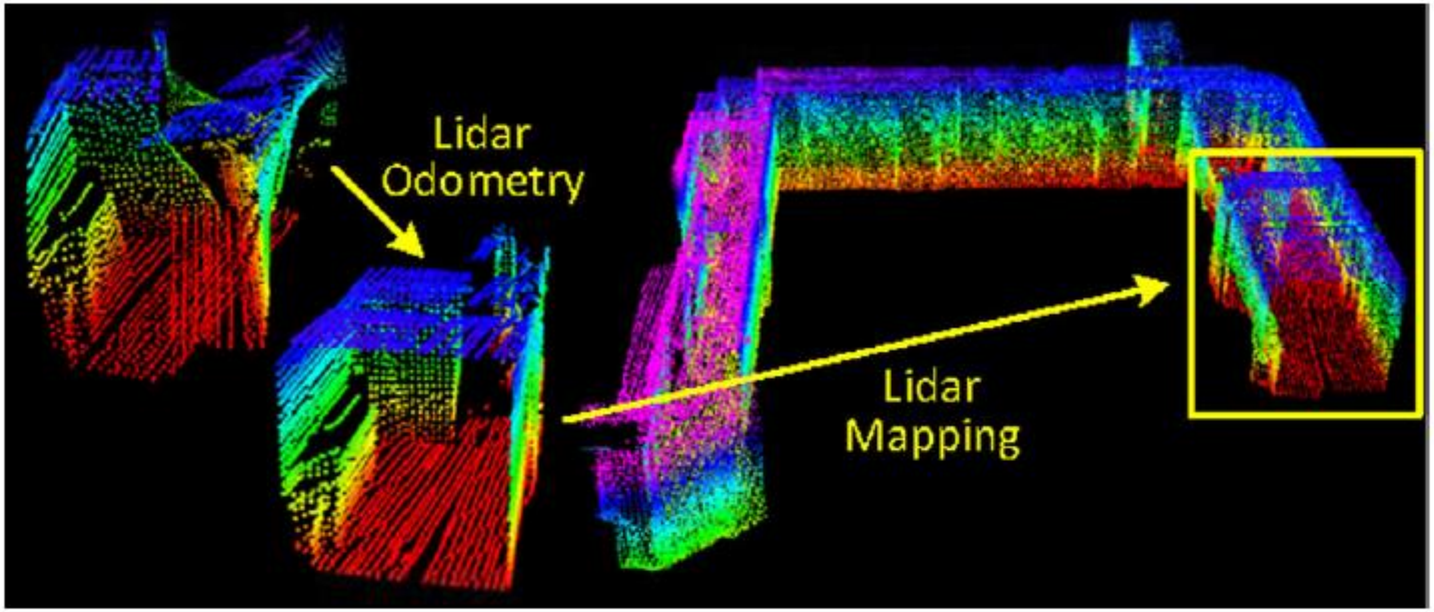
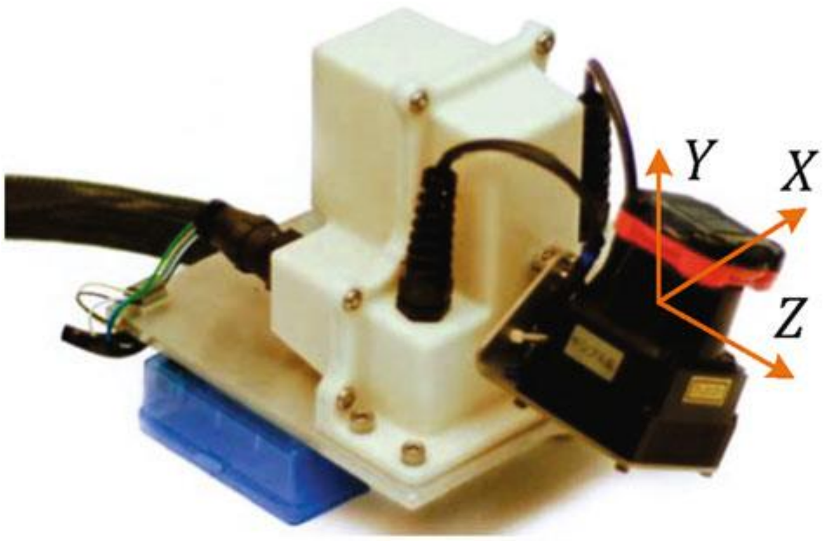
Lidar Odometry And Mapping

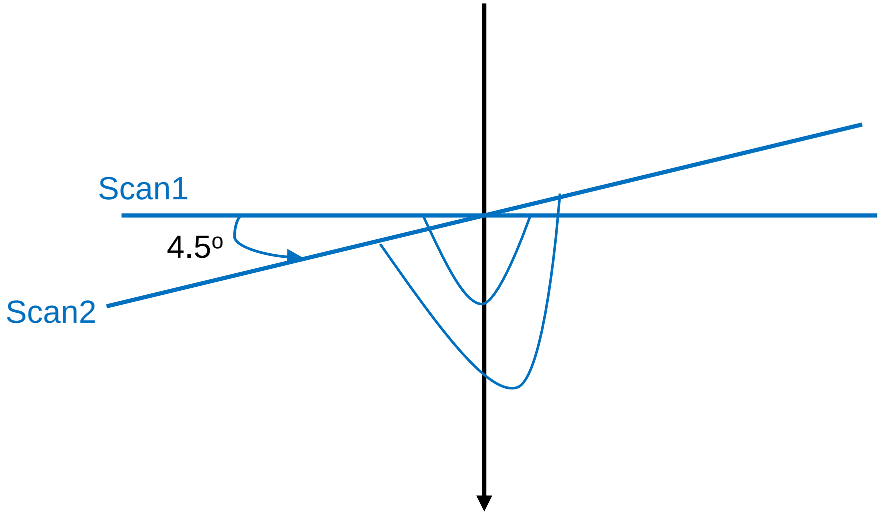
1. LOAM

- Lidar Odometry
- Lidar Mapping



1. LOAM





1) Motor- 1time/s, from -90º to 90º---One Sweep.

2) Lidar- has 180º field of view with 0.25º resolution and 40 lines/s scanning rate.

3) One sweep contains 40 scans.---40 scans== 40lines

4) Two coordinate system: lidar coordinate system - $\{L\}$
world coordinate system - $\{W\}$

1. LOAM

Signs' meaning:

1) $\hat{\mathcal{P}}$ - the points received in a laser scan.

2) $\mathcal{P}_k$ - the combined point cloud during sweep k

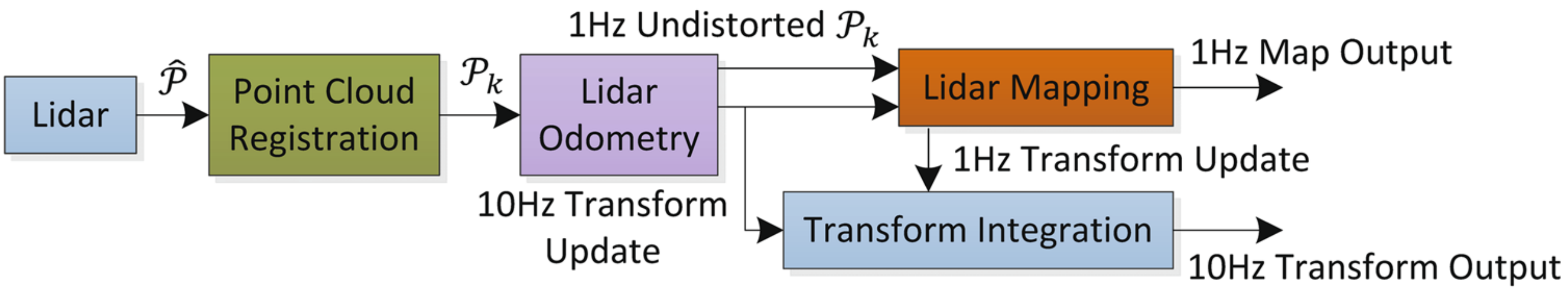
3) $\boldsymbol{X}_{(k,i)}^L$ - a point *i* received during sweep *k*

4) $\boldsymbol{T}_k^L(t)$ - the transform projecting a point received at time *t* to the beginning of the sweep *k*

Problem Given a sequence of lidar cloud $\mathcal{P}_k$, $k \in Z^+$, compute ego-motion of the lidar in the world, $\boldsymbol{T}_k^W(t)$, and build a map with $\mathcal{P}_k$ for the traversed environment.

1. LOAM

Block diagram of the lidar odometry and mapping software system



1) Transform?

2) 1HZ And 10HZ ?

1.1 LOAM-Lidar Odometry

1.1 LOAM-Lidar Odometry

Feature point extraction

2D Example- focus on a scan plane

$$c = \frac{1}{|\mathcal{S}| \cdot ||\boldsymbol{X}^{L}_{(k,i)}||} \left\| \sum_{j \in \mathcal{S}, j \neq i} (\boldsymbol{X}^{L}_{(k,i)} - \boldsymbol{X}^{L}_{(k,j)}) \right\|. \qquad (1)$$

$\mathcal{S}$ -be the set of consecutive points of i returned by the laser scanner in the same scan.

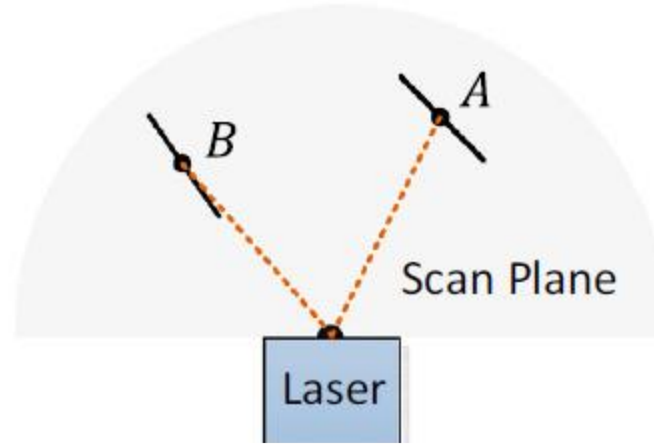
1) separate a scan into four identical subregions
2) Each subregion can providemaximally 2 edge points and 4 planar points
3) A point *i* can be selected as an edge or a planar point only if its *c* value is larger or smaller than a threshold-$5*10^{-3}$

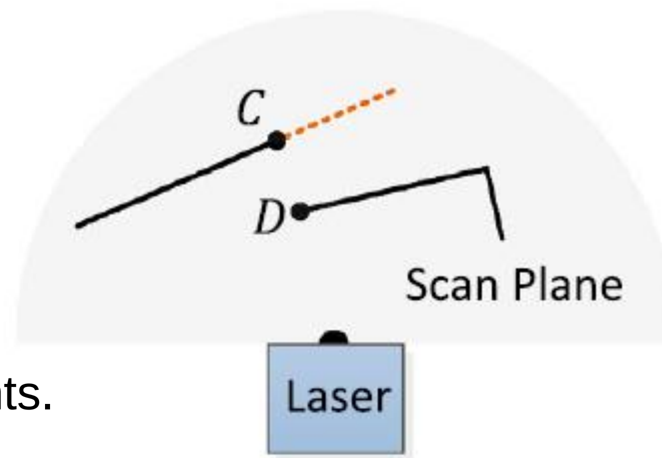
1.1 LOAM-Lidar Odometry

Feature point extraction

2D Example- focus on a scan plane-Trick

able. To avoid the aforementioned points to be selected, we find again the set of points $\mathcal{S}$. A point i can be selected only if $\mathcal{S}$ does not form a surface patch whose normal is within 10° to the laser beam, and there is no point in $\mathcal{S}$ that is disconnected from i by a gap in the direction of the laser beam and is at the same time closer to the lidar then point i (e.g. point B in Fig. 4b).





Point B and C are usually considered as unreliable points.

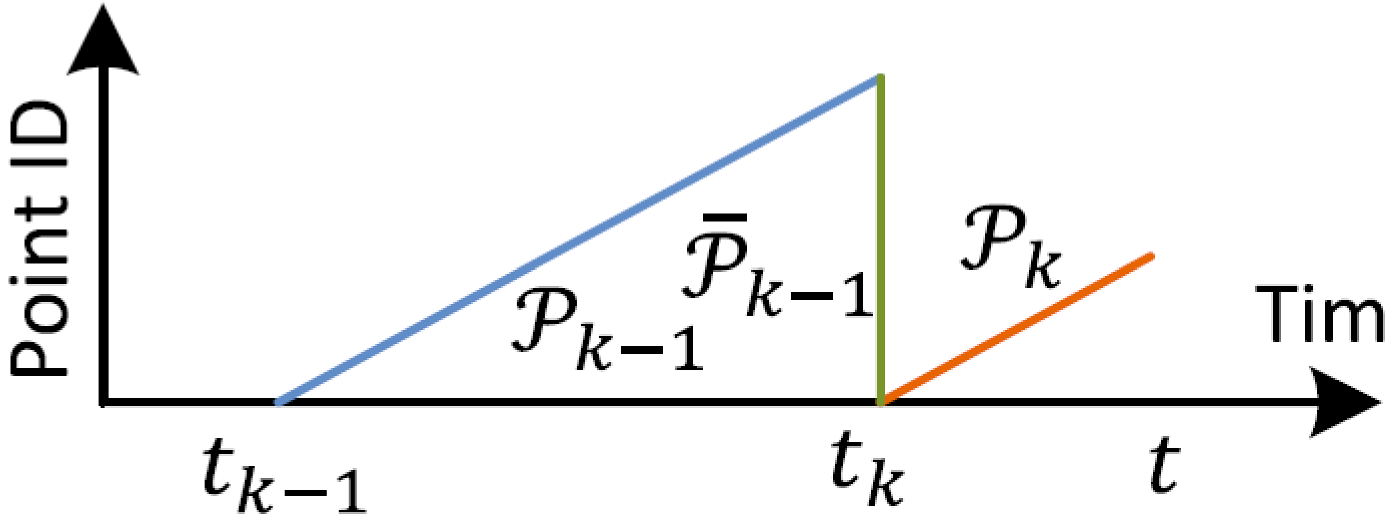
1.1 LOAM-Lidar Odometry

Feature point extraction-summary

- The number of selected edge points or planar points cannot exceed the maximum of the subregion, and
- None of its surrounding point is already selected, and
- It cannot be on a surface patch whose normal is within 10° to the laser beam, or on boundary of an occluded region.

1.1 LOAM-Lidar Odometry

Finding feature point correspondence



1.1 LOAM-Lidar Odometry

Finding feature point correspondence

Signs' meaning:

1) $\mathcal{E}_k$ - the sets of edge points during sweep k
2) $\mathcal{H}_k$ - the sets of planar points during sweep k
3) $\mathcal{E}_k$, $\mathcal{H}_k$ projected into $\tilde{\mathcal{E}}_k$, $\tilde{\mathcal{H}}_k$
4) combine $\tilde{\mathcal{E}}_k$, $\tilde{\mathcal{H}}_k$ with the points in $\bar{\mathcal{P}}_{k-1}$
5) $\bar{\mathcal{P}}_{k-1}$ - the projected point cloud.
   undistorted points modified by the transform.
6) Transform algorithm plays the important role.

1.1 LOAM-Lidar Odometry

Finding feature point correspondence-Edge Points

Find the edge line-(two points) as the correspondence of an edge point:

$i \in \tilde{\mathcal{E}}_k$

j - the closest neighbor of i in $\bar{\mathcal{P}}_{k-1}$

l - be the closest neighbor of *i* in the preceding and following two scans to the scan of j

verify both *j* and *l* are edge points or not.

1.1 LOAM-Lidar Odometry

Finding feature point correspondence-Planar Points

Find the planar patch-(three points) as the correspondence of a planar point:

$i \in \tilde{\mathrm{H}}_k$

j - the closest neighbor of i in $\bar{\mathcal{P}}_{k-1}$

l - be the closest neighbor of *i, but* in the same scan of j

m – be the closest neighbor of i, but in the preceding and following scans to the scan of j

verify both *j, l, m* are planar points or not.

1.1 LOAM-Lidar Odometry

Finding feature point correspondence

After finding the correspondence points:

$$d_{\mathcal{E}} = \frac{\left| (\tilde{X}^L_{(k,i)} - \bar{X}^L_{(k-1,j)}) \times (\tilde{X}^L_{(k,i)} - \bar{X}^L_{(k-1,l)}) \right|}{\left| \bar{X}^L_{(k-1,j)} - \bar{X}^L_{(k-1,l)} \right|}, \qquad (2)$$

Common line

$$d_{\mathcal{H}} = \frac{\left| \begin{matrix} (\tilde{X}^L_{(k,i)} - \bar{X}^L_{(k-1,j)}) \\ ((\bar{X}^L_{(k-1,j)} - \bar{X}^L_{(k-1,l)}) \times (\bar{X}^L_{(k-1,j)} - \bar{X}^L_{(k-1,m)})) \end{matrix} \right|}{\left| (\bar{X}^L_{(k-1,j)} - \bar{X}^L_{(k-1,l)}) \times (\bar{X}^L_{(k-1,j)} - \bar{X}^L_{(k-1,m)}) \right|}. \qquad (3)$$

Common plan

1.1 LOAM-Lidar Odometry

Motion estimation

Formula 2 and 3- optimization problem

Signs' meaning:

1) t - be the current time stamp

2) t_k - is the starting time of the current sweep k

3) $T_k^L(t)$- be the lidar pose transform between t_k and t

4) $\boldsymbol{T}_k^L(t) = [\tau_k^L(t), \ \theta_k^L(t)]^T$

5) $\tau_k^L(t) = [t_x, \ t_y, \ t_z]^T$

6) $\theta_k^L(t) = [\theta_x, \ \theta_y, \ \theta_z]^T$

1.1 LOAM-Lidar Odometry

Motion estimation

Formula 2 and 3- optimization problem

The projection-transform:

$$\tilde{X}^{L}_{(k,i)} = \mathbf{R}^{L}_{(k,i)} X^{L}_{(k,i)} + \tau^{L}_{(k,i)},$$

$$\boxed{\begin{aligned} & f_{\mathcal{E}}(\boldsymbol{X}^{L}_{(k,i)}, \boldsymbol{T}^{L}_{k}(t)) = d_{\mathcal{E}}, \quad i \in \mathcal{E}_k. \\ & f_{\mathcal{H}}(\boldsymbol{X}^{L}_{(k,i)}, \boldsymbol{T}^{L}_{k}(t)) = d_{\mathcal{H}}, \quad i \in \mathcal{H}_k. \end{aligned}}$$

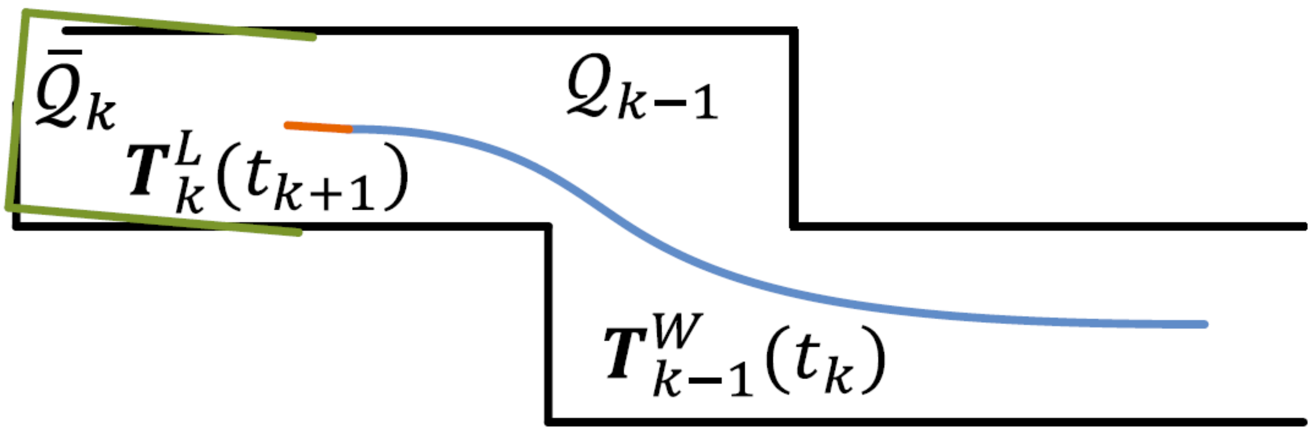
$$\boldsymbol{f}(\boldsymbol{T}^{L}_{k}(t)) = \boldsymbol{d},$$

We solve the lidar motion with the Levenberg-Marquardt method.

1.2 LOAM-Lidar Mapping

1.2 LOAM-Lidar Mapping

The mapping process



Sign Meaning:

1) $\bar{\mathcal{P}}_k$ - a undistorted point cloud at the end of sweep k

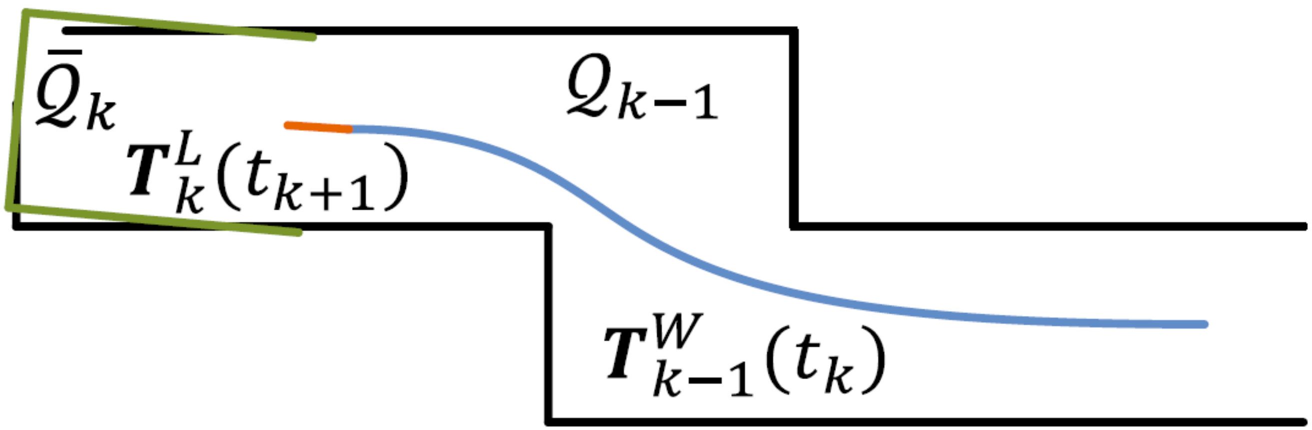
2) $\boldsymbol{T}_k^L(t_{k+1})$ - a pose transform at the end of sweep k

3) $\mathcal{Q}_{k-1}$ - the point cloud on the map, accumulated until sweep k −1

4) $\boldsymbol{T}_{k-1}^W(t_k)$ - the pose of the lidar on the map at the end of sweep k −1

1.2 LOAM-Lidar Mapping

The mapping process



t_k. With the output from lidar odometry, the mapping algorithm extents $\boldsymbol{T}_{k-1}^W(t_k)$ for one sweep from t_k to t_{k+1}, to obtain $\boldsymbol{T}_k^W(t_{k+1})$, and transforms $\bar{\mathcal{P}}_k$ into the world coordinates, $\{W\}$, denoted as $\bar{\mathcal{Q}}_k$. Next, the algorithm matches $\bar{\mathcal{Q}}_k$ with $\mathcal{Q}_{k-1}$ by optimizing the lidar pose $\boldsymbol{T}_k^W(t_{k+1})$.

1.2 LOAM-Lidar Mapping

Contrast the odometry with the mapping

Odometry

$$\left.\begin{array}{l} P_{k-1} \xrightarrow{transform} \bar{P}_{k-1} \\ P_k \xrightarrow{extract} \left\{\begin{array}{l} \varepsilon_k \\ \\ H_k \end{array}\right. \xrightarrow{transform} \left\{\begin{array}{l} \tilde{\varepsilon}_k \\ \\ \tilde{H}_k \end{array}\right. \end{array}\right\} \textit{form an optimization problem under the transform to get } \bar{P}_k \textit{ and } T_k^{\,L}(t_{k+1})$$

Mapping

$$\left.\begin{array}{l} \vartheta_{k-1} \\ \bar{P}_k \xrightarrow{L\ to\ W} \bar{\vartheta}_k \xrightarrow{extract} \left\{\begin{array}{l} \varepsilon_k \\ \\ H_k \end{array}\right. \xrightarrow{transform} \left\{\begin{array}{l} \tilde{\varepsilon}_k \\ \\ \tilde{H}_k \end{array}\right. \end{array}\right\} \textit{form an optimization problem under the transform to get } \vartheta_k \textit{ and } T_k^{\,W}(t_{k+1})$$

1.2 LOAM-Lidar Mapping

Contrast the odometry with the mapping

10Hz vs 1Hz

1) In mapping, 10 times of feature points are used.

2) Eigenvalue decomposition of covariance matrix.

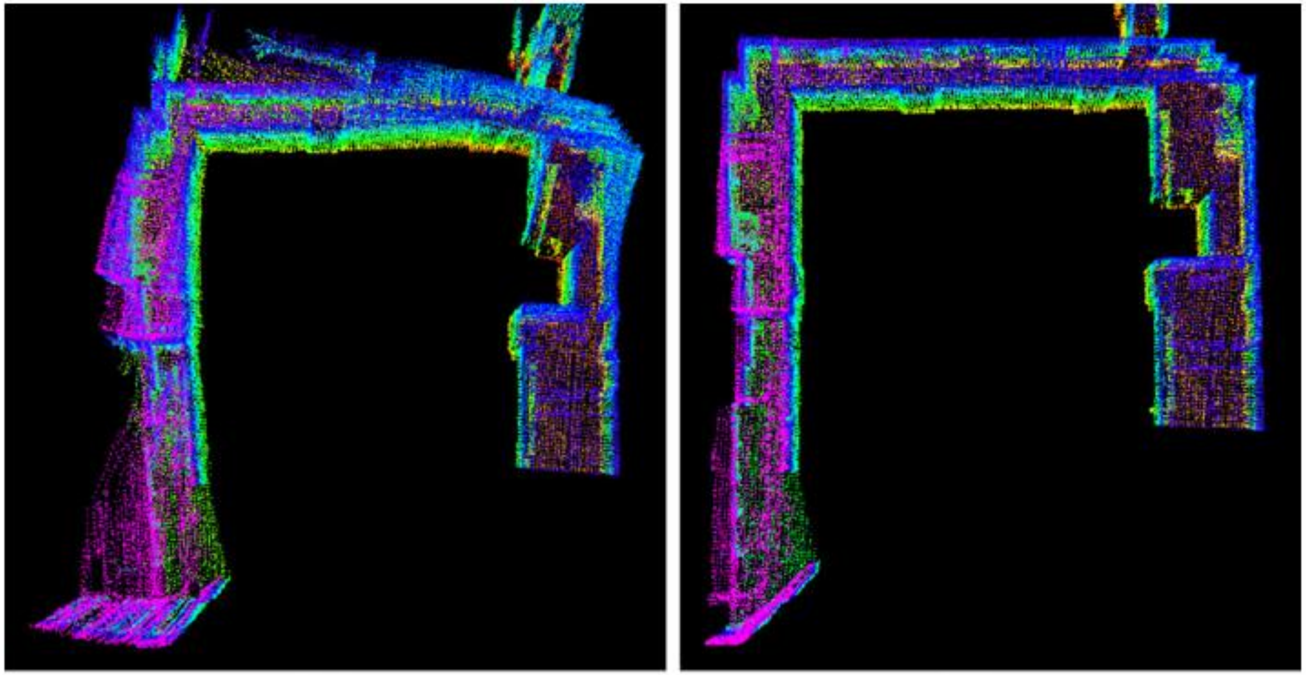
   a) if edge: to obtain the orientation of the edge line.

   b) if planar: to obtain the orientation of the planar patch.

3) voxel-grid filter.

1.3 Example

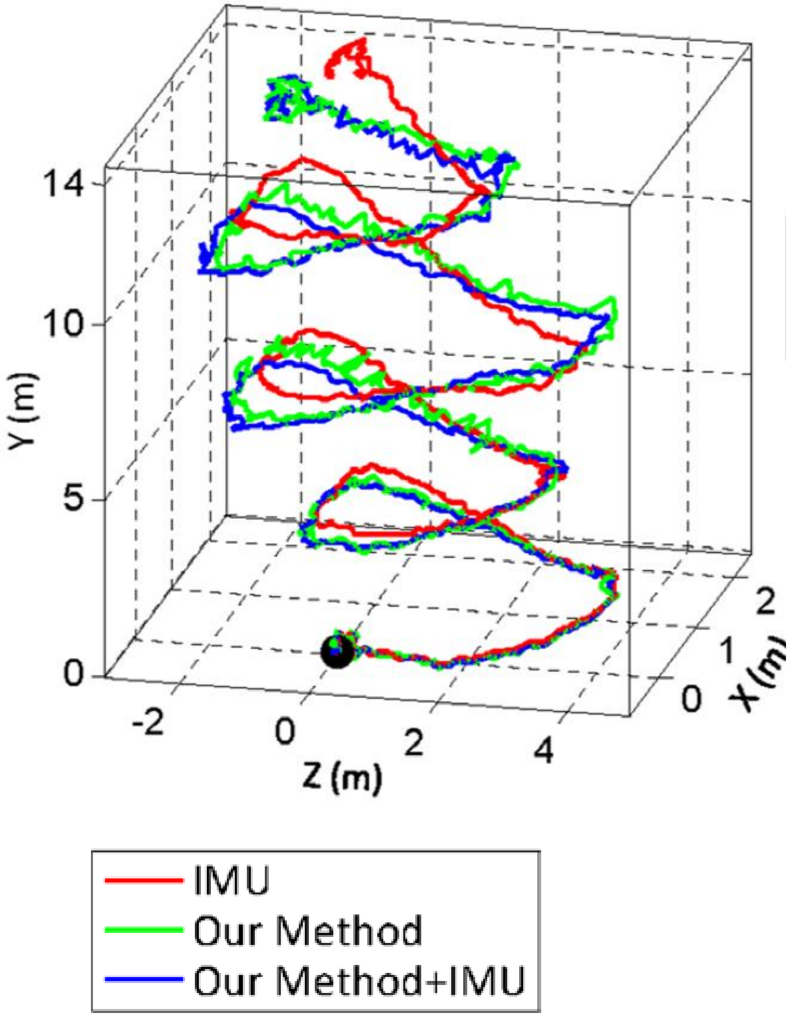
1.3 LOAM-Example

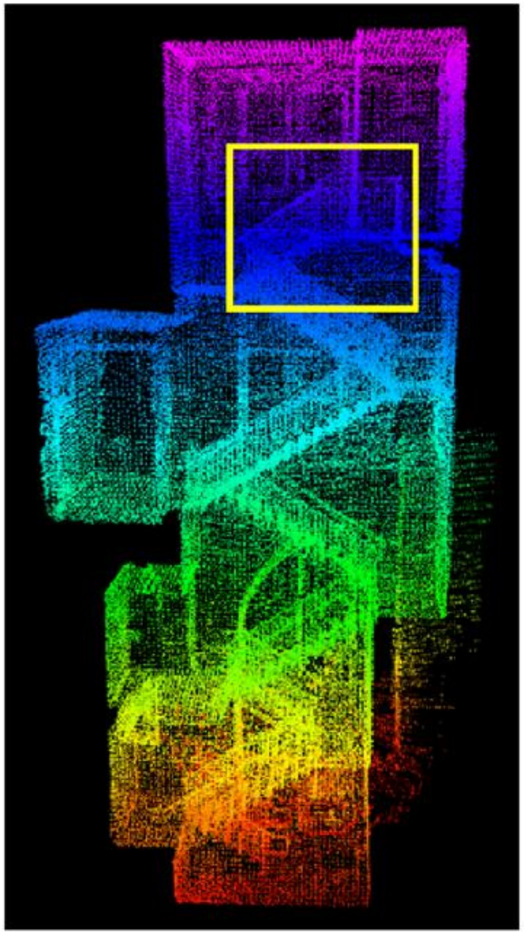


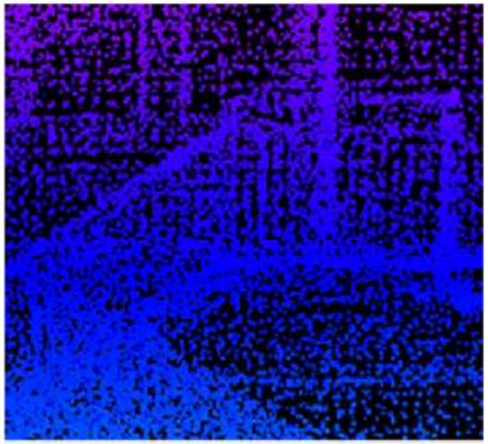
(a) (b)

Fig. 12 Comparison between **a** lidar odometry output and **b** final lidar mapping output with the dataset in Fig. 1. The role of lidar odometry is to estimate velocity and remove motion distortion in point clouds. This algorithm has a low fidelity. Lidar mapping further performs careful scan matching to warrant accuracy on the map

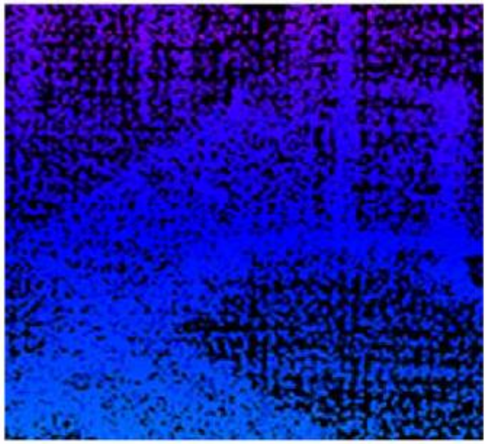
1.3 LOAM-Example







Mapping with green line



Mapping with blue line