Proceedings of The 2018 IEEE International
Conference on Real-time Computing and Robotics
August 1-5, 2018, Kandima, Maldives

Real-time LiDAR Data Assocation Aided by IMU in High Dynamic Environment

Jinhong Xu[1], Jiajun Lv[1], Zaishen Pan[1], Yong Liu[*,1] and Yinan Chen[2]

Abstract— **In recent years, with the breakthroughs in sensor technology, SLAM technology is developing towards high speed and high dynamic applications. The rotating multi line LiDAR sensor plays an important role. However, the rotating multi line LiDAR sensors need to restructure the data in high dynamic environment. Our work is to propose a LiDAR data correction method based on IMU and hardware synchronization, and make a hardware synchronization unit. This method can still output correct point cloud information when LiDAR sensor is moving violently.**

I. INTRODUCTION

In recent years, with the breakthrough of sensor technology, robot technology is being applied to high speed and high dynamic environment.Such as Boston Dynamics's extremely resistant Big Dog[8], agile Wild Cat[3], backflip Atlas, Google's Google Driverless Car[6] and more. These robots are inseparable from LiDAR in environmental perception, and LiDAR has become an indispensable and important sensor.

LiDAR works by rotating a motor with multiple laser transmitters and receivers, rotating the motor, driving the laser transceiver and sending measurement data out.After the accumulated data collected by the calculation unit exceeds 360°, all the data points are mapped to the LiDAR center coordinate system[4]. Figure 1 shows the mapping method. Because of the way LiDAR works, it can cause 360°data not captured at the same time. When the LiDAR moves in high dynamic environment[2], the data points will be distorted directly mapped into the LiDAR coordinate system. This distortion is sometimes catastrophic. See Figure 2(a),Figure 2(b). The common solution is based on software for time alignment. Such as LOAM[9], when the computer is collecting LiDAR data and IMU data[7], it will use the computer system time as the timestamp for each frame of data. The computer corrects the laser data after it has collected one frame of laser data. As the computer is not like the RTOS system[1], there is data transmission delay, the system response delay, based on this time stamping scheme can not solve the problem at the root cause.

In this paper, a time synchronization scheme based on hardware triggering is proposed, and the high precision data reconstruction of LiDAR is realized on the basis of the

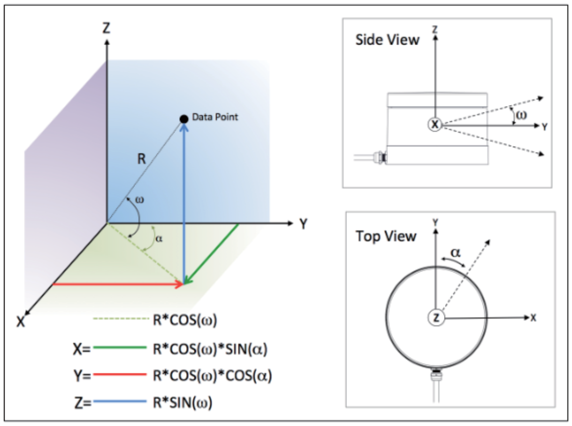
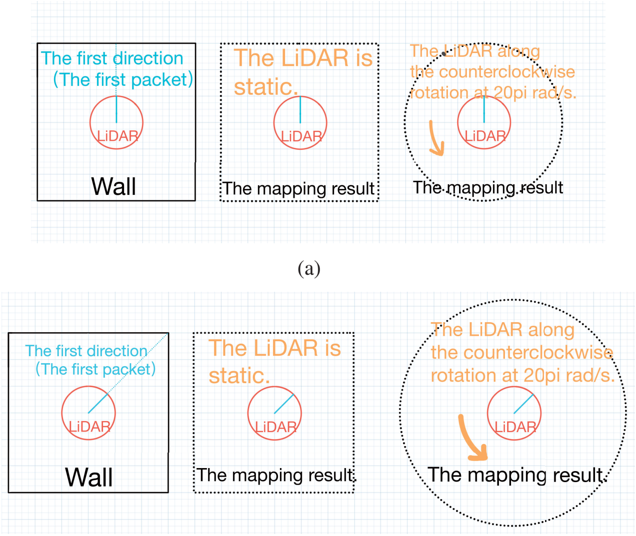


Fig. 1. The Laser point cloud computing standards.



(a)

(b)

Fig. 2. (a) The left image shows the LiDAR operating at $600RPM$ (the laser transceiver moves $20\pi/s$ clockwise), the solid line represents the surrounding obstacles, and the blue line represents the direction of the laser transceiver when the first frame of laser data begins to be acquired. The middle figure represents the point cloud reconstruction results when the whole laser radar is still. The image on the right represents the result of point cloud reconstruction when the LiDAR is moving counterclockwise at $20\pi/s$. (b) The difference between (a) is that when the first frame of laser data begins to be acquired, the direction of the LiDAR transceiver is different and the reconstruction result is also different. It can be seen that the LiDAR transceiver is stationary with respect to the wall when the LiDAR rotates counterclockwise at a speed of $20\pi/s$. However, this reconstruction is only based on the static solution of LiDAR, so this kind of extreme error occurs.

---

[1] Jinhong Xu, Jiajun Lv, Zaishen Pan and Yong Liu are with Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou 310027, China. `xujinhong1992@qq.com; cameling@qq.com; yongliu@iipc.zju.edu.cn;`

[2] Yinan Chen with Xidian University, Xi'an 710126, China. `southtreecyn@gmail.com`

[*] Yong Liu is the corresponding author.

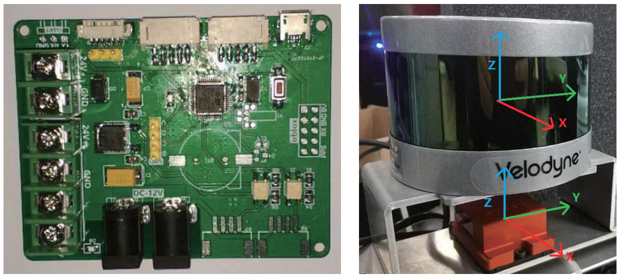
978-1-5386-6869-6/18/$31.00 © 2018 IEEE

scheme. The structure of this paper is as follows:The second section introduces the framework of the system and the time sequence.The third section introduces the synchronization method of IMU and LiDAR and the correction algorithm after synchronization.The fourth section gives the actual test results based on the system in different environments.

II. THE FRAMEWORK OF THE SYSTEM AND TIME SEQUENCE

A. The framework of the system

The system consists of LiDAR, IMU, hardware synchronization trigger board, computing unit. IMU and LiDAR fixed on the same rigid body, to ensure the same direction, shown in Figure 3. The hardware synchronization board is connected with the LiDAR through the serial port and the TTL port, and is connected with the IMU through the TTL port. It is used for sending synchronous data. The LiDAR is connected to the computer through the network port. IMU connects with the computing unit through the USB serial port.Show in Figure 4.



(a) (b)

Fig. 3. (a)The hardware synchronization trigger board. (b)The installation method of LiDAR and IMU. Ensure IMU and LiDAR remains in the same direction.

B. The framework of the time sequence

- Hardware synchronization trigger board
  The hardware synchronous trigger board simulates the GPS-supplied time signal[5] and sends it to the LiDAR through the serial port. At the same time, the PPS synchronization pulse is sent to the LiDAR and the IMU in every second of the GPS-supplied time signal.
- LiDAR and IMU
  The LiDAR motor works at $600RPM$, the output frequency of the packet is $753.58Hz$, IMU works at $400Hz$, and the output frequency of the packet is at $400Hz$. LiDAR receives GPS signals and PPS signals, and adds GPS based timestamps to each packet. IMU will capture the rising edge of PPS signal and position the synchronization symbol in the latest packet after capture.
- Computing unit
  The computing unit will collect $753.58Hz$ LiDAR data packets and $400Hz$ IMU packets. The length of the LiDAR packet is 76 (360°), which triggers a callback function. The trigger frequency is $10Hz$. The callback function of IMU is $400Hz$. However, due to the fact that the computing unit is a non real time operation system, the corresponding data acquisition callback function will have uncertain delay.

III. SYNCHRONIZATION AND CORRECTION ALGORITHM

We define a sweep as the LiDAR completes one time of scan coverage. We assume that the angular velocity of LiDAR are smooth and continues over time, without abrupt changes, and the linear velocity of LiDAR is zero.

Let p be the points received in a sweep. Let X_i^L be a point in p, $X_i^L \in p, i \in [0, 29184)$. There are 76 pckage in a sweep and each package contains the data from 24 firing sequences and the information from two firing sequences of 16 lasers is contained in one Data Block. Let t_i^L be the timestamp of LiDAR point X_i^L. IMU queue holds the last 200 data collected (about $0.5sec$). Let q_j be a quaternion collected by IMU, $j \in [0, 200)$, and t_j^i be the timestamp of q_j.

At the initialization period, the system will wait until a sweep with a second jump arrives. The exact timestamp of synchronization flag can be calculated accurately by sweep with GPS-supplied time. Then the system will check where synchronization flag located in IMU queue, with which, the whole IMU queue's timestamps are easy to know.

Recall that t_j^I and t_i^L are in the same time frame using GPS-supplied time. Let q_i be the LiDAR pose transform in t_i^L. q_i can be computed by linear interpolation of q_j

$$q_i = \frac{t_i^L - t_j^I}{t_{j+1}^I - t_j^I} + q_j, t_j^L \in [t_j^I, t_{j+1}^I] \tag{1}$$

The computing unit is a non real time operation system so recent IMU data have not been collected. q_i can be computed by linear interpolation of the IMU queue tail data q_j

$$q_i = \frac{t_i^L - t_{j-1}^I}{t_j^I - t_{j-1}^I} + q_j, t_j^L > t_j^I, j = 199 \tag{2}$$

We assume that the LiDAR and IMU have been calibrated. There is no rotation between the LiDAR and IMU and the translations are ignored. Let Xc_i^L be the position of point X_i^L after correction,

$$Xc_i^L = q_i X_i^L \tag{3}$$

Finally, we reproject the points in p to the beginning of the sweep

$$f(Xc_i^L) = q_0 Xc_i^L \tag{4}$$

q_0 is the LiDAR pose transform in t_0^L.

IV. EXPERIMENT

We designed a constant speed test device, which was tested indoors and outdoors. The speed of rotation is up to $7rad/s$. The constant speed measuring device is made up of brushless motor drives and brushless motor with deceleration. We use 3D printing to fix the sensor at the end of the motor.

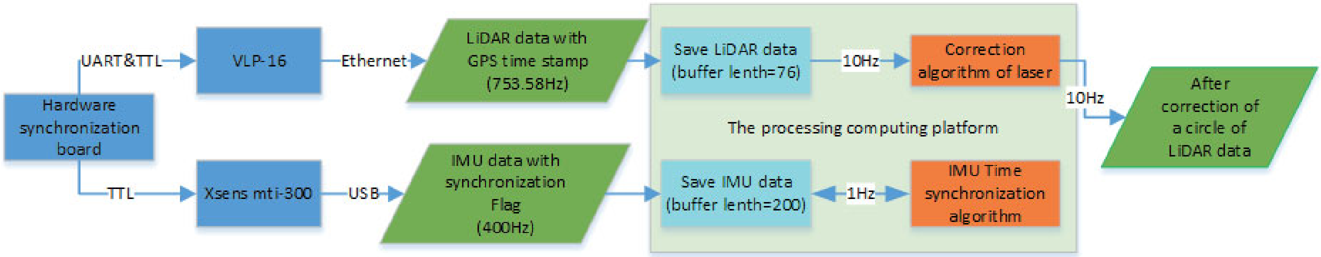


Fig. 4. The flow chart of the system.

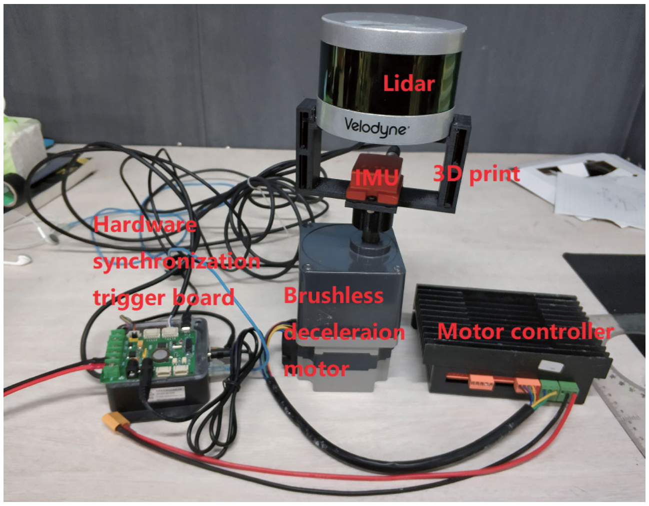


Fig. 5. A constant speed test device

The experimental results show that the original point cloud reconstruction data have serious tearing, and the point cloud after our system correction has maintained the correct shape.

A. *Indoor experiment*

This experiment was carried out in a structured cuboid chamber. The angular velocity of Z direction was $7rad/s$ when tested. It is found that the non corrected point cloud (color data) produces a serious tearing effect, while the point cloud (white data) corrected by this system remains the shape of the rectangular body.

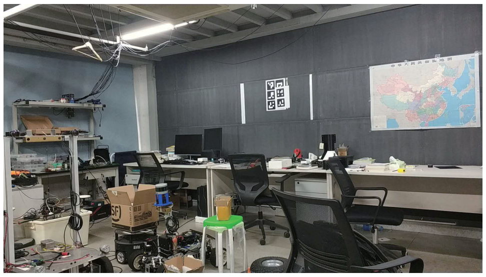
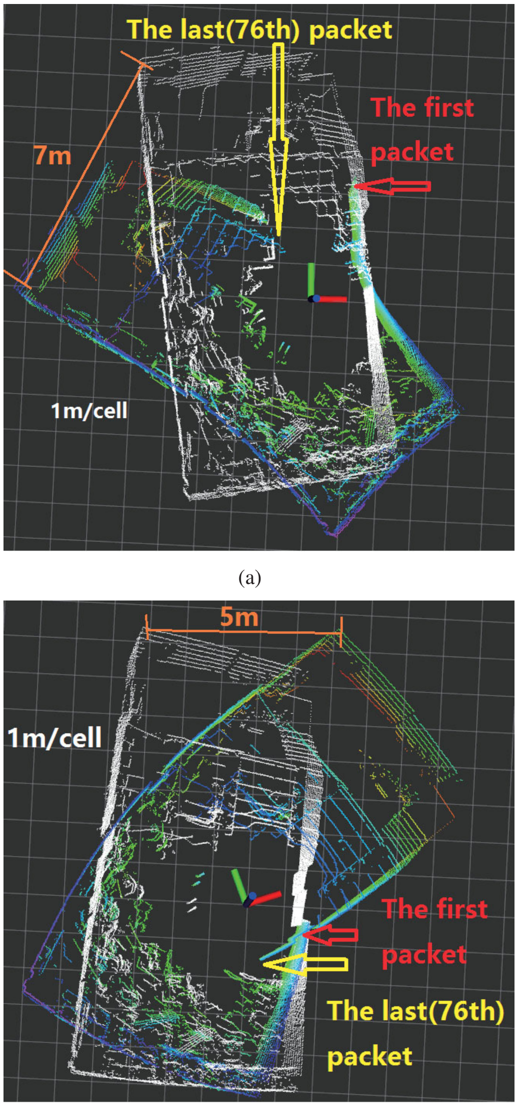


Fig. 6. Structured indoor environment.



(a)

(b)

Fig. 7. The sensor along the Z axis do violently rotate($7rad/s$) back and forth in the laboratory. Color data is the original data. White data is after real-time data processing. We observed the color data points have been torn, white data still remain intact. (a) The sonsor along the clockwise direction do strenuous rotation. The original mapping method will result in the drift of the wall of $7.8m$ by a distance of $7m$. (b) The sonsor along the counterclockwise do strenuous rotation. the original mapping method will result in the drift of the wall of $7.6m$ by a distance of $5m$.

B. Outdoor experiment

This experiment is tested in outdoor playground. When testing, the angular speed of Z direction reaches $7rad/s$. Like indoor data, there is a serious deviation from the corrected data. One of them has an error of $60m$.

Fig. 8. (a) Outdoor playground environment.

V. CONCLUSIONS

In this paper, we propose a real-time correction method for LiDAR data in high dynamic environment. We designed a set of hardware system for synchronous LiDAR and IMU, and analyzed the delay problem of sensor acquisition in real scene. At the same time, a kind of algorithm was given to solve the delay problem. Finally, based on the software and hardware system, we carried out the actual test, and achieved a good correction effect. The following work can be taken into consideration in the future work:

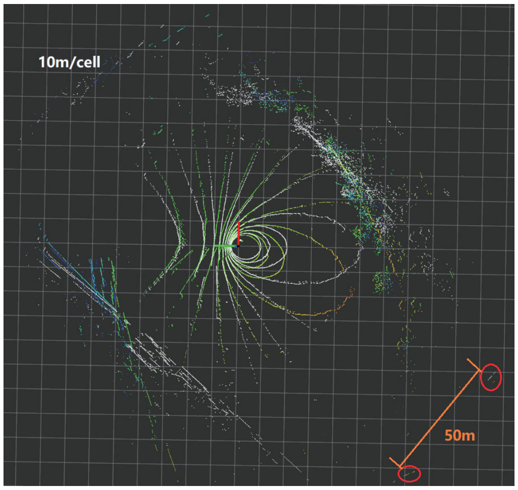
1. Displacement information is considered when the data is more positive. The displacement information between the two LiDAR data is calculated based on the SLAM system and added to the data correction part.

2. The time error exists in the IMU synchronization mark bit. There is $2.5ms$ error in the IMU synchronization flag location, so we need to design a scheme to compensate the error.

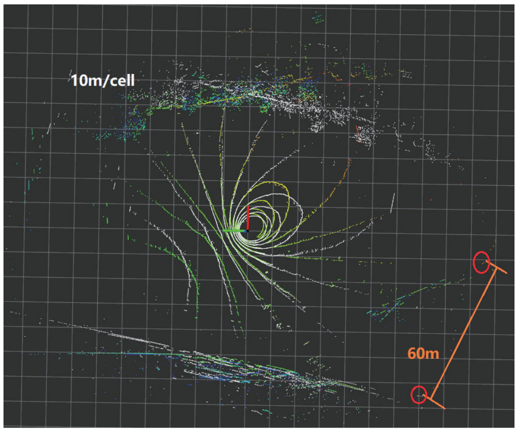
3. The coordinate relation between the LiDAR and the IMU. A high-precision calibration scheme is needed to calibrate the coordinate relationship between LiDAR and IMU.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant U1509210 and in part by the National Key Research and Development Program of China under Grant 2017YFB1302003 and by the Fundamental Research Funds for the Central Universities (2018QNA5010, 2018QNA5012).



(a)



(b)

Fig. 9. The sensor along the Z axis do violently rotate($7rad/s$) back and forth on the playground. Color data is the original data. White data is after real-time data processing. We observed the color data points have been torn, white data still remain intact. (a) The sonsor along the clockwise direction do strenuous rotation. The original mapping method will result in the drift of the ground of $118m$ by a distance of $50m$.(b) The sonsor along the counterclockwise do strenuous rotation. the original mapping method will result in the drift of the ground of $95m$ by a distance of $60m$.

REFERENCES

[1] Christian Dietrich, Martin Hoffmann, and Daniel Lohmann. *Global Optimization of Fixed-Priority Real-Time Systems by RTOS-Aware Control-Flow Analysis*. ACM, 2017.

[2] Assaf Dvir and Michal Tabach. Learning extrema problems using a non-differential approach in a digital dynamic environment: the case of high-track yet low-achievers. *ZDM*, 49(5):785–798, 2017.

[3] Mahafreed Irani. Wildcat is the latest four-legged galloping robot from boston dynamics, creators of bigdog and cheetah: A new robot funded by the us military can gallop at 16 mph and certainly is the latest bot on the block you must check out.

[4] Krishnaswamy Kannan. Development of a reference software platform for the velodyne vlp-16 lidars. 2016.

[5] Elliott D Kaplan. Understanding gps: Principles and application. *Journal of Atmospheric and Solar-Terrestrial Physics*, 59(5):598–599, 1996.

[6] Jianxi Luo, Bowen Yan, and Kristin Wood. Innogps for data-driven exploration of design opportunities and directions: The case of google driverless car project. *Journal of Mechanical Design*, 139(11), 2017.

[7] S. O. Madgwick, A. J. Harrison, and A Vaidyanathan. Estimation of imu and marg orientation using a gradient descent algorithm. In *IEEE International Conference on Rehabilitation Robotics*, page 5975346, 2011.

[8] twostroke. Boston dynamics's bigdog page. 2015.

[9] J. Zhang and S. Singh. Loam : Lidar odometry and mapping in real-time. 2014.