data sources for mapping applications in the photogrammetry and remote sensing communities. Using airborne platforms in urban environment, top faces are seen abundantly, but still on side faces are lacking. Terrestrial platforms give complete and dense information from side faces. Although the density on side faces is proper, these platforms do not give significant information from top faces (e.g. roofs) (Al-Durgham & Habib., 2013). So in order to achieve a complete coverage of urban areas, the fusion of both aerial and terrestrial views is necessary. Registration, as a prerequisite of data fusion, is the process of transforming different sets of data into one coordinate system.

Point cloud registration can be carried out using auxiliary method or based on the databases. In the first method, the auxiliary data (e.g. Ground Control Points [GCPs] and scale bars) and manual operations (e.g. installing artificial targets, identifying and measuring them precisely in both image and object spaces) are usually used to describe an integrated coordinate system for both datasets. Data-based methods are based on the primitive extraction from the point clouds and data registration. There are many reports on the data-based point cloud registration. The most important researches in this field are: registration of Airborne Laser Scanner (ALS) strips (Lee, Yu, Kim, & Habib, 2007; Han, Perng, & Chen, 2013; Favalli, Fornaciari, & Pareschi, 2009; Bang, Habib, Kusevic, & Mrstik, 2008; Skaloud & Lichti, 2006) and registration of multi-station Terrestrial Laser Scanner (TLS) data (Brenner, Dold, & Ripperda, 2008; Kang, Li, Zhang, Zhao, & Zlatanova, 2009; Theiler & Schindler, 2012; Yang & Zang, 2014).

Unlike identical views registration, there is a relatively few reports in the field of registration of aerial and terrestrial heterogeneous views (Cheng, Tong, Li, & Liu, 2013). Registration of ALS with TLS and ALS with vehicle laser scanning (VLS) is the most prominent part of studies in this field. ALS and TLS/VLS registration related researches can be categorised in three groups: point-based, line-based, and surface-based methods.

The Iterative Closest Point (ICP) (Belis & McKay, 1992; Chen & Medioni, 1992) together with its variants is known as a standard point-based method for registering two point clouds. However, ICP requires
an initial coarse registration, as well as the existence of a reasonable overlap between two point clouds, which is hardly affordable by heterogeneous datasets, especially in urban areas. Building corners obtained from intersecting the building boundaries was used as primitive in the proposed method by Cheng et al. (2013). Cheng, Tong, et al., (2015a) improved the geometric consistency of registration proposed in their previous work by compensating the geometric position of aerial corners in order to improve the consistency of conjugate corners set. Nevertheless, corner point extraction may be associated with large errors, and for buildings with eaves, there is a deviation for corners extracted from ALS and TLS. Kedzierski and Fryskowska (2014) proposed a wavelet-based method to improve the accuracy of ALS and detect the characteristic points. Intensity of laser data was used as auxiliary information to extract building corners. Regardless of inconstancy of point-based registration of the heterogeneous point clouds, intensity of laser data together with some assumptions made in this paper are not available in all circumstances.

Yang, Zang, Dong, and Huang (2015) modelled the extracted building outlines from ALS and TLS in a Laplacian matrix and registered the ALS and TLS point clouds by matching building outlines using spectral graph theory. However, the proposed method relies on building outline extraction with a high degree of completeness, which may not work robustly in presence of surrounding objects such as trees. Furthermore, some assumption made in this paper (e.g. clockwise ordering the extracted building outlines with known midpoint and endpoint) may limit its usability for larger and more complicated areas. Wu et al. (2014) proposed a line-constrained registration of ALS and TLS point clouds taken from multiple building. Extracted building outlines from both datasets were presented in slope-intercept form, and subsequently registration was done in this parameter space. However, using slope-intercept form, the perpendicular lines to the x-axis are not definable, and the slope for the nearly perpendicular lines is very big, which causes the numerical instability in registration computations and its uncertainty will be considerably propagated. Therefore, this deficiency beside the low automation grade of this method restricts the generalizability of this method. Cheng, Wu, et al., (2015b) proposed a hierarchical method for registering the ALS and VLS. Road network was used as auxiliary data in coarse registration and 3D building contours were used in fine registration. However, some assumptions in the proposed line-based method for fine registration (e.g. length consistency of extracted conjugate lines) besides the problems arising from building eaves and surrounding objects degrade the efficiency of the proposed method.

An automatic plane-based method for registering the ALS and VLS was proposed by Teo and Huang (2014). In the first phase, the multi-strips TLS were co-registered using plane primitives. In the second phase, integrated VLS were registered to the ALS, and a mathematical model based on least square 3D surface matching was used to minimise the Euclidean distances between the surfaces. However, planar primitives may not guarantee a robust estimation of the registration parameters, especially when there is a low overlap between two heterogeneous dataset. Wu and Fan (2016) proposed an approach for registering the point clouds of different ALS trajectories by finding and matching the corresponding linear plane features. The proposed method uses the parameters of corresponding planes for estimating the 3D rigid body transformation. However, despite a lot of linear plane features in urban areas, in order to avoid an ill-conditioned matrix in the observation system of the least square method, the proposed method is limited to select the linear planes with diverse orientations. This constraint may not be met in all condition (e.g. areas that do not have any sloping roof).

Previous studies in the field of APC and TPC registration have mainly used laser-based point clouds to provide a full coverage on both top and side faces. However, image-based techniques of generating dense point clouds are serious alternative for laser-based methods because they are able to produce accurate 3D point cloud by a small and low-cost camera (Snavely, Seitz, & Szeliski, 2008). Besides, the image-based point clouds can be used in the same way as laser-based point clouds (Rosnell & Honkavaara, 2012; Rossi, Mancini, Dubbini, Mazzone, & Capra, 2017).

In laser-based techniques, the condition that the vertical z-axis aligned with the local plumb line is easy to fulfil, where most TLSs are equipped with an internal levelling sensor or an external level plummets, able to compensate for the Z-axis direction. ALS datasets are normally rendered in the WCS, and TLS directly measures the actual distance, which makes their scale compatible. This is while the scale of the image-based point cloud may have an extreme difference over the real world when no auxiliary data are in used. Using the image-based point clouds two major challenges, arising from the differences in scale and tilt of the CCS relative to the WCS, are imposed on the point cloud registration.

Registering the aerial and terrestrial image-based point clouds is investigated in this paper. The georeferenced image-based APC is regarded as the reference data, while the CCS-referenced image-based TPC is treated as the source data to be registered to the reference data. To generate the image-based TPC,
no auxiliary data, ancillary equipment or manual operation had been used for constraining any parameter of CCS. Therefore, the TPC has seven unknown parameters relative to WCS, including: three translations, three rotations and one scale. Although use of image-based point cloud to provide a full coverage on both top and side faces has some advantages associated with its only dependency on low-cost and low-weight equipments, it has some disadvantages compared to use of laser-based point clouds, as follows:

- Extreme scale difference between the image-based aerial and terrestrial point clouds in comparison with uniform scale between laser-based point clouds.
- Large tilt in the not geo-referenced image-based TPC.
- The relatively lower accuracy and lower uniformity of image-based point clouds compared to laser-based point clouds (Al-Durgham & Habib, 2013).
- The limited field of view (FOV) of camera captures images at ground levels in urban areas, which result in limited coverage of TPC on the ground.

With regard to the abovementioned challenges, our main contribution is on the development of a new approach to register the aerial and terrestrial image-based point clouds, to work robustly even when there is a minimal overlap between datasets, as well as the great differences in scale and rotation angles. An automatic three-step hierarchical method is proposed for calculating the seven transformation parameters. Firstly, a TPC-sufficient method is introduced to resolve two rotational component of the tilt from TPC. Since urban areas are rich of the linear features, the automatically extracted 2D facade lines from both datasets are used for horizontal registration. In this case, a novel mathematical model based on polar parameterisation of 2D façade lines is presented. The extended mathematical model can overcome the problems of the conventional line-based methods. Finally, in the vertical registration phase, the dominant vertical shift existing between two datasets is calculated on the basis of the extracted ground planes.

**Data used and pre-processing**

Two different areas in Aliabad city located in the north of Iran have been used as the experimental area. Each dataset is made up of an image-based APC and its corresponding image-based TPC. These two datasets covered a total area of about $115m \times 91m$ (Figure 1(a–c)) and $98m \times 78m$ (Figure 1(d–f)) for datasets 1 and 2, respectively. APCs of both datasets, as well as the TPCs of both datasets, were captured at the same time, but there is a one-year interval between the APCs and TPCs data acquisition. Because of the inadequate and ill-condition block geometry of terrestrial photogrammetry, caused by camera location restrictions and imaging conditions in urban areas, usually it is not possible to establish well-condition block geometry for vast areas. Therefore, generating the seamlessly image-based point cloud using images, taken at ground

![Figure 1](image1.png)

Figure 1. Experimental dataset. (a) APC of dataset 1 from top view; (b) APC of dataset 1 from side view; (c) TPC of dataset 1; (d) APC of dataset 2 from top view; (e) APC of dataset 2 from side view; (f) TPC of dataset 2.
levels in urban areas, is not possible for a wide range, while there is no such limitation for aerial imaging.

As it is shown in Figure 1, an area containing a set of complex buildings with ceiling roofs and overhanging eaves is selected as experimental area of dataset 1. Most buildings are surrounded by an external wall with an approximately height of 1.80m; and consequently, ground information in TPC of dataset 1 is lacking due to the limited FOV. In dataset 2, some buildings have courtyard with exterior walls and some are not. Most buildings have overhanging eaves and rounded facades. In both datasets, trees as well as some urban elements (e.g. power line) do exist near the walls and facades.

Both APCs are generated using aerial images taken by a non-metric digital camera (Canon EOS 5D) mounted on an Unmanned Aerial Vehicle (UAV). Global Navigation Satellite System (GNSS) and inertial navigation system (INS) have not been installed on UAV to measure the position and rotation of the camera at the instant of exposure, and GCPs are used for geo-referencing.

Terrestrial convergent images, taken with a non-metric digital hand-held camera (Canon EOS M3) at street levels, have been used to produce the TPCs in CCS without use of any auxiliary data or manual operations. Both APCs and TPCs were generated by AgisoftPhotoscan software (Agisoft PhotoScan, 2017).

The accuracy of the APCs used in this study was estimated using a number of properly distributed Check Points (ChPs). These ChPs are measured in each APC dataset by a total station in the same coordinate system with the generated APCs. The horizontal and vertical difference between the coordinates of each ChPs and its corresponding in APC was calculated; the horizontal and vertical RMSE of the calculated differences was estimated as the criterion of APC error.

To assess the accuracy of the TPCs used in this study, since they are generated in the CCS without using GCPs or auxiliary data, the coordinate system of each TPC dataset was transformed to a local coordinate system. For this purpose, four properly distributed GCPs were measured in each dataset by a total station, and the transformation of each TPC dataset to the local coordinate system was established on the basis of these GCPs. Afterwards, a number of properly distributed ChPs in each TPC dataset were measured in the same coordinate system with their established local coordinate system. Finally, the accuracy of each TPC dataset was estimated similar to the method used for APC accuracy assessment. It should be noted that the TPCs transformed to the local coordinate system were not used in any other phases of this study; and the TPCs in the CCS are used in all phases of the proposed method. The number of ChPs used to assess the accuracy of each APC and TPC dataset, together with the estimated horizontal and vertical accuracies, are given in Table 1.

A low-pass filter was used for noise reduction of both datasets, through fitting a plane to each neighbourhood and then removing the points, which are too far away from the fitted plane. Neighbourhoods were constructed using 15 nearest neighbours of each point, and the relative maximum error equals to the average neighbours re-projection error was used to decide if a point is rejected or not. In addition, the points with less than three neighbours were removed as the isolated points. After cleaning the datasets, the average density of APCs on the top faces was 97pts/m2, and for the side faces in the TPCs was 114pts/m2.

### Methodology and results

Because the errors of image-based point clouds are mainly oriented along the lines characterised by sudden slope changes (Rossi et al., 2017), 3D building outlines extraction from image-based point cloud will be associated with a relatively large distortion. Therefore, adopting the 3D building outlines as primitive may not result in robust estimation of registration parameters. Considering this issue as well as the minimal overlap existing between APC and TPC and to simplify the registration problem, we formulated the seven-parameter registration problem into three phases.

The main phases of the proposed method (Figure 2) are: (1) Tilt removing from TPC on the basis of the detected facade planes. (2) Calculating polar parameters of 2D facade lines extracted from TPC based on the parameters of their analogous facade planes. (3) Calculating polar parameters of 2D facade lines extracted from APC based on geometrically constrained facade points and Hough transform. (4) Simultaneous matching and horizontal registration based on a polar parameterised mathematical model. (5) Vertical registration based on detection of the analogous ground planes and minimizing the height difference between the points of these planes.

### TPC tilt removing

Extracting the unique 2D facade lines from tilted TPC is impossible. Therefore, firstly two rotations

| Dataset | No. ChPs | Horizontal (m) | Vertical (m) |
|---------|---------|---------------|--------------|
| APC     | (1) 10  | 0.08          | 0.18         |
|         | (2) 10  | 0.08          | 0.15         |
| TPC     | (1) 8   | 0.06          | 0.07         |
|         | (2) 9   | 0.05          | 0.06         |

**Table 1. The accuracy of datasets used in this paper.**
some local variations in the neighbourhoods, which leads to a low accuracy in estimating the local plane parameters of some points. RANSAC (Fischler & Bolles, 1981) was used to fit a plane with unit normal vector \((n_x, n_y, n_z)\) and distance from origin \(d\) to every neighbourhood according to Equation 1.

\[
(n_x x + n_y y + n_z z + d) = 0
\]  

(1)

After calculating a local plane for each point, TPC was clustered into the robust global planes. Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester, Kriegel, Sander, & Xu, 1996) was used to cluster the points based on their density in feature space and the points with low density in feature space were discarded as noise. In this paper,
a Version of DBSCAN proposed by Sawant (2014), which can handle clustering of datasets with multiple densities, was used.

Four local plane parameters calculated for each point were constituted the 4D clustering feature space. Since the normal vector in Equation 1 is a unit vector, we normalised the parameter "d" in range of [-1, 1], to be comparable to other parameters of normal vector in DBSCAN clustering. The population of each cluster was calculated, and clusters with a population less than 10% of the average population were eliminated as the non-robust cluster. Finally, a plane was fitted to the points of each cluster using RANSAC.

The set $CL = \{cl_i, i = 1, 2, \ldots, m\}$ consist of "m" robust clusters extracted from TPC, in which $cl_i = \{p_j, j = 1, 2, \ldots, n_i\}$ is the $i$-th cluster contained "n_i" points. $(n_x, n_y, n_z, d_j)$ is the local plane parameter of point "p_j". The set $C = \{(n, d), i = 1, 2, \ldots, m\}$ consist of "m" robust planes fitted to "m" clusters of "CL". Then "n_i" and "d_i" are the normal vector and the distance from origin of plane fitted to "cl_i", respectively.

**TPC façade plane detection and tilt removal**

The most points of the TPC are related to the walls, façade and other vertical structures (e.g. trees or man-made objects). This is due to the narrow FOV of the camera and the nearly orthogonal principal axis of imaging to such structures in terrestrial imaging. In other hand, although z-component of the normal vector of a plumb façade plane is zero in WCS, in effect of an unknown tilt, it has different values for façades with different orientations in CCS. However, they have a significant geometric relationship to each other, which can be used to discriminate them from others. Normal vector of façade planes are...
geometrically located on a specific plane, which can be used as the basis of tilt removal. Figure 5 shows an example simple cube in a tilted coordinate system and coplanar normal vector of its facades.

Although there may be some inclined façade in urban areas, most walls and façades are constructed in alignment with plumb line direction (within an acceptable tolerance). The “façade plane” in this paper refers to planes that are aligned with the plumb line direction. Therefore, this section aims to categorise the planes of set “C” into multiple groups with coplanar normal vectors using the co-planarity and co-linearity conditions. Because most of the extracted robust planes from TPC are related to the facades, the most member group will represent the façade group. In the proposed method for this purpose, the normal vectors of all robust planes extracted from TPC are tested using the co-linearity and co-planarity conditions and are grouped in a repetitive procedure (the detailed procedure is presented in Algorithm 1).

Despite the rigorous strategy adopted in extracting the robust planes from TPC, calculated normal vector for each plane may deviate slightly from its original direction, which together with some low-inclined facades can result in over- or under-grouping of the façade planes. Therefore, in this paper, the result of the TPC clustering into the robust planes is used to determine the uncertainty of the calculated normal vectors, and this uncertainty is involved in the co-linearity and the co-planarity conditions (the detailed procedure is presented in Algorithm 2). Finally, least-square plane fitting was used to calculate the best fitted plane to the normal vectors of façade planes.

In Algorithm 1, vectors “n_a” and “n_b”, which are selected randomly from set “N”, are tested using their cross product as the co-linearity condition. If these two vectors are co-linear, their dual selection goes to the forbidden list so that they will not be among the random choice of the next normal vector selection, and then another two random vectors will be chosen. If these two random vectors are not co-linear, all other vectors of “N” will be checked for being coplanar with vectors “n_a” and “n_b”. To check the co-planarity, the inner product of vector “C_r” and other vectors of “N” is used as co-planarity condition. All of the

---

**Algorithm 1: Discriminating the façade and non-façade planes**

Input: \( N = \{ n_1, n_2, \ldots, n_m \} \) /* “n_i” is the normal vector component of the \( i \)-th member of “C” (calculated in previous section)/

Output: Façade planes group (Fac)

1. \( i = 1 \) /* Group Identification/ 
2. BEGIN
3. \( G_i = \{ \} \) /*ith group of planes/
4. \( \{ n_a, n_b \} \leftarrow \text{Random Selection of two members of } N \)
5. \( C_r = | n_a \times n_b | / \) /*The cross product of “n_a” and “n_b”/
6. IF \( C_r > \text{th}_\text{prod} / \) /*The exact value of this Threshold (th_prod) has been determined through Algorithm 2/ 
7. . FOR all members of N 
8. . . IF \( C_r n_j < \text{th}_\text{prod} \) THEN add \( n_j \) to \( G_i \) and remove \( n_j \) from N 
9. . END 
10. i = i + 1 
11. END 
12. DO step 1 to 11 UNTIL numel(N)\neq 0 or all possible cross product of N members be less than “Th_prod”/numel = number of elements/ 
13. Fac ← \( \{ G_i | \text{numel}(G_i) > \text{numel}(G_j), s = 1,2,..,i-1 \} \) 
14. END

---

Figure 5. An example of the relationship between the four normal vectors of the façade planes (\( N_1-N_4 \)) in a tilted CCS and the z-axis of the WCS (\( Z_{WCS} \)).
coplanar vectors are grouped together in a repetitive process.

Ideally, for both abovementioned conditions, the threshold “\( \theta_{\text{prod}} \)” should be equated with zero. However, due to the uncertainty of the calculated normal vectors, choosing the exact zero value for this threshold will not have a logical result for façade plane grouping. For this purpose, TPC clustering results were used for calculating this threshold in accordance with the characteristics of the extracted planes. In Algorithm 2, for each cluster in “CL”, the farthest point is first searched for the cluster centre by calculating the 4D distances based on the four plane parameters. Then, the inner product of its normal vector with the normal vector of the farthest point of the same cluster is calculated. Finally, the average of all calculated values is considered as the threshold “\( \theta_{\text{prod}} \)”.

After detecting the co-planar façade planes (Fac), best-fitted plane to the normal vectors of facades was calculated using least-square method. Normal vector of this plane \( (M_o = [m_x, m_y, m_z]) \) can be regarded as the z-axis of the CCS, and tilt can be removed by aligning it with zenith direction (Figure 5). Indeed, \( M_o \) forms the basis of the 3D rotation matrix for transforming the CCS of the TPC into a new coordinate system, which its z-axis is directed with the zenith direction (Equation 2).

\[
R_{ti} = [\text{null}(M_o^T) \quad M_o^T] \tag{2}
\]

In Equation 2, “\( M_o \)” which is used as the z-axis of the tilt removed coordinate system, forms the third column, and “\( \text{null}(M_o^T) \)” forms the first two column of the tilt removal rotation matrix \( (R_m) \). The “\( \text{null}(M_o^T) \)” is an orthonormal basis for the null space of “\( M_o \)” obtained from the singular value decomposition and create an orthogonal right handed coordinate system for TPC on the basis of “\( M_o \)” as the z-axis. Finally, Equation 3 will transform the TPC into a new tilt removed coordinate system denoted by index “\( T_1 \)”. In this equation, the raw TPC is denoted by index “\( T_0 \)”.

\[
\begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix}
_{T_1} =
R_{ti}
\begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix}
_{T_0} \tag{3}
\]

**Algorithm 2:** Calculating the threshold \( \theta_{\text{prod}} \)

**Input:** \( N = [n_1, n_2, . . . , n_m]^{\times n_i} \) “\( n_i \)” is the normal vector component of the i-th member of “C” (calculated in previous section)/
\( \max_p = \{\max_{n_1}, \max_{n_2}, . . . , \max_{n_m}\} \) “\( \max_{n_i} \)” is the normal vector of the farthest point of the cluster “\( C_l \)” from its centre

**Output:** \( \theta_{\text{prod}} \)

1 BEGIN
2 FOR all members of “C”
3 . Calculate \( \theta_h = \text{abs}(n_i \cdot \max_{n_i}) \)
4 END
5 \( \theta_{\text{prod}} = \sum \theta_h \)
6 END

Aligning orientation of the TPC z-axis to the zenith

Because in this study no auxiliary data is used to align the axes of the TPC coordinate system relative to the WCS, z-axis of TPC may be inverted relative to zenith direction after tilt removal using Equation 3. Therefore, in order to correctly orient the z-axis of TPC, ground planes were detected among the non-façade planes, and their altitude information was compared with the façade planes (Fac). Algorithm 3 shows the detailed process of determining the positive direction of z-axis.

In Algorithm 3, among the non-façade planes, those that are horizontal within a predefined threshold “\( \theta_{\text{prod}} \)” (which was calculated in Algorithm 2) were considered as the ground planes. For both the façade and non-façade groups, the average height of the cluster centre points was calculated. If the calculated average height is higher for façade group, the z-axis of the TPC coordinate system is correctly oriented with zenith, and otherwise it must be reversed, by 180° rotation around the x-axis. It should be noted that for aligning the positive direction of the z-axis with zenith, we could use the rotation around y-axis instead of the x-axis. Equation 4 will transform the tilt-removed TPC into the z-axis oriented coordinate system, denoted by index “\( T_2 \)”.

\[
\begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix}
_{T_2} = R_{tx}
\begin{bmatrix}
    x \\
    y \\
    z
\end{bmatrix}
_{T_1} \tag{4}
\]

**Algorithm 3:** Aligning orientation of the z-axis to the zenith

**Input:** \( N = [n_1, n_2, . . . , n_m]^{\times n_i} \) “\( n_i \)” is the normal vector component of the i-th member of “C” (calculated in previous section)/
Fac* façade planes group/
\( \theta_{\text{prod}} * \) threshold calculated in algorithm 2/

**Output:** Z-axis alignment angle (\( \Omega_z \))

1 BEGIN
2 IF \( |N_{\text{Fac}}| < \theta_{\text{prod}} \)
3 . END
4 ELSE \( \Omega_z = 0 \)
5 END
6 \( \Omega_z = 0 \)
7 Calculate \( \sum Z_{g_i}, \text{ for } \{G_{f_i}, i = 1, 2, . . . , \text{fac} \} \) and \( Z_{g_i} \) is the z component of the i-th cluster center of façade planes groups (Fac)/
8 Calculate \( \sum Z_{g_i}, \text{ for } \{G_{g_i}, i = 1, 2, . . . , \text{ground} \} \) and \( Z_{g_i} \) is the z component of the i-th cluster center of ground planes group (GrP)/
9 IF \( \sum Z_{g_i} > \sum Z_{f_i} \) THEN \( \Omega_z = 0 \)
10 ELSE \( \Omega_z = 0 \)
11 END
12 END
The results of the façade plane detection and tilt removal for TPC is shown in Figure 6(a,d) for datasets 1 and 2, respectively. For the first and second datasets, 16 and 27 façade planes were detected among all of the 21 and 44 extracted planes, respectively. As it is evident from this figure, with the strategy adopted in extracting planes from TPC, the points associated with each plane are not extracted with a high degree of completeness; and points related to the robust planes are detected precisely. For example, as it can be seen from Figure 6(a), the points associated with the columns do exist at regular intervals on the outer walls, as well as the boundary points are not detected as supporting points for their relevant planes. Indeed, the extended mathematical model for horizontal registration (will be introduced in Section Horizontal registration of APC and TPC) does not depend on the complete detection of the planes.

**Extracting façade lines from TPC and APC**

The core of each registration is the extraction of common features from each data that can be used as prospective matching candidates (Safdarinezhad, Mokhtarzade, & Valadan Zoej, 2016). Urban areas are rich of linear features (Habib, Asmamaw, May, & Kelley, 1999), and the linear feature representation scheme could be easily chosen to simplify the registration process (Al-Durgham & Habib, 2014). Therefore, in urban areas, linear features are more convenient to use as primitive in point cloud registration. In other hand, larger vertical distortions in image-based point clouds are encountered in areas where sudden changes in elevation are occurred. As a result, 3D building outlines are less reliable to use as primitive. However, building façades coincide in both aerial and terrestrial views, unlike building outlines that in effect of overhanging eaves are shifted relative to each other. Therefore, in this paper, the 2D façade lines were used for horizontal registration instead the common approaches of using building outlines.

Extracting linear features from point cloud and representing them in a convenient form is an ongoing topic of research, and there are many reports about the transformation estimation using linear features. Most of the proposed methods are based on some assumptions that undermine their comprehensive- ness, such as: endpoint or midpoint correspondence (Alshawa, 2007) (Habib, Ghanma, & Tait, 2004) (Fangning & Habib, 2016), prior information about the unknown transformation parameters (Al-Durgham & Habib, 2014) (Renaudin, Habib, & Kersting, 2011) (Habib et al., 2004), existence of overlap between corresponding lines (Poreba & Goulette, 2015) (Al-Durgham & Habib, 2014), length consistency of extracted conjugate lines (Cheng, Wu, et al., 2015b) (Poreba & Goulette, 2015), lack of scale estimation (Al-Durgham & Habib, 2014) (Hansen, Gross, & Thoennessen, 2008). Indeed, previous line-based methods depend highly on complete detection and extraction of linear features from point clouds that may fail due to the surrounding objects near the building façades (e.g. trees).

Accordingly, it is obvious that the scheme of representing the linear features is of great importance in estimating the transformation parameters of two datasets. In this paper, parameters of 2D façade lines
are used as primitive for horizontally registering the APC and TPC. In this regard, Wu et al. (2014) proposed a slope-intercept representation scheme for line-based point cloud registration. They used the slope-intercept scheme for representing the 2D outlines. However, their proposed parameter-based model for registration fails when there is scale difference between datasets. In other hand, lines perpendicular to the x-axis are not definable using such representation, and slope of nearly perpendicular to the x-axis lines are a very large numbers and their uncertainty are considerably propagated (Habib et al., 1999). Figure 7(a) shows the enormous error propagation for nearly perpendicular lines to the x-axis. The polar representation of a line is shown in Figure 7(b).

In this paper, to overcome the defects of common line-based methods, a polar form was used for representing the extracted 2D façade lines based on Equation 5, and the mathematical model for horizontal registration was extended based on the polar parameterised facade lines. In this equation, "r" is the distance of the façade line from the origin, "θ" is the angle that the segment "r" makes with the positive direction of the x-axis and "(x,y)" is the Cartesian coordinate of a point, which x = r cos θ and y = r sin θ (Figure 7(b)).

\[ r = x \cos(\theta) + y \sin(\theta) \]  \hspace{1cm} (5)

Using such a form of line representation, unlike the usual slope-intercept form, any 2D line is definable without destructive effects resulted from enormous error propagation of extracted lines. In other hand, by converting the extracted lines into their corresponding polar parameters, the line-based matching and registration can be accomplished in an unconstrained and unambiguous space. For example, extracted lines from both dataset can be matched and registered without any constraint of their relevant points as well as the ambiguities caused by the nearly perpendicular lines to the x-axis (which do abundantly exist in urban areas) (Figure 7(a)). Detailed information about the extended mathematical model for horizontal registration based on the polar parameterised facade lines is presented in Section Horizontal registration of APC and TPC.

**Extracting façade lines from TPC**

This section aims to extract the polar parameters (r, θ) of 2D facade lines from TPC. Given the parameters of robust facade planes after tilt removal (calculated in Section TPC tilt removing), the polar parameters of 2D façade lines will be simply obtained through projecting the façade planes onto x-y plane.

Supposing that \( C_{Fac} = \{(n_f, d_f), i = 1, 2, ..., f\} \) consisted of the parameters of facade planes related to the group “Fac”, in which “(n_f)” and “(d_f)” are the normal vector and the distance from origin of the i-th facade plane. Simply, \( \{(r_f = d_f), i = 1, 2, ..., f\} \) in which \( \{(r_f), i = 1, 2, ..., f\} \) is the distances of the corresponding façade line from the origin. Algorithm 4 shows the detailed procedure of calculating the parameter “θ” for each façade line from the corresponding façade plane parameters.

Figure 8 shows the 16 and 27 extracted facade lines from datasets 1 and 2, respectively.

**Extracting façade lines from APC**

This section aims to extract the 2D facade lines located on building facades and walls. The main steps of the proposed method for extracting façade lines from APC are shown in Figure 9.

**Grid-based detection of candidate façade areas.** To confine the search space towards the candidate facade points, APC was structured in a 2D grid, with a cell-size of 1m × 1m. Because the walls and building facades are narrow linear features and APC is

---

**Figure 7.** (a) An example of the enormous error propagation for nearly perpendicular lines to the x-axis using the slope-intercept representation; (b) Representing the example line in polar form and the non-exacerbation of its error propagation.
sufficiently dense, choosing the \(1m \times 1m\) cell-size will guarantee laying such features in cells with an appropriate degree of freedom to detect the areas containing the façade points. Indeed, increasing the cell-size will result in a larger search space for façade points.

In each cell, the maximum height difference between the highest and lowest point was calculated, and those with a value greater than a pre-defined threshold were selected as the candidate façade areas. This threshold was set to “1.5m” due to the existence of some walls to a minimum height of 1.5m.

3.2.2.2. Normal vector-constrained detection of candidate façade points in the candidate façade areas

Because in the geo-referenced APC (WCS) most of the walls and building façades are plumb, most of the façade planes have level normal vector. Accordingly, candidate façade points can be detected through applying a level constraint on the estimated normal vector for each point in the candidate façade areas. The steps of the proposed method for this purpose are as follows:

1. Constructing a spherical neighbourhood with radius “\(R_s\)” for each point in candidate façade areas.
2. RANSAC plane fitting to the points within each neighbourhood containing more than five points.
3. Labelling the points with level normal vector, within the precision range defined by threshold “\(th_{nv}\)”, as the candidate façade points (the

![Figure 8](image-url)

**Figure 8.** 2D façade line extraction from TPC in polar form. (a) Dataset 1; (b) Dataset 2.

![Figure 9](image-url)

**Figure 9.** The main steps of the proposed method for extracting façade lines from APC.
details of how this threshold is calculated is given below).

The radius “$R_r$” in the above steps defines the size of the neighbourhood in searching for candidate façade points. This parameter is related to density of APC on façades, which vary greatly from high to low altitudes, and was set to “0.50m”, empirically. With a radius of “0.50m”, most of the neighbourhoods in upper surfaces of the façades will contain more than five façade points, although it may not be so at their lower surfaces. Nevertheless, increasing the size of the neighbourhoods may include some non-façade points, which leads to the over detection of candidate façade points.

In the third step of the proposed method for candidate façade point detection, points where the absolute z-component of their estimated normal vector are less than a predefined threshold “$\theta_{nv}$” are considered as candidate façade. Error propagation law (Hirvonen, 1971) was employed to estimate the threshold “$\theta_{nv}$” based on accuracy of APC.

Given three points $P_1, P_2$ and $P_3,$ in the neighbourhood of a consecutive distance of “r” from each other, the z-component of their normal vector, obtained from cross product of two vectors $U = P_1P_2 = (u_1, u_2, u_3)$ and $V = P_1P_3 = (v_1, v_2, v_3),$ is calculated by Equation (6).

$$n_z = u_1v_2 - u_2v_1 \quad (6)$$

By applying the error propagation law to the Equation 6, the uncertainty of the calculated z-component of a normal vector is derived according to Equation 7.

$$\delta^2_{n_z} = \delta^2_{u_1} v_2^2 + \delta^2_{u_2} u_1^2 + \delta^2_{u_3} v_2^2 + \delta^2_{v_1} u_1^2 \quad (7)$$

In Equation 7, the uncertainty of horizontal components of the U and V can be considered equal to the uncertainty of horizontal position of a point in APC ($\delta_p$). In other hand, by assuming both the horizontal components of the vectors U and V equally, Equation 7 will be rewritten according to the Equation 8.

$$\delta_{n_z} = \frac{2R_r\delta_p}{\sqrt{2}} \quad (8)$$

Therefore, given the horizontal accuracy of APC ($\delta_p$) and the neighbourhood radius ($R_r$), threshold “$\theta_{nv}$” can be calculated by Equation 8. Since both APCs used in this paper have a horizontal accuracy of 0.08m, then threshold “$\theta_{nv}$” was calculated equal to “0.056” for both APCs according to Equation 8. Figure 10 shows the results of detecting the candidate façade areas and the candidate façade points.

**Hough-based façade line detection and RANSAC-based façade line extraction.** As it is evident from Figure 10(c,d), some non-façade points are extracted as candidate façade point, which are mainly related to the trees and other urban elements. However, façade points are distributed on the façade planes in different elevations, and their projection on X-Y plane will result in a high density along 2D façade lines, while the outliers (e.g. trees-related points) do not follow the linear distribution with a high altitude density. Therefore, by projecting the candidate-faced points onto the X-Y plane, each 2D façade line will be supported by a large number of points located on the corresponding façade plane. Because the purpose of this section is to extract polar parameters of 2D façade lines from APC, a simple Hough transform (Duda & Hart, 1972), which is based on voting on parameter space, is an appropriate choice.

The façade points, which are distributed in the direction of 2D façade lines with much higher density and quantity than the outlier points, will vote for the respective façade lines with a greater higher number.

Nevertheless, estimating the line parameters based on the Hough transform is prone to inaccuracies depending on the binning size. In this paper, in order to overcome this problem, the method proposed by Dalitz, Schramke, and Jeltsc (2017) is used for extracting the 2D facade lines from candidate facade points. The set “$AF_p$” contains candidate façade points, which are mapped on the X-Y plane. Extracting the 2D façade lines from “$AF_p$” was done according to the following steps:

1. Binning the Hough space over the $(r, \theta)$ (with cell-size of “$\Delta r$” and “$\Delta\theta$”).
2. Selecting the bin with the highest vote as the initial line “$(r, \theta)_{ini}$”.
3. Calculating the perpendicular distance for any point in “$AF_p$” to the initial line “$(r, \theta)_{ini}$” and finding points $L_{ini} \subseteq AF_p$ where their distances are less than “$\Delta r$”.
4. Calculating the best fitted line “$(r, \theta)_{best}$” to the points of set “$L_{ini}$” using RANSAC.
5. Calculating the perpendicular distance for any point in “$AF_p$” to the “$(r, \theta)_{best}$” and finding points $L_{fin} \subseteq AF_p$ where their distances are less than “$\Delta r$”; and removing the points “$L_{fin}$” from “$AF_p$”.
6. Repeat the steps 2 to 6 until “$AF_p$” contains 20% of its initial points.

The abovementioned implementation has two modifications to the proposed method in (Dalitz et al., 2017). Input points and output lines are in 2D space.
in this paper, which is less complicated than 3D space; and RANSAC is used for calculating the best fit line in step 5, instead of fitting a plane to all points of \(L_{ini}\). Indeed, this choice was made in order to prevent the interference of noises in calculating the line \((r, \theta)_{best}\). In this way, Hough transform is responsible for detecting the points associated with the strongest line, existing in each repetition of the algorithm, and fitted line to these points is obtained through RANSAC line fitting. Using such strategy, 2D façade lines will be robustly extracted, unlike some districts full of trees, which have low density while projecting them onto X-Y plane.

The cell-sizes \(\Delta r\) and \(\Delta \theta\) regulate the binning size of the Hough space in façade line extraction from the candidate façade points. Choosing large values for these parameters will result in selection of more support points in \(L_{ini}\) for each line, which may include

![Figure 10](image-url)  
**Figure 10.** Candidate façade point extraction from APC. (a) Detected candidate façade areas in APC of dataset 1; (b) Detected candidate façade areas in APC of dataset 2; (c) Detected candidate façade points in APC of dataset 1; (d) Detected candidate façade points in APC of dataset 2.

![Figure 11](image-url)  
**Figure 11.** 2D façade line extraction from APC in polar form. (a) Dataset 1; (b) Dataset 2.
some outliers. However, because in this paper RANSAC is used for line extraction (in step 5), there is a relatively less sensitivity to these parameters than the standard Hough transform. In this paper, based on some experimental tests, “Δr” and “Δθ” were set to “0.15m” and “0.5°”, respectively. Figure 11 shows the extracted 2D façade lines from APC.

**Horizontal registration of APC and TPC**

Two sets \( A = \{ (r_A, \theta_A), i = 1, 2, ..., n_A \} \) and \( T = \{ (r_T, \theta_T), i = 1, 2, ..., n_T \} \) represent the extracted 2D façade lines from APC and TPC, respectively. In this study, 2D Similarity mathematical model was used as the basis of horizontal registration (Equation 9). The source coordinate system (o-xyz) could be transformed to the reference coordinate system (O-XYZ), according to this model:

\[
\begin{bmatrix} x \\ y \end{bmatrix} = \lambda \begin{bmatrix} \cos \alpha & \sin \alpha \\ -\sin \alpha & \cos \alpha \end{bmatrix} \begin{bmatrix} X \\ Y \end{bmatrix} + \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix}
\]  

(9)

Where \((x, y)\) and \((X, Y)\) are the terrestrial and the aerial coordinates, \(\lambda\) is the scale factor and “\(\alpha\)" is the in-plane rotation. Lines of two sets “\(A\)” and “\(T\)” are represented using Equations 10 and 11, respectively.

\[
r_A = X \cos \theta_A + Y \sin \theta_A
\]  

(10)

\[
r_T = X \cos \theta_T + Y \sin \theta_T
\]  

(11)

Equation 12 is derived from expansion of Equation 9:

\[X = \lambda(X\cos\alpha + Y\sin\alpha) + \Delta x \ y = \lambda(Y\cos\alpha - X\sin\alpha) + \Delta y
\]  

(12)

Equation 13 is derived from substituting the Equation 12 in Equation 11:

\[
r_T = \lambda \left\{ X(\cos \theta_T \cos \alpha - \sin \theta_T \sin \alpha) \right. \\
+ Y(\cos \theta_T \sin \alpha + \sin \theta_T \cos \alpha) \right. \\
+ \Delta x \cos \theta_T + \Delta y \sin \theta_T
\]  

(13)

Equation 13 is converted to Equation 14 using some trigonometric functions:

\[
r_T = \lambda \left\{ X\cos(\theta_T + \alpha) + Y \sin(\theta_T + \alpha) \right. \\
+ \Delta x \cos \theta_T + \Delta y \sin \theta_T
\]  

(14)

Equation 14 is converted to Equation 15 using some mathematical operations:

\[
\frac{r_T - (\Delta x \cos \theta_T + \Delta y \sin \theta_T)}{\lambda} = X \cos(\theta_T + \alpha) + Y \sin(\theta_T + \alpha)
\]  

(15)

Finally, Equations 16 and 17 are derived using comparison of Equations 15 and 10:

\[
r_A = \frac{r_T - (\Delta x \cos \theta_T + \Delta y \sin \theta_T)}{\lambda}
\]  

(16)

\[
\theta_A = \theta_T + \alpha
\]  

(17)

Using Equation 16, three parameters “\(\lambda\)”, “\(\Delta x\)” and “\(\Delta y\)” will be estimated using at least three pairs of conjugate lines. Using Equation 17, “\(\alpha\)” will be estimated independently of other three parameters using at least one conjugate line. Since the number of lines in sets “\(A\)” and “\(T\)” are not the same and their mutual correspondence is not given, similar to the method proposed by Cheng et al. (2013), line matching and horizontal registration were done using a try and error scheme. In this paper, the independence of the parameter “\(\alpha\)” from the other three parameters of the horizontal registration was used as a constraint to line matching, which could reduce the search space. For this purpose, rotational consistency of each pair of selected lines is estimated on the basis of the Equation 17. The proposed method for simultaneous matching and horizontal registration is as the following steps:

(1) Selecting three random lines from “\(A\)” and “\(T\)”, respectively.
(2) Calculating three rotation parameters \((a_i, i = 1, 2, 3)\) for these three matching candidate.
(3) Calculating the mean (\(\alpha\)) and standard deviation (\(\sigma_{\alpha}\)) of these three estimated rotations (Equation 18).

\[
\sigma_{\alpha} = \sqrt{\frac{1}{2} \sum_{i=1}^{3} (a_i - \alpha)^2} = \frac{1}{3} \sum_{i=1}^{3} a_i
\]  

(18)

(1) Repeating the steps 1 to 3 while the standard deviation of the estimated “\(\sigma_{\alpha}\)” exceeds from a predefined threshold (5° in this study); otherwise, calculating the initial horizontal registration parameters \((X_{ini} = \{\alpha, \lambda, \Delta x, \Delta y\})\) using Equations 16 and 17.
(2) Converting the lines of set “\(T\)” to “\(T_{ini}\)” using “\(X_{ini}\)”.
(3) Counting the number of the correctly matched lines between two sets of “\(T_{ini}\)” and “\(A\)” using a predefined distance threshold “\(th_{dis}\)”.
(4) Accepting the “\(X_{ini}\)” as a reliable estimation \((X_{rel})\) if the number of correctly matched lines exceeds from a pre-defined threshold (was set to 5 in this study, experimentally); otherwise, the steps 1 to 6 are repeated.
(5) Converting the lines of set “\(T\)” to “\(T_{rel}\)” using “\(X_{rel}\)”.
(6) Selecting the correctly matched lines between “\(A\)” and “\(T_{rel}\)” based on the distance threshold “\(th_{sel}\)”;
and constituting the two sets “LA” and “LT” with corresponding lines of “A” and “T”, respectively.

(7) Calculating the final horizontal registration parameters according to the conjugate lines of two sets “LA” and “LT”.

In the above steps, the distance thresholds \( \text{th}_{\text{dis}} \) and \( \frac{1}{2} \text{th}_{\text{dis}} \) determines the consistency of the matched lines after transformation by “X_{ini}” and “X_{rel}” in two phases of “initial matching” (denoted by the index “ini”) and “reliable matching” (denoted by the index “rel”), respectively. Equation 19 defines the threshold \( \text{th}_{\text{dis}} \). In this equation, “\( \delta r \)” and “\( \delta \theta \)” represent the calculated discrepancies in “r” and “\( \theta \)”, respectively.

\[
\text{th}_{\text{dis}} = \begin{cases} 
\delta r < \text{th}_r \\
\delta \theta < \text{th}_\theta 
\end{cases} 
\]  

In Equation 19, “\( \text{th}_r \)” and “\( \text{th}_\theta \)” represent the thresholds for distance between matched lines in “r” and “\( \theta \)” directions, respectively. Choosing the small values for these thresholds may lead to obtain a few numbers of matched lines. More matched lines will be obtained through setting the larger values for these thresholds, which may include some outliers. Nevertheless, using the 2D façade lines as primitive for horizontal registration and robustly extracting the façade planes from both datasets, it is not expected to extract the approximately parallel façade lines with distances closer than “0.30m” from each dataset. Therefore, for the “initial matching” two lines are considered correctly matched if their distance is less than “0.30m” (\( \text{th}_r \)), and their deviation from being parallel is less than “5\(^{\circ}\)” (\( \text{th}_\theta \)). To prevent the contribution of weakly matched lines, these thresholds are reduced by a factor of “3” in “reliable matching”.

### Vertical registration of APC and TPC

Since APC has an overhead view, then the ground planes extracted from TPC will probably have corresponding in APC. In other hand, APC and TPC are horizontally aligned upon each other after applying the horizontal registration. Therefore, to calculate the vertical shift, ground planes (\( \text{GrPl} \)) extracted from TPC (Algorithm 3) was considered as reference and its corresponding was obtained by clipping their horizontal common areas in APC. Points of the “\( \text{GrPl} \)” form the setPT = \{PT\( _i \), i = 1, 2, ..., nPT\}, and then the dominant vertical shift between the APC and TPC was calculated as the following steps:

1. Searching the nearest neighbour between each point “PT\( _i \)” and each point of APC by calculating the horizontal Euclidean distances and forming the corresponding points of APC and TPC (\( \text{NP} = \{PA_iPT\_i, i = 1, 2, ..., n_{PT}\} \)).
2. Calculating the height difference for each pair “PA\( _iPT\_i \)” and forming the set \( \Delta H = \{\Delta H\_i, i = 1, 2, ..., n_{PT}\} \) consisted of the height differences of each pair of “NP”.
3. Applying the vertical shift “\( \Delta H_{\text{rel}} \)” to each point of the TPC in order to vertically register it to the APC.

### Accuracy assessment and discussion

In this paper, the seven transformation parameters between APC and TPC are estimated in three phases. Table 2 lists the values of these parameters for both datasets. It should be noted that in the proposed method for TPC tilt removal, x- and y-components of tilt angle were not estimated independently and tilt was removed on the basis of the defining a new z-axis for TPC. Therefore, in order to show the x- and y-components of the tilt angle, which is equivalent to the two transformations applied by Equations 3 and 4 in this paper, Equation 20 was solved between the raw TPC (denoted by index “T\(_0\)” and the tilt-removed TPC (denoted by index “T\(_2\)” and the estimated values for rotation around x- and y-axis were inserted in Table 2 as a representation of rotation around x- and y-axis (\( \Omega \) and \( \Phi \) in Table 2, respectively). In this equation, “\( R_x \)” and “\( R_y \)” are the 3D rotation matrices defined for rotation around x- and y-axis with rotational elements of \( \Omega \) and \( \Phi \), respectively.

\[
\begin{bmatrix} x \\ y \\ z \end{bmatrix}_{T_2} = R_yRx \begin{bmatrix} x \\ y \\ z \end{bmatrix}_{T_0}
\]  

\( \delta r \) and \( \delta \theta \) are the 3D \( \Omega \) and \( \Phi \) with horizontal common areas in APC. Points of the “\( \text{GrPl} \)” form the setPT = \{PT\( _i \), i = 1, 2, ..., nPT\}, and then the dominant vertical shift between the APC and TPC was calculated as the following steps:

**Table 2.** The summary of transformation parameters for TPC to APC registration.

| parameters | \( \lambda \) (deg) | \( \Omega \) (deg) | \( \Phi \) (deg) | \( K \) (deg) | \( \Delta X \) (m) | \( \Delta Y \) (m) | \( \Delta H \) (m) |
|-----------|-----------------|-----------------|-----------------|-------------|----------------|----------------|----------------|
| Dataset 1 | 2.74            | 86.79           | 12.05           | 65.62       | 309,346.06     | 4,091,344.75   | 85.14          |
| Dataset 2 | 3.15            | 271.26          | 10.33           | 15.25       | 309,212.02     | 4,091,114.22   | 80.68          |
and “3.15” for datasets 1 and 2, respectively. The results showed that the proposed method is able to robustly estimate the transformation parameters, even when the two datasets have a great deal difference in term of coordinate system. Figure 12 shows the visual registration results for datasets 1–2.

Two different approaches are used for accuracy assessment, and in each approach, horizontal and vertical accuracies are evaluated separately. In the first approach, in order to compare the proposed method in this paper with the most common method of point cloud registration, a manual registration based on the precise selection of corresponding points from APC and TPC, followed by the well-known ICP algorithm, is implemented. Registered TPC by manual/ICP method is used as reference data to compare with the registered TPC by the proposed method in this paper.

Table 3 shows the estimated discrepancies between the TPCs registered by the proposed and the manual/ICP methods. In horizontal comparison, 15 and 16 check areas located on the facades were selected from dataset 1 and 2, respectively. Number of check areas in vertical comparison, selected from ground regions, is 15 and 17 for datasets 1 and 2, respectively. Each check area is a spherical neighbourhood with a radius of 0.5 m. Two planes were fitted to each check area using RANSAC; one is related to the registered TPC by the proposed method and another is related to the TPC registered by the manual/ICP method. The distances between each conjugate plane were calculated and used as a representation of the discrepancy between the TPC registered by the proposed and the manual/ICP methods.

As the second approach of the accuracy assessment, the discrepancies between the registered TPC and the APC were estimated in location of several check regions. For horizontal accuracy assessment, four common regions in APC and TPC (regions A1–D1 in dataset 1 (Figure 13) and A2–D2 in dataset 2 (Figure 14)), located on the façades and walls, were selected from both datasets. The distances between the APC and the Triangulated Irregular Network (TIN) structured TPC were calculated in location of each check region. The visual results of this comparison are showed for each check region in Figure 13 and 14 for datasets 1–2, respectively; and Table 4 lists the corresponding statistics results. The third column of

![Figure 12. Visual result of TPC to APC registration. (a) Dataset 1. (b) Dataset 2.](image_url)

| Horizontal comparison | Dataset | Max | Mean | RMSE |
|-----------------------|---------|-----|------|------|
| Horizontal discrepancy (m) |         |     |      |      |
| (1)                   | 0.21     | 0.11| 0.12 |
| (2)                   | 0.22     | 0.15| 0.18 |

| Vertical comparison | Dataset | Max | Mean | RMSE |
|---------------------|---------|-----|------|------|
| Vertical discrepancy (m) |         |     |      |      |
| (1)                  | 0.17     | 0.13| 0.14 |
| (2)                  | 0.16     | 0.11| 0.13 |
Table 4 (APC points (No.)) shows the number of the points existed in each check region of APC, and fourth column (APC/TPC (%)) shows the ratio of the number of the APC points to the TPC points in each check region. The values of these two columns indicate that there is a big difference between the density of points on the façades in APC and TPC. In these figures, the error distribution is nearly uniform and illustrates that the registration errors are distributed evenly.

In second approach of the accuracy assessment, to evaluate the vertical consistency of the registered TPC with the APC, points of the extracted ground planes in Section Aligning orientation of the TPC z-axis to the zenith (GrPl) were considered as reference data, and distances between them and TIN constructed APC were calculated. Figure 15 shows the vertical discrepancies for both datasets. Detailed statistical results of the vertical discrepancies are showed in Table 5. To better present the distribution of discrepancies in Figures 13–15, cloud to cloud distances are displayed for TPCs.

The distribution of errors in Figures 13–15 shows that, although TPC is properly registered to APC, there are few errors with a relatively high difference compared to the mean error. This is due to the presence of a lot of noise and outlier in the image-based point clouds, especially at the step edges. In area "A₂" of Figure 14, errors are shown in the eave of a ceiling roof, which has multiple step edges in its components. As it is evident from this figure, the variance of errors varies greatly with the occurrence of step edges, while in other surfaces we see a relatively uniform accuracy.

**Conclusion**

This paper addresses the issue of registering the terrestrial and aerial image-based point clouds, to
achieve a complete and dense coverage on both top and side faces in urban areas. In image-based methods, point clouds are originally generated in the CCS, which in terms of scale and 3D rotations has a relatively large difference than WCS. Coping with this deficiency requires some manual operations or auxiliary data, which is operationally exhausting and time consuming. Aerial image-based point clouds can easily be geo-referenced using direct geo-referencing methods or GCPs, but it seems hard and time consuming in the case of terrestrial image-based point clouds, due to the limited FOV in imaging at ground levels, as well as the relatively smaller coverable areas.

Common methods in the field of aerial and terrestrial point cloud registration are designed for registering the laser-based datasets and consequently have less complexity regarding to the transformation parameters. In other hand, using image-based methods for generating the TPC, there is a less probability for

Table 4. Horizontal consistency of the registered TPC with the APC (second approach of accuracy assessment).

| Dataset | Area | APC points (No.) | APC/TPC (%) | Mean error (m) | Maximum error (m) | RMSE (m) | Standard Deviation (m) |
|---------|------|-----------------|-------------|----------------|-------------------|----------|------------------------|
| (1) A_1 | 433  | 4               | 0.08        | 0.26           | 0.09              | 0.05     |                        |
| B_1     | 2090 | 0.7             | 0.11        | 0.31           | 0.13              | 0.05     |                        |
| C_1     | 430  | 2               | 0.05        | 0.13           | 0.06              | 0.02     |                        |
| D_1     | 496  | 6               | 0.16        | 0.46           | 0.18              | 0.08     |                        |
| A_2     | 949  | 2               | 0.19        | 0.56           | 0.22              | 0.13     |                        |
| B_2     | 924  | 0.7             | 0.13        | 0.32           | 0.14              | 0.07     |                        |
| C_2     | 206  | 0.8             | 0.09        | 0.31           | 0.11              | 0.05     |                        |
| D_2     | 127  | 0.2             | 0.12        | 0.23           | 0.14              | 0.05     |                        |

Figure 14. A representation of horizontal discrepancies between the registered TPC and the APC in four check region selected on facades (Dataset 2).
an appropriate overlap between the TPC and the APC. This is due to the narrow FOV of camera in imaging at ground levels compared to the laser scanner, which has the ability to capture point cloud with a more altitude range. However, the use of image-based methods has the advantages, associated with its only dependency on low-cost and low-weight equipments.

A novel method is proposed for hierarchically estimating the seven transformation parameters between the APC and the TPC. The main contributions related to the type of the input data in the method proposed in this paper are: its independence from any initial user guidance or manual operation, its ability to accurately estimate the transformation parameters despite the significant differences between two coordinate systems and its ability to perform registration even when there is a minimal overlap between two point clouds (e.g. a narrow strip of overlap around the walls and façades).

The common feature extraction, as a key part of each registration, is a serious challenge for TPC to APC registration in this paper, because these two data have totally different views and there is a relatively low overlap between them. The tilt of the TPC, composed of the two rotations around x- and y-axis, was eliminated using a TPC-sufficient method, which does not rely on extracting common features from both datasets. In this paper, 2D façade lines, which are highly frequent features in urban and rural areas and coincident in both aerial and terrestrial views and will not shifted relative to each other in effect of the overhanging eaves, were used as primitive in horizontal registration. The main contribution of this paper, related to the

| Dataset | TPC points (No.) | Mean error (m) | Maximum error (m) | RMSE (m) | Standard Deviation (m) |
|---------|-------------------|----------------|-------------------|----------|------------------------|
| (1)     | 3249              | 0.07           | 0.34              | 0.08     | 0.05                   |
| (2)     | 1,184,932         | 0.08           | 0.29              | 0.08     | 0.03                   |

Figure 15. A representation of vertical discrepancies between the registered TPC and the APC in ground regions. (a) Dataset 1; (b) Dataset 2.

Table 5. Vertical consistency of the registered TPC with the APC (second approach of accuracy assessment).
horizontal registration, is the mathematical modelling of the registration in a novel polar parameterised form, which does not rely on detecting and extracting of a line with any constraint, while most of the previous line-based registration methods were based on some limiter assumptions. Indeed, the important characteristic of the proposed method for horizontal registration is its independence of detecting and extracting the façade points and subsequently façade planes and façade lines with a high degree of completeness, specifically when disturbing objects (e.g. trees) do interfere with the walls and building façades.

A comparison between the proposed method in this paper and the manual registration followed by ICP showed the superiority of the proposed method in cases where the overlap between two point clouds is very low. Furthermore, the proposed method, unlike ICP, does not require an initial alignment between two point clouds. The proposed method in this paper has a high potential to apply for the point clouds generated by other sensors and techniques (e.g. laser-based point clouds) with a few changes.

Future works

Registering the aerial and terrestrial image-based point clouds was investigated in this paper. The extension of the proposed method to the point clouds from different sources (include laser based and image based) is proposed for future research. Furthermore, for vast areas, it is not possible to generate the terrestrial image-based point clouds with preserving an acceptable accuracy. Therefore, registering the multiple local TPCs acquired from small areas to a global APC acquired from a vast area is suggested for future research.

Most of the existing methods do not address the errors in reference data, which can affect the estimation of registration parameters. Modelling such error sources in the process of registration is suggested for future research. For example, in accordance with the proposed method in this paper, considering an additional tilt removal step for aerial datasets is suggested for more consistency between the registered TPC with the APC.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Agisoft PhotoScan. (2017). Retrieved June 27, 2017 from http://www.agisoft.com/
Al-Durgham, M., & Habib, A. (2013). A framework for the registration and segmentation of heterogeneous LiDAR data. Photogrammetric Engineering and Remote Sensing Journal, 79, 135–145.
Al-Durgham, M., & Habib, A. (2014). Association-matrix-based sample consensus approach for automated registration of terrestrial laser scans using linear features. Photogrammetric Engineering and Remote Sensing Journal, 80(10), 23–33.
Alshawa, M. (2007). ICL: Iterative closest line - a novel point cloud registration algorithm based on linear features. Ekssentar, 10, 53–59.
Bang, K.I., Habib, A.F., Kusevic, K., & Mrstik, P. (2008). Integration of terrestrial and airborne LiDAR data for system calibration. Proceedings of the International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Beijing, China, 3–11 July 2008; pp. 391–398.
Besl, P.J., & McKay, N.D. (1992). A method for registration of 3-D shapes. IEEE trans. on pattern analysis and machine intelligence. Los Alamitos, CA, USA: IEEE Computer Society, 14(2), 239–256.
Brenner, C., Dold, C., & Ripperda, N. (2008). Coarse orientation of terrestrial laser scans in urban environments. ISPRS Journal of Photogrammetry and Remote Sensing, 63(1), 4–18.
Chen, Y., & Medioni, G. (1992). Object modelling by registration of multiple range images. Image and Vision Computing, 10(3), 145–155.
Cheng, L., Tong, L., Li, M., & Liu, Y. (2013). Semi-automatic registration of airborne and terrestrial laser scanning data using building corner matching with boundaries as reliability check. Remote Sensing, 5, 6260–6283.
Cheng, L., Tong, L., Wu, Y., Chen, Y., & Li, M. (2015a). Shiftable leading point method for high accuracy registration of airborne and terrestrial lidar data. Remote Sensing, 7, 1915–1936.
Cheng, L., Wu, Y., Tong, L., Chen, Y., & Li, M. (2015b). Hierarchical registration method for airborne and vehicle lidar point cloud. Remote Sensing, 7, 13921–13944.
Dalitz, C., Schramke, T., & Jeltsch, M. (2017). Iterative hough transform for line detection in 3D point clouds. Image Processing On Line, 7, 184–196.
Duda, R.O., & Hart, P.E. (1972). Use of the hough transformation to detect lines and curves in pictures. Communications of the ACM, 15(1), 11–15.
Ester, M., Kriegel, H., Sander, J., & Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD’96), Portland, USA, 1996.
Fangning, H., & Habib, F. (2016). A closed-form solution for coarse registration of point clouds using linear features. Journal of Surveying Engineering, 142(3), 1–14.
Favalli, M., Fornacai, A., & Paresci, M.T. (2009). LIDAR strip adjustment: Application to volcanic areas. Geomorphology, 111(3), 123–135.
Fischler, M.A., & Bolles, R.C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Commun. ACM, 24, 381–395.
Habib, A., Asmamaw, A., May, M., & Kelley, D. (1999). Linear features in photogrammetry, department of civil and environmental engineering and geodetic science, the ohio state university. Departmental Report # 451.
Habib, A., Ghana, M.S., & Tait, M. (2004). Integration of LIDAR and photogrammetry for close range
applications. Proceedings of the XXIth ISPRS Congress, Istanbul, Turkey, July 12–23, 2004.
Han, J.Y., Perng, N.H., & Chen, H.J. (2013). LiDAR point cloud registration by image detection technique. *IEEE Geoscience and Remote Sensing Letters*, 10, 746–750.
Hansen, W., Gross, H., & Thoennessen, U. (2008). Line-based registration of terrestrial and aerial LiDAR data. The *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 37 B3a, 161–166.
Hirvonen, R.A. (1971). Adjustments by least squares in geodesy and photogrammetry. New York: Ungar. 261. ISBN 0804443971, 978-0804443975.
Kang, Z., Li, J., Zhang, L., Zhao, Q., & Zlatanova, S. (2009). Automatic registration of terrestrial laser scanning point clouds using panoramic reflectance images. *Sensors*, 9, 2621–2646.
Kedzierski, M., & Fryskowska, A. (2014). Terrestrial and aerial laser scanning data integration using wavelet analysis for the purpose of 3D building modeling. *Sensors*, 14, 12070–12092.
Lee, J., Yu, K., Kim, Y., & Habib, A.F. (2007). Adjustment of discrepancies between LiDAR data strips using linear features. *IEEE Geoscience and Remote Sensing Letters*, 4, 475–479.
Musialski, P., Wonka, P., Aliaga, D.G., Wimmer, M., Van Gool, L., & Purgathofer, W. (2013). A survey of urban reconstruction. *Computer Graphics Forum*, 32, 146–177.
Poreba, M., & Goulette, F. (2015). A robust linear feature-based procedure for automated registration of point clouds. *Sensors*, 15, 1435–1457.
Renaudin, E., Habib, E., & Kersting, A.P. (2011). Feature-based registration of terrestrial laser scans with minimum overlap using photogrammetric data. *ETRI Journal*, 33, 517–527.
Rosnell, T., & Honkavaara, E. (2012). Point cloud generation from aerial image data acquired by a quadrocopter type micro unmanned aerial vehicle and a digital still camera. *Sensors*, 12, 453–480.
Rossi, P., Mancini, F., Dubbini, M., Mazzone, F., & Capra, A. (2017). Combining nadir and oblique UAV imagery to reconstruct quarry topography: Methodology and feasibility analysis. *European Journal of Remote Sensing*, 50, 1, 211–221.
Safdarinazhad, A., Mokhtarrzade, M., & Valadan Zoej, M.J. (2016). Shadow-based hierarchical matching for the automatic registration of airborne LiDAR data and space imagery. *Remote Sensing*, 8, 466.
Sawant, K. (2014). Adaptive methods for determining DBSCAN parameters. *International Journal of Innovative Science, Engineering & Technology*, 1(4), 329–334.
Skaloud, J., & Lichti, D. (2006). Rigorous approach to bore-sight self-calibration in airborne laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 61(1), 47–59.
Snively, N., Seitz, S.M., & Szeliski, R. (2008). Modeling the world from internet photo collections. *International Journal of Computer Vision*, 80, 189–210.
Teo, T.-A., & Huang, S.H. (2014). Surface-based registration of airborne and terrestrial mobile LiDAR point clouds. *Remote Sensing*, 6, 12686–12707.
Theiler, P., & Schindler, K. (2012). Automatic registration of terrestrial laser scanner point clouds using natural planar surfaces. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, 39(3), 86–91.
Wu, H., & Fan, H. (2016). Registration of airborne LiDAR point clouds by matching the linear plane features of building roof facets. *Remote Sensing*, 8(6), 447.
Wu, H., Scailion, M., Li, H., Li, N., Lu, M., & Liu, C. (2014). Feature-constrained registration of building point clouds acquired by terrestrial and airborne laser scanners. *Journal of Applied Remote Sensing*, 8(1), 083587(1)-083587(22).
Yang, B., & Zang, Y. (2014). Automated registration of dense terrestrial laser-scanning point clouds using curves. *ISPRS Journal of Photogrammetry and Remote Sensing*, 95, 109–121.
Yang, B., Zang, Y., Dong, Z., & Huang, R. (2015). An automated method to register airborne and terrestrial laser scanning point clouds. *ISPRS Journal of Photogrammetry and Remote Sensing*, 109, 62–76.