A MARKER-FREE CALIBRATION METHOD FOR MOBILE LASER SCANNING POINT CLOUDS CORRECTION

Bisheng Yang [1,*], Yuhao Li[1], Xianghong Zou[2], Zhen Dong [1]

[1]State Key Laboratory of Information Engineering in Surveying, Mapping and Remote Sensing, Wuhan University, Wuhan 430079, China - (bshyang, yhaoli, dongzhenwhu)@whu.edu.cn
[2]Autopilot Business Center, Tencent, Beijing, China - ericxhzou@tencent.com

Commission II, WG II/3

KEY WORDS: Mobile Laser Scanning, Point Clouds, Calibration, Feature Extraction, Global Optimization

ABSTRACT:

Mobile laser scanning systems (MLS) have been widely used in collecting three-dimensional point clouds for many applications, such as 3D mapping, road facilities inventory and high definition map. Although MLS is calibrated accurately to obtain precise locations of point clouds, it is still challenging to obtain precise locations of point clouds in the areas of GPS signal denied or narrow streets with high dense buildings, resulting in uneven position deviations of point clouds between the overlapping trajectory areas. In this paper, a marker-free calibration method is proposed to solve the above problems. The proposed method firstly partitions the trajectory into segments according to the error distribution while collecting the point clouds. Secondly, the features in each overlapped area are extracted and a kind of Locally Aggregated Descriptors are built for the matching. Thirdly, a coarse-to-fine pairwise point clouds alignment is applied on the overlapping areas. Finally, the global alignment of point clouds is fulfilled with minimizing the position deviations between the overlapping areas and the adjacent segments. The proposed method has been used to correct the point clouds from several different MLSs. Experiments show that this method automatically locates and corrects the uneven position deviations in terms of good robustness and efficiencies. Besides, it proves that the proposed method is also an easy-to-use tool for the automatic correction of MLS point clouds position and boresights.

1. INTRODUCTION

Mobile laser scanning system can efficiently and densely obtain three-dimensional geometry and texture information of the road environment, which provides a new technical means for high-resolution earth observation. Mobile laser point cloud plays a very important role in national major engineering applications such as 3D mapping, road facilities inventory, high definition map (Kukko, 2013). However, due to the comprehensive influence of GNSS positioning error, IMU attitude determination error, scanner angle and distance measurement error, multi-sensor installation error and so on, the position accuracy of MLS point clouds is difficult to meet the centimeter-level accuracy requirements, showing in Figure 1. Especially in high-rise urban areas, satellite signal occlusion is serious and positioning accuracy is poor, resulting in a decimeter or even meter-level deviation between mobile laser scanning system. The position inconsistency phenomenon occurs among the point clouds collected from the same area, which is difficult to meet the accuracy requirements of the above applications without making a registration pass. (Xu et al, 2015; Yang et al, 2017). More severely, it is quite difficult to automatically locate and correct the uneven position deviations from a long-distance trajectory (e.g., over 20 km). Directly align the MLS point clouds using the registration method such as Iterative Closest Point (ICP) or normal distribution transform (NDT) will not solve the inconsistency and even causing a bad matching result due to the non-rigid deformation inside the point clouds, the mimic geometry and context and the limited overlapped areas. In this paper, a series of approaches including point cloud partition, feature extraction and description, hierarchical pairwise registration and global optimization are designed to address this issue and effectively produce a more consistency point clouds in the same areas.

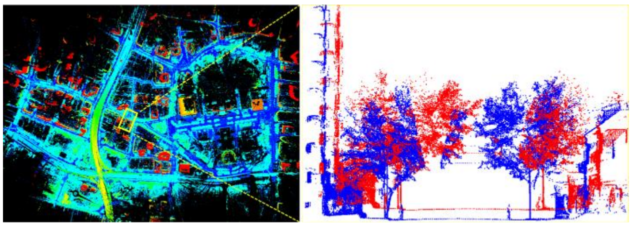


Figure 1. Position inconsistencies in MLS point clouds, the red and blue point clouds are collected by the back and forth scans in different temporal.

Overall, the main contributions of this paper are three-fold:
1) An adaptive partition method for MLS point clouds considering the error distribution characteristics of point clouds is proposed, which ensures the overlapping areas among the overlapping segments and improves the reasonableness of partition for MLS point clouds.
2) A pairwise registration strategy of MLS point clouds from coarse to fine is designed, which effectively solves the large deviation and non-rigid deformation of MLS point clouds.
3) The global alignment optimization of MLS point clouds is applied through optimizing the constructed objective function, which combines the distance of correspondence between overlapping sub-regions with the transformation smoothness between adjacent sub-regions.

Before giving a detailed description of the proposed approach for solving the uneven position deviations from a long-distance

[*] Corresponding author

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLIII-B2-2020-347-2020

trajectory in Section 3, the outline of MLS position inconsistency correction and point cloud alignment is elaborated in Section 2. In Section 4 experimental results for correcting point clouds collected from different MLSs are described. The conclusion and feature work are drawn at the end of this paper.

2. RELATED WORKS

Alignment of point clouds is a long-standing problem in the field of computer vision and photogrammetry. The steps to deal with the inconsistency of point clouds are usually point clouds partition, local pairwise registration and global optimization.

The MLS point clouds partition strategy is summarized into two categories: time-based partition (Han et al., 2014; Yan et at., 2018) and space-based partition (Takai et al., 2013). The former usually builds a time error model to refine the accumulation error from IMU drift and GPS signals loss. One assumption is that there is a consecutive changing relationship between pose error and time stamp, thus the error could be compensated by a pairwise low order Polynomial function. Once the errors in trajectory are corrected, the MLS point clouds could be recalculated with the pose parameter and sensor observation equation. Han et al. (2014), proposed a time-variant model of vehicle trajectory and calculated the coordinate of the point clouds using GCPs. Yan et al. (2018) sliced the MLS point clouds by a fixed period of time to solve the non-rigid registration of multi-strip MLS point clouds. The registration of MLS point clouds could usually obtain high precision but once the vehicle runs slowly or even stops at a traffic light, the MLS point clouds in this area will be very dense, and time-based partition will not work at all. The latter method directly corrects MLS point clouds by slicing the MLS point clouds along the trajectory into segments, assumes that the error in each segment is very small considering the IMU's stability. The length of the segments is usually altered. Treat segment as process unit and then calculate the transformation relationship of overlapping areas' point clouds through geometric and semantic features extracted from point clouds. Takai et al. (2013) partitioned the point cloud into segments according to the key points of the trajectory. After the registration the difference among point clouds are reduced to 50mm.

The point clouds registration has been drawn considerable attention (Pomerleau et al., 2015; Cheng et al., 2018; Dong et al., 2020). Coarse-to-fine is a common strategy employed. The coarse alignment approximately finds an initial transformation contains rotation and translation matrix in a relatively fast and accurate way, thus the fine registration approach such as the NDT and its variants (Magnusson et al,.2007; Das∗ and Waslander†, 2012), and the ICP algorithm (Besl and McKay, 1992) and its variants (Tazir et al., 2018), are implemented to achieve the final transformation. Most of the fine registration algorithm have a bottleneck that the processing time is too long while the point clouds' number is more than 0.1 million, but the dense point clouds are widely seen in MLS. Although there are also many tricks to accelerate the calculation, when the overlapping area is small and the context is similar, the result is poor and unstable. Recently the learning-based point cloud alignment methods developed rapidly, such as deep neural network (Choy et al.,2019; Lu et al., 2019), but the point clouds are more sophisticated and these methods have difficulties in handling this.

Based on the point cloud partition strategy and point clouds registration method, Yan et al., (2018) split point clouds into segments uniformly according to scanning time, and obtained accurate homonymous correspondence through two-step ICP algorithm, finally corrected trajectory error by using the time-variant error model. Yu et al., (2015) extracted the object of different scales in MLS point cloud and found correspondences between the objects by using multi-scale ICP, improving the quality of MLS point cloud data from coarse-to-fine. This method can solve the problem of position consistency correction of MLS point cloud in urban scale, but it depends heavily on the quality of an object and feature extraction of various types of objects and is less feasible in complex urban scenarios. Besides, Shiratori et al. (2015) detected loop closures in MLS point clouds, and used loop constraint to boost the overall quality of point clouds in the overlapping regions. Yang et al. (2018) proposed a method of urban point cloud mapping based on pose map. They combined the classifier and optimization method to eliminate the unstable local correction results, and the conditional random field is used to identify and prune the dynamic targets, preventing the negative impact from dynamic objects.

Global optimization has been a necessary step on point cloud position correction (Theiler et al. 2015, Yang et al. 2018 and Li et al. 2020) Especially in MLS point clouds registration mission, the individual point cloud segment pose could be tackled as nodes in a graph and the constraints between point clouds be formulated as edges. And to find a spatial configuration of the nodes means solving the MLS point cloud registration problem and achieving the global position consistency in overlapped areas. Theiler et al. (2015) formulated the TLS point clouds into a graph structure and conducted a global optimization to evenly spread the residual errors into all TLS point clouds after coarse-to-fine pairwise alignment.

Although the above methods have achieved promising results in specific scenarios, there are still many limitations in robustness and large deviations correction. To address the challenging issues, this paper proposes a novel method to address the position inconsistency of MLS point clouds.

3. METHODOLOGY

The proposed method consists of four key steps: MLS point clouds partition, feature extraction and description, pairwise registration of overlapping point clouds, and global optimization, as illustrated in Figure 2.

3.1 MLS Point Clouds Partition

Inspired by the Takai et al. (2013), This paper slices the MLS point clouds into segments based on the error distribution. Specifically, the position inconsistency deviations in the MLS point clouds often occur to the places where the speed and direction of the moving vehicle change sharply, the position deviation in these places are non-linear distribution. Besides, to improve the adaptability in large deviation, symmetry, and sparse geometric features scenarios, the overlapping areas and crossroads should stay completion for the convenience of the further process. The proposed partition strategy could be summarized as the following four steps: selection of candidate segment points, overlapping points detection, segment points filter and MLS point clouds partition.

3.1.1 Selection of Candidate Segment Points

Based on the trajectory of vehicle, the acceleration and angular velocity of the trajectory will be calculated and the trajectory point whose acceleration or heading angular velocity is larger than the threshold a_{th} or ω_{th} is treated as a velocity change point or angular changing point respectively. Let $q_{a_{start}}$, $q_{a_{end}}$, $q_{o_{start}}$ and $q_{o_{end}}$ be the first, last velocity changing points, angular changing points of a continuous pose trajectory. The point

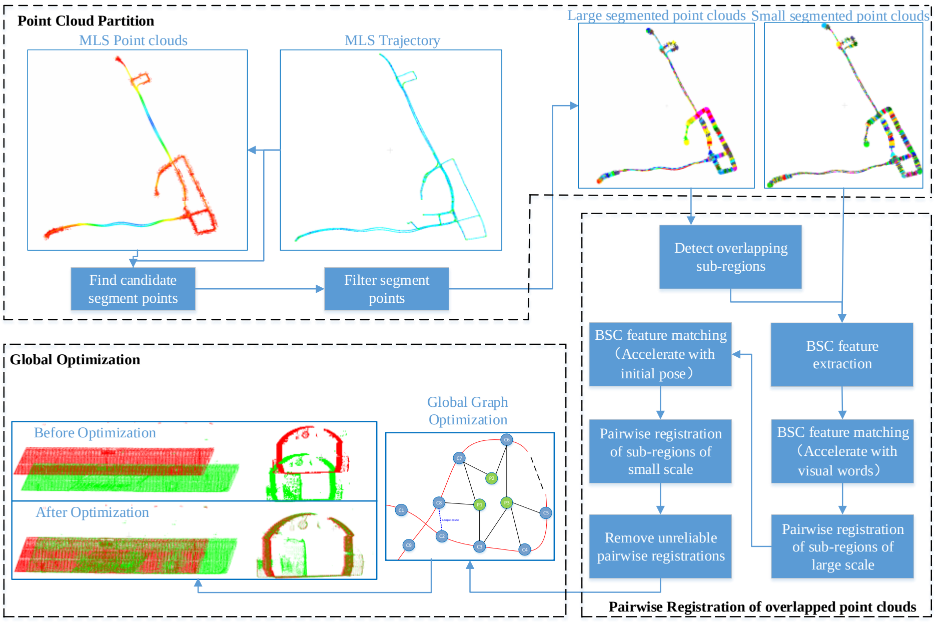

Point Cloud Partition
MLS Point clouds
MLS Trajectory
Large segmented point clouds
Small segmented point clouds
Find candidate segment points
Filter segment points
Detect overlapping sub-regions
BSC feature matching（Accelerate with initial pose）
BSC feature extraction
Pairwise registration of sub-regions of small scale
BSC feature matching（Accelerate with visual words）
Remove unreliable pairwise registrations
Pairwise registration of sub-regions of large scale
Pairwise Registration of overlapped point clouds
Global Optimization
Before Optimization
After Optimization
Global Graph Optimization

Figure 2. The workflow of the proposed method.

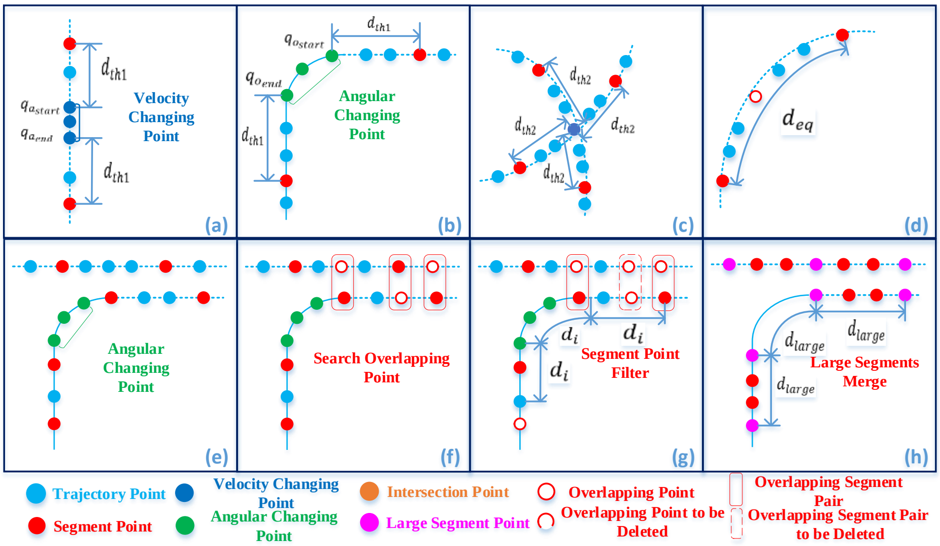


Figure 3. The pipeline of MLS point clouds partition. (a) find segment points by acceleration, (b) find segment points by angular velocity, (c) find segment points by intersections of vehicle trajectory, (d) insert new segment points if adjacent segment points are far away from each other, (e) candidate segment points, (f) find overlapping segment points, (g) filter candidate segment points, (h) obtain large segments by merging small sub-segments.

selected by extending a certain distance d_{th1} outward along the trajectory as the candidate segment point, dotted in red solid points in Figure 3(a), 3(b). Meanwhile, the intersection of trajectories q_{cross} is detected. Crossroad candidate segment point is found by extending a certain distance d_{th2} outward along q_{cross}, shown in Figure 3(c). When the interval distance between adjacent candidate partition points is larger than d_{eq} , the trajectory points at the midpoint are taken as new candidate segment points, dotted in hollow red point in Figure 3(d). Then the candidate segment dotted in red solid points are found and illustrated in Figure 3(e).

3.1.2 Overlapping Points Detection

To correct the inconsistency in the overlapping areas, for each red point as dotted in Figure 3(e), the nearest trajectory point whose accumulated distance along the trajectory is large than d_i but also falls in a radius region are labelled as its overlapping point, as dotted in the hollow red point encircle by a rounded rectangle in Figure 3(f). And the candidate segment point who has the overlapping point will not be candidate segment point any more. The purpose of searching for overlapping segment points is to maximize the overlapping part between the overlapping segments and make subsequent registration more robust.

3.1.3 Segment Points Filter

The distribution of candidate segment points may be uneven, and the segment points near the intersection may be too dense. Hence, the final segment points need to be selected again from the candidate segment points. Firstly, the candidate segment point who has the overlapping point should be deleted; Secondly, candidate segment points in the middle of the intersection points should be deleted and then the trajectory is evenly segmented with a certain interval d_i, as shown in Figure 3(g). And the non-rigid deviation in each segment might be too small to ignore, the rigid transformation between small segments should be handled.

3.1.4 MLS Point Clouds Partition.

According to above three steps, the MLS point cloud could be segmented by the time stamp between trajectory and point clouds. With the small segments obtained, the small segments are then merged into bigger large sub-segments according to the accumulated length d_{large}, as shown in Figure 3(h).

3.2 Feature Extraction And Description

The small segments and the large segments are used as process units. A feature-based approach is applied to correct the position deviation of the overlapping areas in MLS point clouds. Binary Shape Context (BSC, Dong. et al, 2017) is adopted to extract the feature points descriptors in each small segment for matching the correspondence without artificial markers. The covariance matrix is constructed based on a point and its neighboring points in a certain radius, and the eigen vector and eigen value is calculated. Once a point's curvature large enough, it is selected as a key point and then BSC feature descriptor is extracted around it. The BSC features in small segments are concatenated to form the large segments' BSC features. After local feature descriptor is extracted, a common method Vector of Locally Aggregated Descriptor (VLAD, David et al., 2006) was adopted to leverage the function of the feature descriptors. Specifically, BSC features are clustered by K-means to build a visual word dictionary for large segments and small segments.

3.3 Coarse-To-Fine Pairwise Registration

For MLS point cloud data acquired in complex urban scenarios, the general pairwise registration algorithm is of poor robustness and is difficult to apply into complex scenarios with large deviations, symmetrical structures, and sparse geometric features. When the segment length is long, the non-rigid deformation may be large, but the geometric features are rich with a high precision in pairwise registration under the above mentioned hard scenarios. When the segments' length is short, the deformation in the segment may be small, but the geometric features are relatively sparse, and the accuracy of pairwise registration is high in local region, but the robustness is relatively poor. Thus, a hierarchical coarse-to-fine method combining long segment and short segment is proposed to complement each other in pairwise registration. The approach includes three parts: visual words accelerated large segments registration; initial pose information assisted small segments registration and mismatch removal.

3.3.1 Large Segments Registration

When IOU_1 between two large segments is larger than IOU_{large}, the two large segments are considered as revisit. For the sub-segments contained in the large overlapping segments, the IOU_2 between the possible overlapping sub-segments is calculated. When IOU_2 is larger than IOU_{small}, the two small segments are considered as revisit. For a large segment Seg_i, a set $\Omega = \{Seg_j \,|\mathrm{IOU}_{Seg_i Seg_j} > \mathrm{IOU}_{large}, \mathrm{j} < \mathrm{i}\}$, containing its' overlapped segments Seg_j, is built. $\mathrm{IOU}_{Seg_i Seg_j}$ is the overlap value between the Seg_i and Seg_j. And the segments in Ω whose unique id is consecutive are grouped as sub-segments. That is, for instance, Seg_k is an element in Ω, Seg_k and its adjacent large segments consist of $\Omega_A = \{Seg_m \,|Seg_m \in \Omega,\ |\mathrm{k} - \mathrm{m}| < \ \varepsilon\}$, shown in Figure 4. BSC features of all sub-segments contained in Ω_A are grouped to get a feature set F_{Ω_A}, while BSC features of sub-segments in Seg_i forming a feature set F_{Seg_i}. Then the visual word dictionary is used to search the correspondence of Seg_k, for further eliminating completely dissimilar features and improving the efficiency of feature matching. The root mean square error (RMSE) then are spread to the inside small segments.

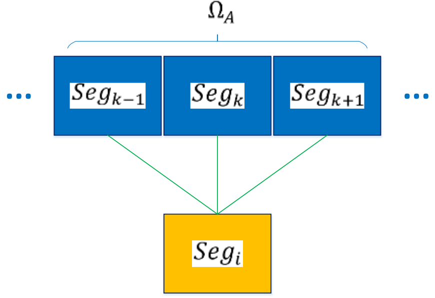


Figure 4. Pairwise registration of large segments. $\mathrm{Seg_i}$ is the target point cloud segment, and $\mathrm{Seg_{k-1}}$, $\mathrm{Seg_k}$, $\mathrm{Seg_{k+1}}$ are its overlapping segments.

3.3.2 Small Segments Registration

According to the pairwise registration result of large segments, the search range could be narrowed during the small segments registration. Searching radius boosts the BSC feature matching considering not only the similarity between features, but also the location constraints of feature points. However, because of the feature representation capabilities, there are mismatch segments in the registration, thus, a mismatch removal is essential.

3.3.3 Mismatch Removal

The mismatch removal includes: the number of the feature matching, self-rotation of each segment and the relative transformation relationship between consecutive adjacent segments. Apparently, the more the number of the homonymous features among the segments, the more reliable the registration results are. The feature matching number which is smaller than a certain threshold is pruned. The POS system carried by the mobile laser scanning system generally has high accuracy, and the attitude deviation of the acquired point clouds in the overlapping area should be small. Therefore, when the relative rotation (roll, pitch, yaw) is greater than Rot_{th}, the pairwise registration result is considered unreliable.

T-test is utilized to eliminate pairwise registration results of abnormal relative transformation in the consecutive adjacent segments. Formulating the moving distance of bounding box center after registration as statistical variables. For segments $\mathrm{Seg_i}$ and $\mathrm{Seg_j}$, the bounding box center's local coordinates are $\mathrm{O_i}$ and $\mathrm{O_j}$, corresponding pose in the global coordinate system separately are $\mathrm{T_i}$ and $\mathrm{T_j}$, and the relative pose obtained by pairwise registration is $T_{Reg_{ij}}$. The distance that $\mathrm{Seg_j}$ moves after registration could be calculated by the equation as: $\mathrm{Dist_{ij}} = |\mathrm{T_j O_j} - \ \mathrm{T_{Reg_{ij}}^{-1} T_i O_j}|$. Mean value, $\overline{\mathrm{D}}$ and the standard deviation, σ, could be calculated from all overlapping segments' moving distance. Once a sampled $Dist$, its corresponding t value calculated using $\mathrm{t} = \frac{\mathrm{Dist} - \overline{\mathrm{D}}}{\frac{\sigma}{\mathrm{n}-1}}$ is greater than the critical value, its corresponding feature matching is discarded.

3.4 Global Optimization

After correcting the local position of the MLS point cloud from coarse-to-fine pairwise registration, the relative transformation relationship between the segments of the overlapping point cloud is obtained, but the transformation relationship between the segments is complex and inconsistent. Therefore, these transformation relationships need to be adjusted globally so that the position deviations are evenly distributed in the point clouds. The small segments and correspondence between segments are described as a graph structure, and are optimized by solving a non-linear least square equation through Levenberg-Marquardt method (Hartley et al., 2003). When there are many overlapping areas, the optimization will loop more iterations and consume too much time. Thus, a simple graph is firstly constructed based on the relative transformation in adjacent segments to get a finer pose, then construct the graph with additional constraint to get the optimal pose transformation.

3.4.1 First Stage Optimization

The overlapping small segments are constrained with small variance before and after optimization, while maintaining the transformation between the adjacent segments. Based on this idea, an object function is formulated which could be divided into data term and smooth term, as equation 1:

$$E=\sum_{i}^{|S|}\sum_{j}^{|S|}\Phi(S_i,S_j)d(T_{S_iS_j}^{-1}T'_{S_iS_j})+\sum_{i}^{|S|}\sum_{j}^{N(S_i)}d(T_{S_iS_{N(S_i)_j}}^{-1}T'_{S_iS_{N(S_i)_j}}) \quad (1)$$

where S and |S| = the segments and the total number of small segments

$\Phi(\mathrm{S_i},\mathrm{S_j})$ = the relationship between segment $\mathrm{S_i}$ and $\mathrm{S_j}$

T_{S_i} = the transformation matrix corresponding to $\mathrm{S_i}$

$T_{S_iS_j}, T_{S_i}T_{S_j}^{-1}$ = the transformation matrix between segment $\mathrm{S_i}$ and $\mathrm{S_j}$

$\mathrm{T'_{S_iS_j}}$ = the transformation matrix between segment $\mathrm{S_i}$ and $\mathrm{S_j}$ after global optimization

$\mathrm{d(T_{S_iS_j}^{-1}T'_{S_iS_j})}$ = the variance of relative transformation matrix between $\mathrm{S_i}$ and $\mathrm{S_j}$ before and after global optimization,

$\mathrm{N(S_i)}$ = set of adjacent segments of $\mathrm{S_i}$.

$\Phi(\mathrm{S_i},\mathrm{S_j})$ is an indicator, that is, equal to 1 when the segment is an overlapping segment, otherwise 0. Data term forces the variance during iteration will not be too large but just a slight adjustment of the pair-wise registration. And the smooth term keeps the transform consistency of the adjacent segments. In case of multi overlapping segments, the pairs that have the most numbers of feature correspondence are remained, ensuring the high precision of correspondence. Here, the weights are set to be equal between data term and smooth term.

3.4.2 Second Stage Optimization

Rely on the fast convergence of the first stage optimization, each overlapping segment has a better initial pose. Then, a more powerful constraint is designed to cover all the segments with adding an inertial term, representing as equation 2. The new object function consisting of data term $\mathrm{E_{data}}$, smooth term $\mathrm{E_{smooth}}$, and inertial term $\mathrm{E_{inertial}}$.

$$\begin{aligned}E= &\sum_{(F_i,F_i')\in C}\mathrm{d}(T_{F_t}(F_t),T_{F_t'}(F_t'))+\\ &\omega_{smooth}\sum_{i}^{|S|}\sum_{j}^{|\varphi(S_i,S_{i+1})|}d(T_{S_i}(\varphi_j),T_{S_{i+1}}(\varphi'_j))+ \qquad (2)\\ &\omega_{inertial}\sum_{i}^{|S|}d(T_{S_i},T'_{S_i})\end{aligned}$$

where C = the set of the homonymous correspondence

$(\mathrm{F_i},\mathrm{F'_i})$ = a pair of homonymous features

$\mathrm{T_{F_i}}$ and $\mathrm{T_{F'_i}}$ = transformation matrices associated with $\mathrm{F_i}$ and $\mathrm{F'_i}$

$\mathrm{T_{F_i}(F_i)}$ and $\mathrm{T_{F'_i}(F'_i)}$ = means transform the $\mathrm{F_i}$ and $\mathrm{F'_i}$ to target coordinate system

$\mathrm{d(T_{F_i}(F_i),\ T_{F'_i}(F'_i)}$ = distance residuals

$\varphi(\mathrm{S_{i,}S_{i+1}})$ and $|\varphi(\mathrm{S_{i,}S_{i+1}})|$ = homonymous points and size between adjacent segment S_i and S_{i+1}

φ_j and φ'_j = jth pair of homonymous points

T_{S_i} = pose of segment S_i

$d(T_{S_i}(\varphi_j),\ T_{S_{i+1}}(\varphi'_j)$ = Euclidean distance residuals of homonymous points in adjacent segments S_i and S_{i+1}

$d(T_{S_i},T'_{S_i})$ = variance of transformation of segment S_i

Note that, ω_{smooth} and $\omega_{\mathrm{inertial}}$ are the weights of the smooth term and inertial term, $\omega_{\mathrm{smooth}} = \mathrm{E_{data}}/\ \mathrm{E_{smooth}}$, $\omega_{\mathrm{inertial}}$ is very small and is generally set to 10^{-3}. The data term is the sum of Euclidean distance residuals of the homonymous points inside the overlapping segment. The data term minimizes the gap in geometry space between point clouds scanned in the mostly same area. Due to the mismatch in hard scenarios, the smooth term is used to punish the abnormal transformation at a local scope of adjacent segments, controlling the distance residuals of the homonymous point and the third inertial term forces the variance of each segment limited in a small scope during the optimization process. Both smooth and inertial term compact the segments and relief the gap between segments. The graph is constructed incrementally along the vehicle trajectory. If there are overlapping between the current and previous segments, both data term and smooth term are updated. If there are no overlapping detected, only the smooth term will be updated.

4. EXPERIMENTS

4.1 Dataset

The MLS point clouds partition performance is evaluated on point clouds acquired from a RIEGL scanner in Shanghai including 1 billion points. The total trajectory is about 20km, shown in Figure 6. And small point clouds, as listed in Table 1 and shown in Figure 5, are used to demonstrate the result of the position inconsistency correction.

	Trajectory Length (km)	The number of points (billion)	Description of scene
Data 1	1.1	1.2	Symmetrical structure
Data 2	0.85	1.57	Large inconsistency & hard scenario
Data 3	0.65	1.57	Large inconsistency & intersections

Table 1. MLS point cloud information.

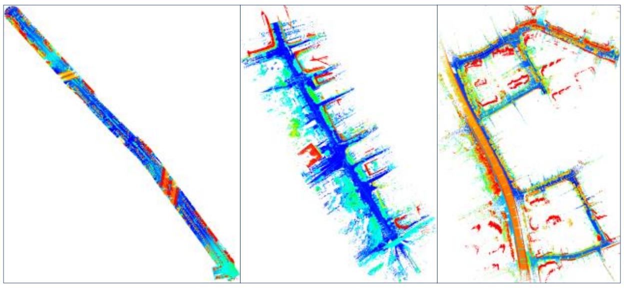
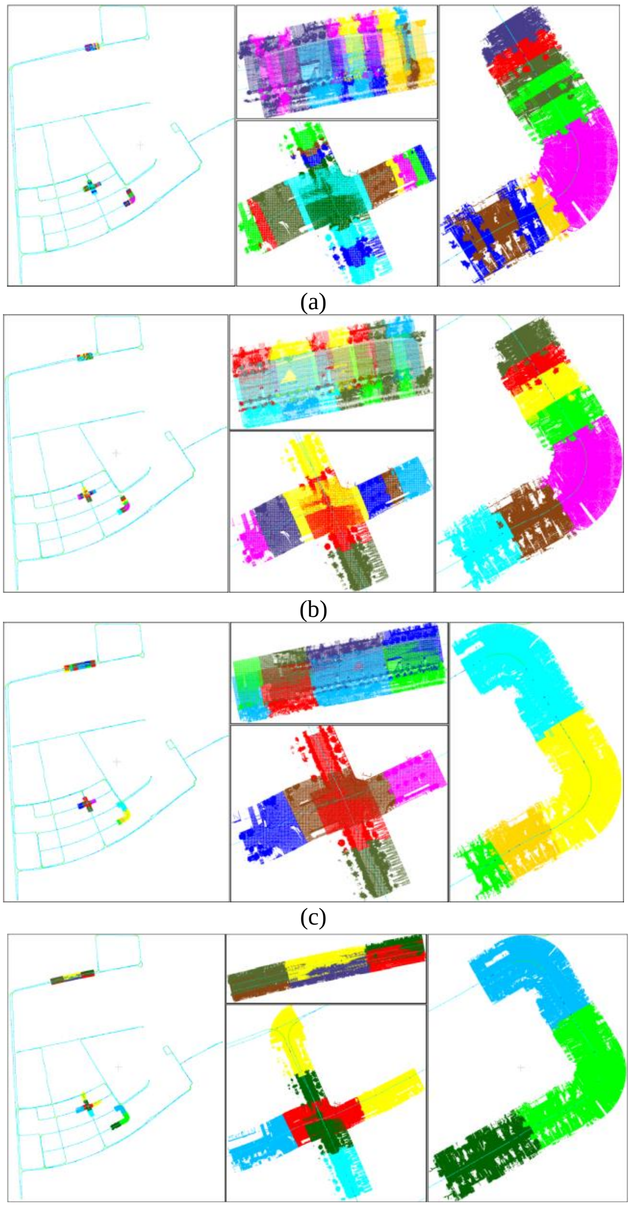


Figure 5. The overview of point clouds Data 1-3.
(a) Data 1, (b) Data 2, (c) Data 3.

4.2 Result and Evaluation

4.2.1 Point Clouds Segments

We set the segment length separately from 5m to 40m and the result is displayed in Figure 6. It can be seen that no matter how long the sub-segment is, the MLS point clouds are basically in the same position, there is adequate overlap between the overlapping segments. The MLS point clouds are completely preserved at intersections and bends. Other parameter we set is similar to Takai et al. (2013), but adjusted according to our real data. And we find there are slight effect on choosing parameters except the d_{eq}, which dominates the length of the segments, the d_c, which enlarges the possibility of detecting overlapped and the d_i which keeps the even distribution of segment points.



(a)

(b)

(c)

(d)

Figure 6. Result of point cloud partition, segment length is (a) 5m, (b) 10m, (c) 20m, (d) 40m. The left column is top view of segments; the upper center shows the segments at round-trip; the down center shows the segments at crossroads, and the right column shows the segments at bend.

4.2.2 Position Deviation Correction

Given no ground truth provided. the manually labelled homonymous points in the overlapping point cloud are used as the ground truth to evaluate the relative position accuracy. RMSE of homonymous points between overlapping segments is used to evaluate the relative position accuracy of overlapping segments, and RMSE of virtual homonymous points between adjacent segments is used to evaluate the relative position accuracy. Shown in Table 2.

From considerable experiments, we find that 20m is almost the optimal length for the MLS point clouds in Shanghai. When the segment length is small, such as 5m, or even smaller, mismatch results in registration stage are too many to obtain a good optimization result, although the smoothing terms of adjacent segments are the largest. These mismatch makes the optimization convergence very slow, even failure, as shown in Table 2. When the segment length is 40m, or even larger, the non-rigid deformation inside the segment is relatively large, and there is still a relatively large position inconsistency after position consistency correction. But larger segments we merged from small segments stabilize the registration result, preventing many gross errors and the final result is also acceptable. When the segment length is between 10 and 20m, the registration results of pairwise registration are still reliable, with negligible non-rigid deformation inside the segments. Meanwhile, the global optimization succeeds as expecting.

Figure 7 shows the location changings during the process, and the adjacent segments' consistency is qualitatively demonstrated. The position inconsistency in the same areas after processing by our algorithm could be reduced to cm level depending on the origin point cloud, for example, the beam parameter calibration value. If enough GCPs are available, the MLS point clouds could be easily transformed into absolute coordinate. Figure 8 shows more position correction result.

5. CONCLUSION

A marker-free MLS point clouds position inconsistency correction method is proposed in this paper. The point clouds collected from different sensors were used to verify the effectiveness and robustness of the proposed method. The adaptive hierarchical partition considering the error distribution characteristics of point clouds guarantees the overlapping segments and improves the reasonableness of MLS point clouds partition. Visual word and prior pose information acceleration matching strategy ensures the efficiencies of the matching, and a coarse-to-fine hierarchical registration method guarantees the robustness of point clouds alignment, leading to effective correction of large deviation in the overlapping MLS point clouds. In the future, learning-based methods will be evaluated on the large urban scene point cloud alignment and pruning the moving object to relieve the mismatch.

ACKNOWLEDGEMENTS

This research is jointly supported by the National Natural Science Foundation of China Projects (No. 41531177, No. 41725005)

REFERENCES

Besl, P.J., McKay, N.D., 1992. A method for registration of 3-D shapes. IEEE Trans. Pattern Anal. Mach. Intell. 14, 239–256. doi.org/ 10.1109/34.121791.

Choy, C., Park, J., Koltun, V., 2019. Fully Convolutional Geometric Features, 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 8957-8965. doi.org/ 10.1109/ICCV.2019.00905.

Das∗, A., Waslander†, S.L., 2012. Scan Registration with Multi-Scale K-Means Normal Distributions Transform, IEEE/RSJ International Conference on Intelligent Robots & Systems. doi.org/10.1109/IROS.2012.6386185.

David Nistér, & Henrik Stewénius. 2006. Scalable Recognition with a Vocabulary Tree. Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. IEEE. doi.org/10.1109/CVPR.2006.264.

Dong, Z., Liang, F., Yang, B., Xu, Y., Zang, Y., Li, J., Wang, Y., Dai, W., Fan, H., Hyyppäb, J., Stilla, U., 2020. Registration of large-scale terrestrial laser scanner point clouds: A review and benchmark. ISPRS Journal of Photogrammetry and Remote Sensing 163, 327-342. doi.org/10.1016/j.isprsjprs.2020.03.013.

Dong, Z., Yang, B., Liu, Y., Liang, F., & Zang, Y. 2017. A novel binary shape context for 3d local surface description. ISPRS Journal of Photogrammetry and Remote Sensing, 130, 431-452. doi.org/10.1016/j.isprsjprs.2017.06.012.

Gojcic, Z., Zhou, C., Wegner, J.D., Wieser, A., 2019. The perfect match: 3d point cloud matching with smoothed densities, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 5545-5554. doi.org/ 10.1109/CVPR.2019.00569.

Han, J. Y., Chen, C. S., & Lo, C. T. 2014. Time-variant registration of point clouds acquired by a mobile mapping system. IEEE Geoscience and Remote Sensing Letters, 11(1), 196-199. doi.org/10.1109/LGRS.2013.2252417.

Kukko, A. 2013. Mobile laser scanning - system development, performance and applications. Ph.D. Dissertation. Finnish Geodetic Institute.

Li, W., Wang, C., Lin, C., Xiao, G., Wen, C., Li, J., 2020. Inlier extraction for point cloud registration via supervoxel guidance and game theory optimization. ISPRS Journal of Photogrammetry and Remote Sensing 163, 284-299. doi.org/10.1016/j.isprsjprs.2020.01.021.

Liang, C., Song, C., Xiaoqiang, L., Hao, X., Yang, W., & Manchun, L., et al. 2018. Registration of laser scanning point clouds: a review. Sensors, 18(5), 1641-. doi.org/ 10.3390/s18051641.

Lu, W., Wan, G., Zhou, Y., Fu, X., Yuan, P., & Song, S. (2019). DeepVCP: An End-to-End Deep Neural Network for Point Cloud Registration. 2019 IEEE/CVF International Conference on Computer Vision (ICCV), 12-21. doi.org/ 10.1109/ICCV.2019.00010.

Magnusson, M., Lilienthal, A., & Duckett, T., 2007. Scan registration for autonomous mining vehicles using 3d-ndt. Journal of Field Robotics, 24(10), 803-827. doi.org/10.1002/rob.20204.

Pomerleau, F., Colas, F., & Siegwart, R., 2015. A review of point cloud registration algorithms for mobile robotics.Foundations & Trends in Robotics, 4(1), 1-104. doi.org/ 10.1561/2300000035.

Shiratori, T., Berclaz, J., Harville, M., Shah, C., Li, T., & Matsushita, Y., et al. 2015. Efficient Large-Scale Point Cloud Registration Using Loop Closures. 2015 International Conference on 3D Vision (3DV). IEEE Computer Society. doi.org/10.1109/3DV.2015.33.

Takai, S., Date, H., Kanai, S., Niina, Y., Oda, K., & Ikeda, T. 2013. Accurate registration of MMS point clouds of urban areas using trajectory. ISPRS Ann. Photogramm. Remote Sens. Spat. Inf. Sci, 277-282. doi.org/10.5194/isprsannals-II-5-W2-277-2013.

Tazir, M.L., Gokhool, T., Checchin, P., Malaterre, L., Trassoudaine, L., 2018. Cluster ICP: Towards Sparse to Dense Registration, Proceedings of the 15th International Conference on Intelligent Autonomous Systems.

Theiler, P.W., Wegner, J., Schindler, K., 2015. Globally consistent registration of terrestrial laser scans via graph optimization. ISPRS J. Photogram. Remote Sens. 126 – 138. doi.org/10.1016/j.isprsjprs.2015.08.007.

Xu S., Cheng P., Zhang Y., et al., 2015. Error Analysis and Accuracy Assessment of Mobile Laser Scanning System [J]. Open Automation & Control Systems Journal, 7(1):485-495. doi.org/10.2174/1874444301507010485.

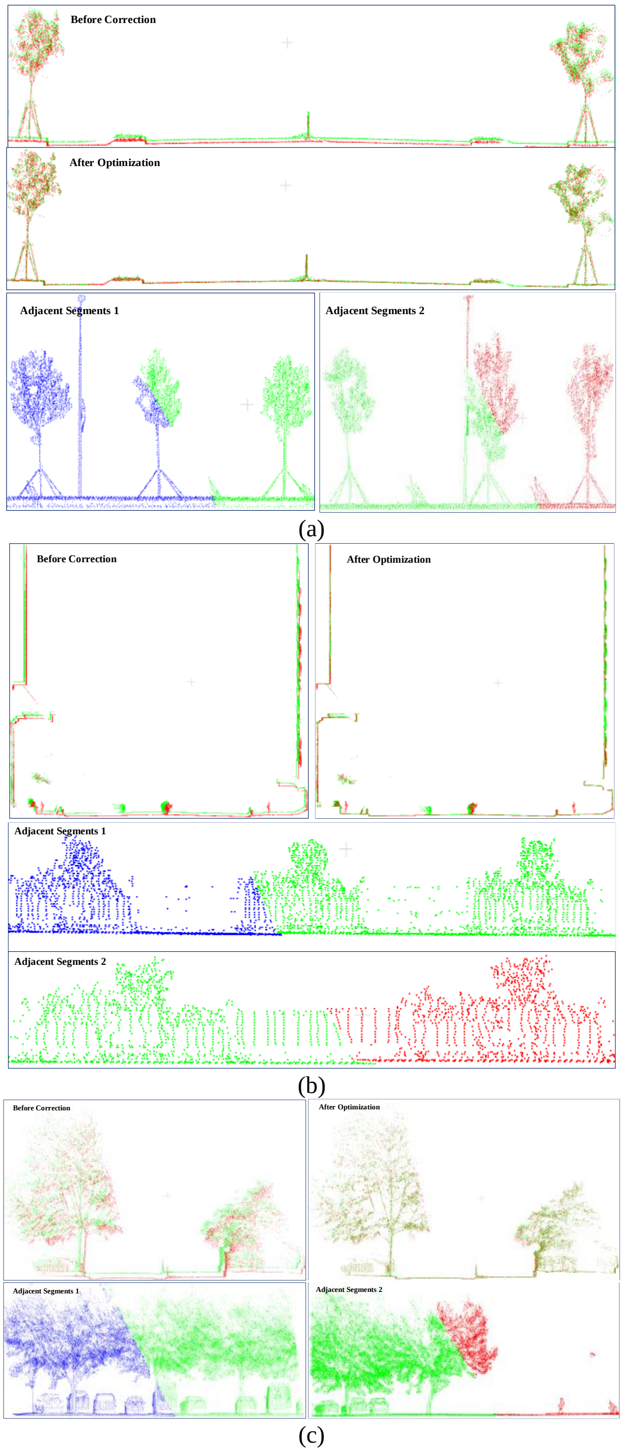
Yan, L., Tan, J., Liu, H., Xie, H., & Chen, C. 2018. Automatic non-rigid registration of multi-strip point clouds from mobile laser scanning systems. International Journal of Remote Sensing, 39(6), 1713-1728. doi.org/ 10.1080/01431161.2017.1410248.

Yang, S., Zhu, X., Nian, X., Feng, L., Qu, X., & Ma, T. 2018. A robust pose graph approach for city scale LiDAR mapping. 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 1175-1182. doi.org/10.1109/IROS.2018.8593754.

Yang, B., Dong, Z., Liu, Y., Liang, F., & Wang, Y. 2017. Computing multiple aggregation levels and contextual features for road facilities recognition using mobile laser scanning data. ISPRS Journal of Photogrammetry and Remote Sensing, 126, 180-194. doi.org/10.1016/j.isprsjprs.2017.02.014.

	Segment Length (m)	Origin Point Clouds		After Pairwise Registration		After Global Optimization		Optimization Time Consuming (s)
		data term (m)	smooth term (m)	data term (m)	smooth term (m)	data term (m)	smooth term (m)	
Data 1	5	0.188	0	0.068	0.184	0.329	0.544	11.534
	10	0.191	0	0.052	0.184	0.007	0.104	0.664
	20	0.199	0	0.039	0.193	0.064	0.022	0.156
	40	0.182	0	0.066	0.177	0.067	0.033	0.066
Data 2	5	0.396	0	0.091	0.387	0.99	0.166	6.321
	10	0.422	0	0.083	0.399	0.093	0.113	0.268
	20	0.459	0	0.092	0.386	0.093	0.06	0.168
	40	0.403	0	0.09	0.378	0.103	0.029	0.105
Data 3	5	0.737	0	0.093	0.736	0.154	3.135	4.407
	10	0.729	0	0.91	0.736	0.124	0.089	0.743
	20	0.765	0	0.084	0.752	0.097	0.061	0.129
	40	0.771	0	0.103	0.759	0.111	0.065	0.086

Table 2. Position inconsistency correction result of MLS point clouds

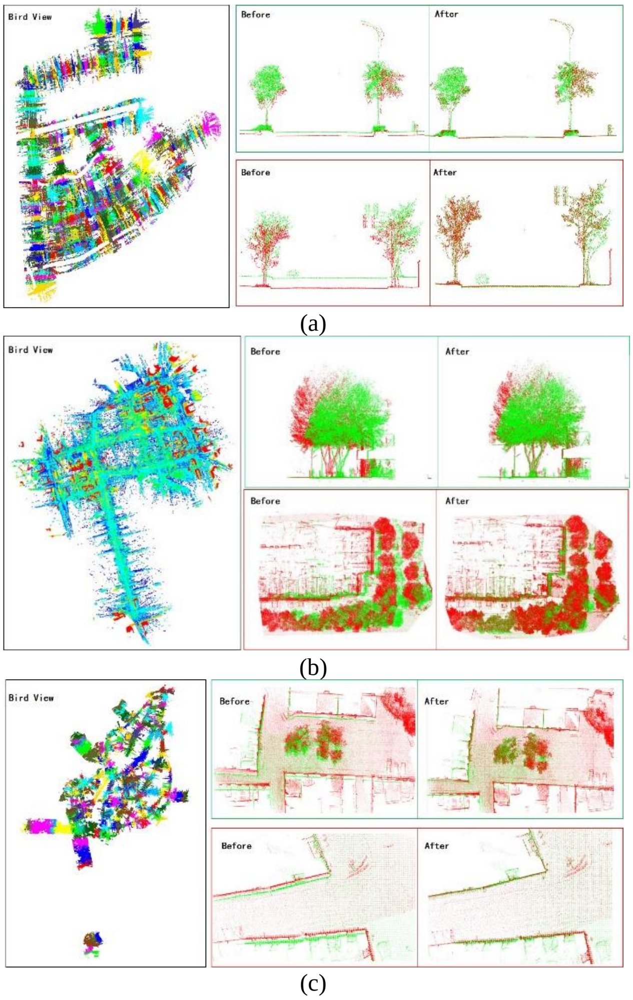


(a)

(b)

(c)

Figure 7. Data 1-3 (segment length is 20m) after position consistency correction. (a) Data 1, (b) Data 2, (c) Data 3. In each sub-figure shows the original point clouds before position inconsistency correction, sectional view after optimization, and sectional view of adjacent segments along the trajectory respectively.



(a)

(b)

(c)

Figure 8. Position correction result of MLS point clouds in: (a) Shanghai, China (b) Wuhan, China, (b) Tallinn, Estonia

This contribution has been peer-reviewed.
https://doi.org/10.5194/isprs-archives-XLIII-B2-2020-347-2020 | © Authors 2020. CC BY 4.0 License.