# Correction Algorithm of LIDAR Data for Mobile Robots

Wenzhi Bai, Gen Li(✉), and Liya Han

Huazhong University of Science and Technology,
Wuhan 430074, People's Republic of China
{wen, ligen_hust, hly1993}@hust.edu.cn

**Abstract.** Laser range finder (LRF) or laser distance sensor (LDS), further referred to as LIDAR (light detection and ranging). LIDAR can obtain environmental point cloud data, while a robot can realize environmental sensing by adoption of the point cloud data generated and LIDAR-based SLAM (Simultaneous Localization And Mapping) algorithm. The precision of point clouds provided by the LIDAR determines that of environmental sensing of the LIDAR-based mobile robot. In this paper, a common correction algorithm has been proposed to correct the inaccuracy of measured point cloud data caused by mobile LIDAR, effectively improving the precision of point cloud data measured by the LIDAR under a mobile state. It also conducts mathematical derivation of the algorithm, presents simulation and real world experiments performed and verifies the necessity and effectiveness of the algorithm derived by experimental results in the paper.

**Keywords:** LIDAR · Data correction · Mobile robot · SLAM

## 1 Introduction

LIDAR were mostly applied in research-based robots in the past, featuring high scanning frequency and dense sampling points [1, 2]. At the same time, researchers ignored the missing data precision caused by LIDAR movement in a scan period due to the robot using LIDAR with slow moving speed. At present, LIDAR is developed towards low cost with the increasing demands of consumer robots [3]. Although low-cost LIDAR has low scanning frequency and sparse sampling point, demanding movement speed of the mobile robot has become requested. In this way, the precision of the measured point cloud of LIDAR in the moving state has begun to become crucial. It is worth noting that LIDAR supplier lack the power to achieve due to lack of trace data of the moving robots and the application party lack feature knowledge of LIDAR to solve the problems, since LIDAR data correction in the moving state situates in the intermediate zone between LIDAR supplier and the application party. Thereby, a data correction algorithm has been put forward in this paper in order to realize a more precise measurement for LIDAR.

The remainder of this paper is organized as follows. The second section and third section of the paper has introduced related research works and problem description as well as theoretical derivation of LIDAR data correction algorithm in moving state; In

Y. Huang et al. (Eds.): ICIRA 2017, Part II, LNAI 10463, pp. 101–110, 2017.
DOI: 10.1007/978-3-319-65292-4_10

the fourth section, it has analyzed experiments and results as well as the contribution of the algorithm on the environment sensing of the robot; Conclusion has been conducted in the fifth section.

## 2 Related Work

In the early LIDAR application, laser scanning equipment with high scanning frequency was applied in detecting land-forms under the condition of airborne. With scanning frequency of LIDAR is high, flight speed in airborne is still fast. Thus, it's necessary to resolve the space attitude located by each measuring transient within the LIDAR scanning period, so as to obtain measured data of practical significance [4, 5]. Unlike other distance sensors such as sonars or IR sensors, an LiDAR is capable of fine angular and distance resolution, realtime behavior, and low false positive and negative rates. Similar to the application of airborne laser scanning equipment, LIDAR has been extensively used in the existing highly-efficient SLAM algorithm [6–8], serving numerous mobile robots. With the emergence of consumer robots, LIDAR has been developed in a low-cost plan with advantages of smaller size and lower consumption, scanning frequency, range and sampling points [3]. Although movement speed of the moving robot gets faster with decreasing scanning frequency of LIDAR, calculation is still conducted in scan [9] in the algorithms of mapping, localization and obstacle avoidance, which not only ignores the pose changes generated by movement in the measuring process of point cloud data in the scan, but also fails in presenting correction methods and realization means for data in the scan.

## 3 Correction Algorithm

### 3.1 Problem Describer

A common LIDAR application is presented in Fig. 1. The robot can sense the environment by the LIDAR carried. In general, LIDAR consists of fixed part and rotational part; the former is for fixing on the robot or other moving platform, while the latter is for an optical path realizing 360° environment scanning with utilization of rotated measuring units, so as to obtain environmental point cloud data of the whole plane. LIDAR is an important sensor for the application to sense the environment. The point cloud data obtained can construct a grid map through the SLAM algorithm. Further, the grid map is the basis of realizing navigation and obstacles avoidance of the robot. Also, cloud point data is important for matching with the existing map to realize the functions like localization. In this way, the precision of point cloud data will directly affect the realization of its dependent functions.

LIDAR is operated by scanning the whole plane environment with the optical path via rotating the rotational part to measure distances of the measured objects at a certain interval. Thus, measured point cloud data of the whole circle can be obtained. The whole data with time stamp can be offered to the robot for calculation. Two extreme cases greatly affecting the precision of LIDAR are proposed here. One is that we can

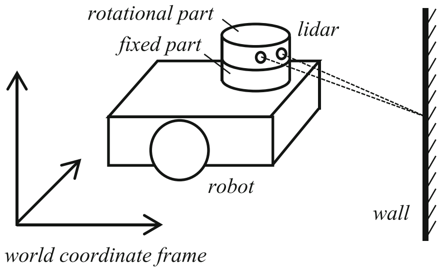


**Fig. 1.** Application of LIDAR

predict the point cloud data on its measured flat of the LIDAR are just multiple measurements of the same point when angular velocity of rotation of the rotational part of the LIDAR is consistent with that of robot in a reverse direction, as shown in case 1 of Fig. 2. Two is when the robot moves and scans a wall that is vertical to its moving direction, the robot moves rapidly from a distance far from the wall to near the wall. We can predict that the point clouds generated by the wall that is vertical to the moving direction will not vertical to the moving direction any more upon observing the point cloud data in a scanning period.

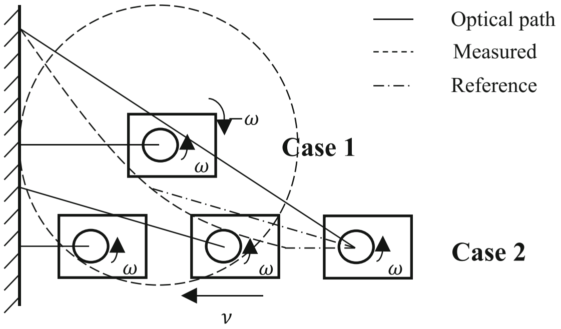


**Fig. 2.** Schematic diagram of extreme cases

We can obtain from the two cases that plane point cloud data given are close to the measured object with only LIDAR measuring errors, under the situation of rotation movement of the rotational part of LIDAR and no other relative movement, when the fixed part of LIDAR is static with respect to the measured object. However, when LIDAR is a moving state, the scanning movement within a scanning period contains not only the rotation movement of the rotational part of LIDAR, but also translation and rotation movements generated by the robot with fixed LIDAR with respect to the measured object. What we do in the paper is to consider these movements in moving state of LIDAR while correcting point cloud data of LIDAR in the scanning period, so that the point cloud data in the scanning plane can accurately present the measured object and improve the precision of LIDAR point cloud data.

### 3.2 Mathematical Derivation

Cartesian coordinates indicating the position and orientation of an object in the environment is a basic method used in the research field of robot. Method of the coordinates transformation in the moving process of the robot has been presented in [10]. The robot and all parts of LIDAR are assumed to be rigid bodies. Plane will be considered only. Plane data are parallel to the floor collected for both single-line LIDAR and multi-line LIDAR, ignoring Z axis data. Therefore, all data are projected to the OXY plane along Z axis for the calculation, when the 3D coordinate are simplified to a 2D one.

The world coordinate is defined as $\{W\}$ coordinate; The robot coordinate is defined as $\{R\}$ coordinate; The coordinate of the robot fixed with LIDAR is defined as $\{L\}$ coordinate; The robot coordinate is defined as $\{R'\}$ coordinate with Δt (Δt $<T$) time; The LIDAR coordinate is defined as $\{L'\}$ coordinate with Δt (Δt $<T$) time; $P_1$ and $P_2$ are the observing points in the world coordinate, of which, $P_1$ is the targeted point observed in the starting time, $P_2$ is the targeted point observed with Δt (Δt $<T$) time. Relative positions among each coordinate are presented in Fig. 3.

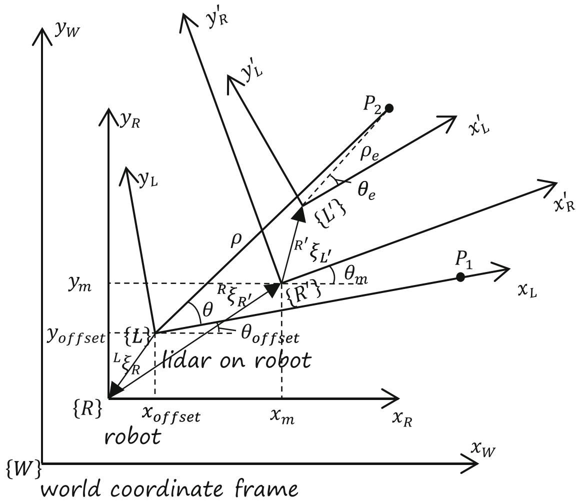


**Fig. 3.** Relative positions of coordinates

It can be seen from Fig. 3, $\{L\}$ coordinate is fixed on the $\{R\}$ coordinate due to the robot equipped with LIDAR, whose relative position is ${}^{L}\xi_R = \ominus {}^{R}\xi_L = \ominus\xi\left(x_{offset}, y_{offset}, \theta_{offset}\right)$; $\{R'\}$ coordinate system is obtained by $\{R\}$ coordinate upon moving with Δt (Δt $<T$), whose relative position is regarded as ${}^{R}\xi_{R'} = \xi(x_m, y_m, \theta_m)$; $\{L'\}$ coordinate is obtained by $\{L\}$ coordinate upon moving with Δt (Δt $<T$); Thus, the relative position of $\{L'\}$ coordinate relating to $\{R'\}$ coordinate is ${}^{R'}\xi_{L'} = \xi\left(x_{offset}, y_{offset}, \theta_{offset}\right)$ since LIDAR is fixed on the robot.

The conversion relation of coordinate:

$$^{L}\boldsymbol{P}_2 = \left({}^{L}\xi_R \oplus {}^{R}\xi_{R'} \oplus {}^{R'}\xi_{L'}\right) \cdot {}^{L'}\boldsymbol{P}_2 \tag{1}$$

Where, each symbol stands for:

- $^{A}\boldsymbol{P}$ — the point P is described by coordinate vectors relative to frame $\{A\}$
- $\xi$ — abstract representation of 3-dimensional Cartesian pose
- $^{A}\xi_B$ — abstract representation of 3-dimensional relative pose, frame $\{B\}$ with respect to frame $\{A\}$
- $\oplus$ — pose composition operator
- $\ominus$ — inverse of a pose (unary operator)
- $\cdot$ — transformation of a point by a relative pose, e.g. $\xi \cdot \boldsymbol{p}$

From the above relations, it can obtain:

$$^{L}\xi_R = \ominus {}^{R}\xi_L \sim \begin{pmatrix} \cos\theta_{offset} & -sin\theta_{offset} & x_{offset} \\ sin\theta_{offset} & cos\theta_{offset} & y_{offset} \\ 0 & 0 & 1 \end{pmatrix}^{-1} = {}^{R}\boldsymbol{T}_L^{-1} = {}^{L}\boldsymbol{T}_R \tag{2}$$

$$^{R}\xi_{R'} \sim \begin{pmatrix} \cos\theta_m & -sin\theta_m & x_m \\ sin\theta_m & cos\theta_m & y_m \\ 0 & 0 & 1 \end{pmatrix} = {}^{R}\boldsymbol{T}_{R'} \tag{3}$$

$$^{R'}\xi_{L'} \sim \begin{pmatrix} cos\theta_{offset} & -sin\theta_{offset} & x_{offset} \\ sin\theta_{offset} & cos\theta_{offset} & y_{offset} \\ 0 & 0 & 1 \end{pmatrix} = {}^{R'}\boldsymbol{T}_{L'} \tag{4}$$

In the formula, we use the symbol $^{A}\boldsymbol{T}_B$ to denote that homogeneous transform representing frame $\{B\}$ with respect to frame $\{A\}$. Note that $^{A}\boldsymbol{T}_B = \left({}^{B}\boldsymbol{T}_A\right)^{-1}$. The symbol $\sim$ is denoted that the two representations are equivalent.

$^{L}\boldsymbol{P}_2$ and $^{L'}\boldsymbol{P}_2$ are presented as $^{L}\widetilde{\boldsymbol{P}}_2 = \left({}^{L}x, {}^{L}y, 1\right)^T$ and $^{L'}\widetilde{\boldsymbol{P}}_2 = \left({}^{L'}x, {}^{L'}y, 1\right)^T$ in homogeneous form. That is,

$$^{L}\widetilde{\boldsymbol{P}}_2 = {}^{L}\boldsymbol{T_R} \cdot {}^{R}\boldsymbol{T}_{R'} \cdot {}^{R'}\boldsymbol{T}_{L'} \cdot {}^{L'}\widetilde{\boldsymbol{P}}_2 \tag{5}$$

$(\rho_e, \theta_e)$ is the $^{L'}\boldsymbol{P}_2$ observing data obtained by LIDAR, thus,

$$\begin{cases} {}^{L'}x = \rho_e \cdot \cos\theta_e \\ {}^{L'}y = \rho_e \cdot \sin\theta_e \end{cases} \tag{6}$$

Measured correction value of point $^{L}\boldsymbol{P}_2$ is $(\rho, \theta)$ with $\Delta$t ($\Delta$t $< T$) time based on timestamp:

$$\begin{cases} \rho = \left| \sqrt{{}^{L}x^{2} + {}^{L}y^{2}} \right| \\ \theta = \begin{cases} \arctan \frac{{}^{L}y}{{}^{L}x} & ({}^{L}x > 0,\ {}^{L}y \in \mathrm{R}) \\ \arctan \frac{{}^{L}y}{{}^{L}x} + \pi & ({}^{L}x < 0,\ {}^{L}y \geq 0) \\ \arctan \frac{{}^{L}y}{{}^{L}x} - \pi & ({}^{L}x < 0,\ {}^{L}y < 0) \\ \frac{\pi}{2} & ({}^{L}x = 0,\ {}^{L}y < 0) \\ -\frac{\pi}{2} & ({}^{L}x = 0,\ {}^{L}y < 0) \end{cases} \end{cases} \tag{7}$$

A special situation should be noted, when it's ${}^{L}x = 0, {}^{L}y = 0$, $\theta$ has infinitely many solutions with $\rho = 0$. It has a practical significance in physics, namely, the measured value after Δt time is just the position of LIDAR at the moment represented by the timestamp. However, the measured value $\rho = 0$ is less than the minimum range of LIDAR, which will be ignored in the later use.

All data measured in a frame of LIDAR are converted into the coordinate of moment represented by the timestamp to complete the correction of data in that frame.

## 4 Experiments

In this section, it aimed to validate the necessity and effectiveness of the algorithm proposed in this paper through conducting simulation tests and real world experiment.

### 4.1 Simulation Tests

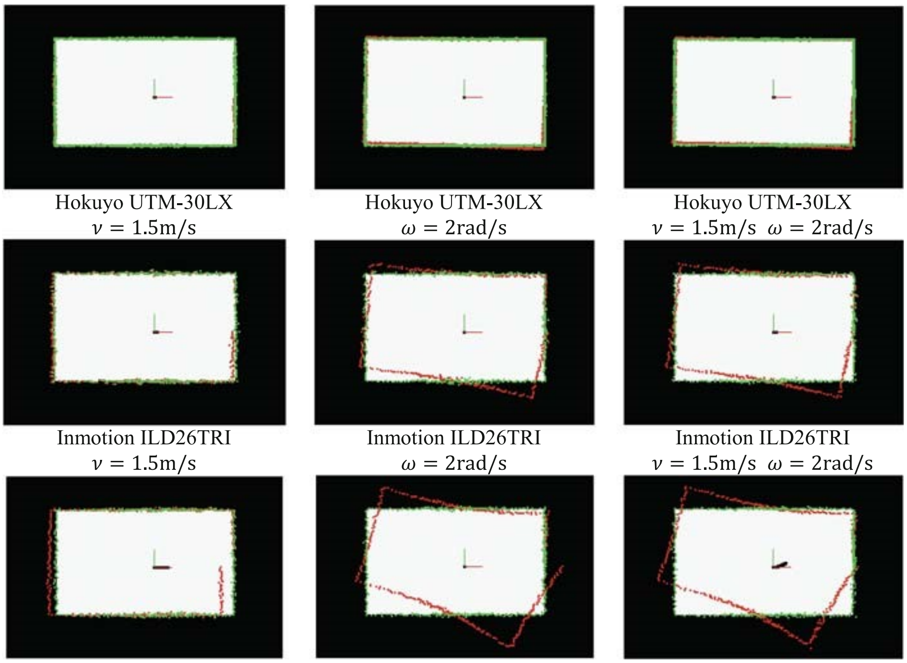
The LIDAR simulation parameters used in the simulation is conducted based on performance parameters of Hokuyo UTM-30LX and Inmotion ILD26TRI described in literature [1, 11] (Table 1):

**Table 1.** List of LIDAR simulation parameters

| Type | Sample (point/scan) | Range (°) | Frequency (Hz) | Std dev (mm) |
|---|---|---|---|---|
| Hokuyo UTM-30LX | 1440 | 360 | 40 | 30 |
| Inmotion ILD26TRI | 360 | 360 | 8 | 60 |

Linear velocity of the moving robot is set as $v = 1.5$ m/s and the angular velocity as $\omega = 2$ rad/s in normal speed, while linear velocity is set as $v = 6$ m/s and the angular velocity as $\omega = 5$ rad/s in high speed. Deviation generating between LIDAR installation center and rotational center of the robot has been considered in the translation speed occurring in the high-speed rotation during simulation.

The simulation is divided into three groups, including: measuring in a moving robot with installation of UTM-30LX LIDAR in normal speed, measuring in a moving robot with installation of ILD26TRI in normal speed and measuring in a moving robot with installation of UTM-30LX LIDAR in high speed; And moving status of each group is



Hokuyo UTM-30LX $v = 1.5\text{m/s}$

Hokuyo UTM-30LX $\omega = 2\text{rad/s}$

Hokuyo UTM-30LX $v = 1.5\text{m/s}$ $\omega = 2\text{rad/s}$

Inmotion ILD26TRI $v = 1.5\text{m/s}$

Inmotion ILD26TRI $\omega = 2\text{rad/s}$

Inmotion ILD26TRI $v = 1.5\text{m/s}$ $\omega = 2\text{rad/s}$

Inmotion ILD26TRI $v = 6\text{m/s}$

Inmotion ILD26TRI $\omega = 5\text{rad/s}$

Inmotion ILD26TRI $v = 6\text{m/s}$ $\omega = 5\text{rad/s}$

**Fig. 4.** Simulation result (Color figure online)

composed of translation, rotation and translation as well as rotation. Coordinate presented in Fig. 4 is the LIDAR coordinate at the moment of starting scanning. We can see from the simulation results, the black trace simulates the moving trace of a moving robot in the scanning process of a frame; the black and white areas simulate areas with obstacles and without obstacle in the environment, respectively. The red point cloud simulates a frame of point cloud data collected by LIDAR; while the green one is the point cloud data of a scan upon corrected by the algorithm proposed in this paper. Starting from the positive direction of X-axis, the scanning process is completed upon scanning a circle along the counterclockwise.

### 4.2 Real World Experiments

The experiment setup is composed of a mechanical arm carrying LIDAR, as shown in Fig. 5. Highly-precise movement can be realized and accurate movement locus can be obtained with the mechanical arm that can well restore the real scene of moving process of the moving robot.

The experiment is divided into two groups that robots are carried Hokuyo UTM-30LX and Inmotion ILD26TRI, respectively; each group consists of scanning plane point cloud in a static state, in the translation of normal speed and in the translation and rotation of normal speed. LIDAR parameters are presented in Table 2.

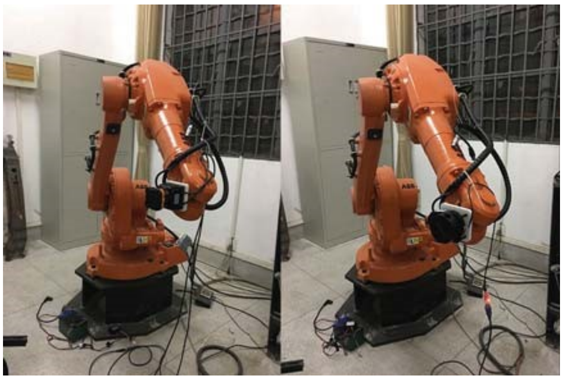


**Fig. 5.** Experiment setup

**Table 2.** List of LIDAR parameters

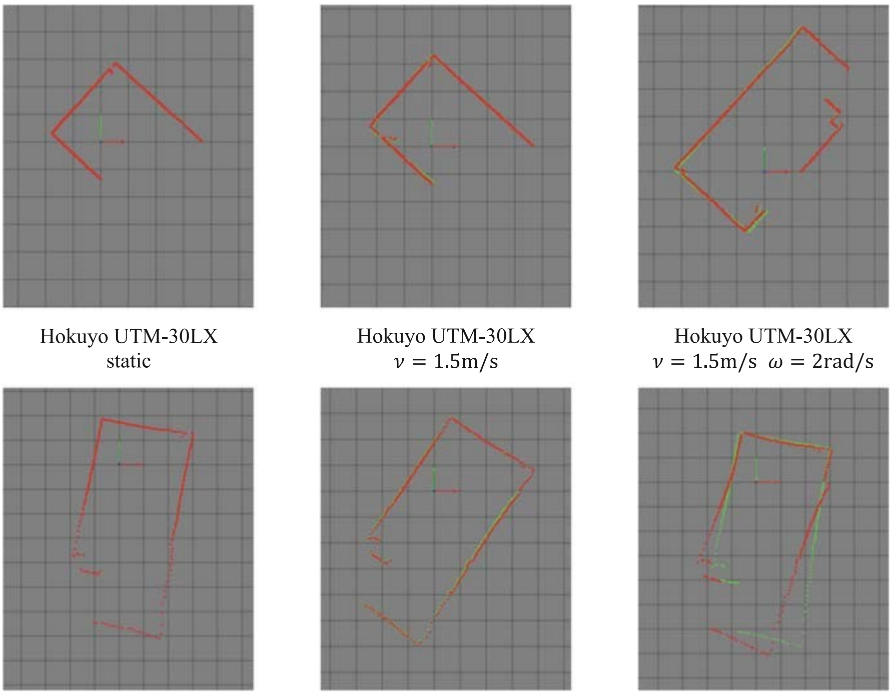
| Type | Sample (point/scan) | Range (°) | Frequency (Hz) | Std dev (mm) |
|---|---|---|---|---|
| Hokuyo UTM-30LX | 1080 | 270 | 40 | 30 |
| Inmotion ILD26TRI | 360 | 360 | 8 | 60 |

Normal linear velocity is set as $v = 1.5\,\text{m/s}$, while angular velocity is set as $\omega = 2\,\text{rad/s}$. Experiment results are shown in Fig. 6. The coordinate system shown in the figure is the LIDAR coordinate when starting scanning. The red point represents a scan of point cloud data collected by LIDAR; While the green one represents a scan of point cloud data upon correction. Starting from positive direction of X-axis, the scanning process is completed upon scanning a circle along the counterclockwise.

### 4.3 Discussion

We can see from the simulation and experiment results that the scanning frequency and movement speed of the robot are important factors affecting the measuring accuracy of point cloud. The LIDAR with low scanning frequency has a large pose change in the scanning process of a scan under the same moving state, presenting poor precision of its point cloud; The faster the movement speed, the larger influence will be exerted on the precision of the point cloud. Furthermore, rotation movement has a larger influence on the precision of cloud point compared to the translation movement. To some extent, although the correction algorithm changes the density distribution of the point cloud, but the correction results that point cloud accurately reflects the real environment to validate the necessity and effectiveness of LIDAR data correction algorithm in moving state.

Hence, we suggest that the correction algorithm is essential to the family service robot using low-cost LIDAR with low scanning frequency. Also, it shall pay attention



Hokuyo UTM-30LX static

Hokuyo UTM-30LX $v = 1.5\text{m/s}$

Hokuyo UTM-30LX $v = 1.5\text{m/s}$ $\omega = 2\text{rad/s}$

Inmotion ILD26TRI static

Inmotion ILD26TRI $v = 1.5\text{m/s}$

Inmotion ILD26TRI $v = 1.5\text{m/s}$ $\omega = 2\text{rad/s}$

**Fig. 6.** Experiment results (Color figure online)

to lower the rotational speed of such kind of robot in control. Moreover, the LIDAR shall be fixed around the rotational center of the robot to decrease offset, thus reducing high-speed translation caused by rotation.

## 5 Conclusions

In this paper, we have presented the necessity of correcting LIDAR measuring point cloud data in moving state and validated the effectiveness of the correction algorithm upon experiments. Considering pose changes of LIDAR moving platform in the scanning period, translation and rotation movement of the moving platform were overlaid with the rotational scanning movement in the algorithm, whose data can accurately reflect the real measured environment. The algorithm is suitable for processing point cloud data of 2D and 3D LIDAR flat scanning, especially the family service robot using low-cost LIDAR and the pursuit of rapid movement and guarantee no collision of the environment. The correction algorithm given in the paper can be popularized to all environmental sensing tasks using LIDAR, which can also provide algorithm references for sensor data pre-processing of the algorithm end and data encapsulation of LIDAR supplier, contributing to improving mapping, positioning, obstacle avoidance and other algorithms of robots using LIDAR data.

**Acknowledgments.** The authors thank National Engineering Research Center of Manufacturing Equipment Digitization and State Key Laboratory of Material Processing and Die & Mould Technology for supporting our work. Thank INMOTION ROBOT and Hokuyo Automation Co., LTD for providing LIDAR.

## References

1. Hokuyo Automation: Scanning range finder, distance data output type for robotics (2017). http://www.hokuyo-aut.jp
2. SICK: Detection and ranging solutions (2017). https://www.sick.com
3. Konolige, K., Augenbraun, J., Donaldson, N., Fiebig, C., Shah, P.: A low-cost laser distance sensor. In: IEEE International Conference on Robotics and Automation, pp. 3002–3008 (2008)
4. Wehr, A., Lohr, U.: Airborne laser scanning-an introduction and overview. ISPRS J. Photogramm. Remote Sens. **54**, 68–82 (1999)
5. Baltsavias, E.P.: Airborne laser scanning: existing systems and firms and other resources. ISPRS J. Photogramm. Remote Sens. **54**(2–3), 164–198 (1999)
6. Montemerlo, M., Thrun, S.: Large-scale robotic 3-D mapping of urban structures. In: ISER, Singapore (2004)
7. Kohlbrecher, S., von Stryk, O., Meyer, J., Klingauf, U.: A flexible and scalable SLAM system with full 3D motion estimation. In: IEEE International Symposium on Safety, Security, and Rescue Robotics, Kyoto, Japan, September 2011
8. Hess, W., Kohler, D., Rapp, H., Andor, D.: Real-time loop closure in 2D LIDAR SLAM. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 1271–1278 (2016)
9. The Contributors of the Robot Operating System (ROS): LaserScan Message (2017). http://docs.ros.org/api/sensor_msgs/html/msg/LaserScan.html
10. Corke, P.: Robotics, Vision and Control-Fundamental Algorithms in MATLAB, vol. 73. Springer, Heidelberg (2011). pp. 32–40
11. INMOTION ROBOT: 2D LiDAR product (2017). https://robot.imscv.com