Velodyne SLAM

08/31/2017

Subjects

- I. INTRODUCTION
- II. PROPOSED METHOD
  - a. Scan Acquisition
  - b. Pre-processing
  - c. Localization
  - d. De-Skew
  - e. Mapping
  - f. Map Refine
- III. EXPERIMENTS
- IV. CONCLUSION AND FUTURE WORK

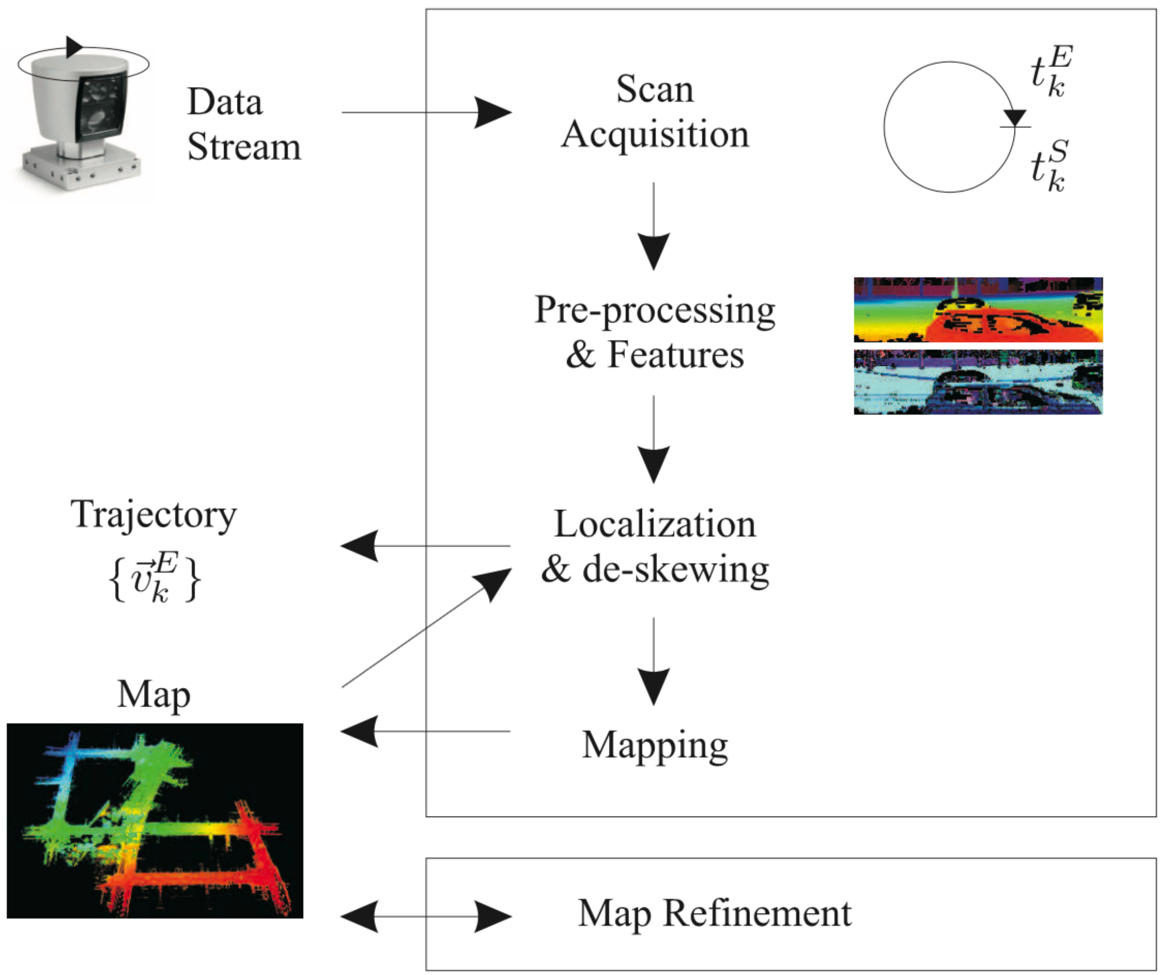
Introduction

- Two trajectory approaches:
  - Incremental: local sensors e.g. wheel-speed, inertial, camera, laser; limited history
  - Global: GPS, global map (inaccurate and not available)
  - Fusing GPS into local approaches
  - Semi-Global: SLAM

This Work

- **Laser**: precision, field of view, SLAM benefits from dense data
- Scan-matching algorithm[6] vs probabilistic techniques (computationally hard, 2D laser scanners, indoors)
- **Velodyne HDL-64E S2**, pro: high data rate, con: noise
- Handling of noise; off-line map-refinement; sensor movement

Proposed Method



A. Scan Acquisition

- HDL-64E
  - Column of 64 laser diodes
  - Pitch range 26 degrees
  - Rotation sweep 360 degree, 10/20 Hz
  - ***Scan k***: 870(?)x64 pixel

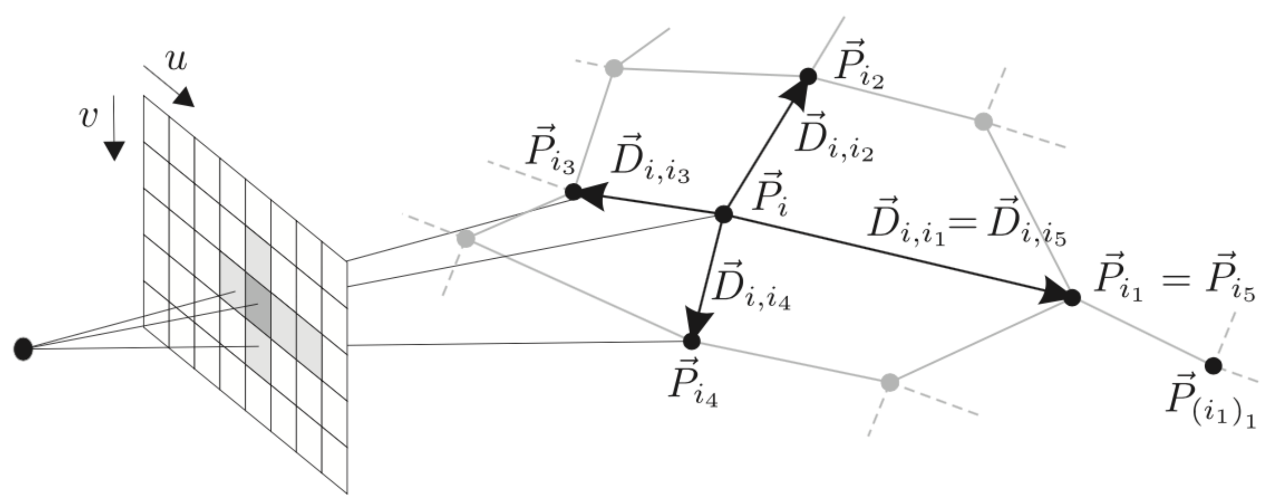
B. Pre-processing

- 2D array of range measurements, R(u, v): Ri

i1 :=(u+1,v) i3 :=(u−1,v) i5 :=i1

i2 :=(u,v−1) i4 :=(u,v+1) (i1)1 :=((u+1)+1,v)

range measurement	R_i	:	$i \mapsto r$
point coordinates	$\vec{P}_i$	:	$i \mapsto (x, y, z)^T$
distance vector	$\vec{D}_{i,j}$	=	$\vec{P}_j - \vec{P}_i$
linkage value	$L_{i,j}$	:	$i, j \mapsto l$
normal vector	$\vec{N}_i$	:	$i \mapsto (n_x, n_y, n_z)^T$
normal confidence	C_i	:	$i \mapsto c$



B1. Linkage

A pixel connection gets assigned a high linkage value if neighbouring distance vectors have similar length.

$$L_{i,i_1} = \min(\mathrm{sigm}(|\tfrac{(R_i - R_{i_1}) - (R_{i_3} - R_i)}{(R_{i_3} - R_i)}|), \\ \mathrm{sigm}(|\tfrac{(R_i - R_{i_1}) - (R_{i_1} - R_{(i_1)_1})}{(R_{i_1} - R_{(i_1)_1})}|)) \tag{1}$$

$$\mathrm{sigm}(x) = 0.5 - \frac{0.5(x - \theta_1)\theta_2}{\sqrt{1 + (x - \theta_1)^2 \theta_2^2}} \tag{2}$$

B2. Surface

local surface plain represented by its normal vector

For a given pixel with its four neighbours the normal vector is calculated as the average of the four cross products, each weighted by the product of their linkage values

$$\vec{N}'_i = \sum_{j=1}^{4} L_{i,i_j} L_{i,i_{j+1}} (\vec{D}_{i,i_j} \times \vec{D}_{i,i_{j+1}}) \qquad (3)$$

A moving average filter is then applied to the field of surface normals in order to reduce noise:

$$\vec{N}_i = \frac{\sum_{j=1}^{4} \vec{N}'_{i_j}}{|| \sum_{j=1}^{4} \vec{N}'_{i_j} ||} \qquad (4)$$

B3. Confidence

- Evaluate horizontal and vertical plain separately
- For a given con- nection from i to j, the angle of the distance vector to the plain defines a probability that the plain assumption holds for this connection:

$$C_{i,j} = \exp\{-\theta_3 \arcsin \left| \frac{\vec{D}_{i,j} \cdot \vec{N}_i}{||\vec{D}_{i,j}||} \right|^2 \} \qquad (5)$$

- limited by the maximum linkage product from the normal calculation:

$$C'_i = \min(L_i^{\max}, \max(C_{i,i_1} C_{i,i_3}\, , \, C_{i,i_2} C_{i,i_4})) \qquad (6)$$

$$L_i^{\max} = \max{}_{j=1}^{4} L_{i,i_j} L_{i,i_{j+1}} \qquad (7)$$

- **Median filtering** is afterwards applied on the 4-neighbourhood to smooth the confidence values.

C. Localization

Current scan $\{S_i = (\vec{P}_i, \vec{N}_i, C_i)\}$

Map is built from measurements, it is in this work just a collection of surfaces in a world reference frame $\{s_i = (\vec{p}_i, \vec{n}_i, c_i)\}$,

Estimate the pose of the vehicle $\vec{v}(t)$ relative to some **global coordinate frame**

Given $\vec{v}S_k = \vec{v}E_{k-1}$, estimate $\vec{v}E_k$

C. Localization

Search a pose that results in a *best fit* (thus $\vec{v}_k^E = \arg\min_v E(v)$) according to the energy

$$E(v) = \sum_{i \in \text{scan}} (\vec{n}_{\text{NN}(i)}^T (\text{T}(\vec{P}_i, v) - \vec{p}_{\text{NN}(i)}))^2 \tag{8}$$

This minimization is solved using the popular *Iterative Closest Points* algorithm (ICP)

$$\tilde{\vec{v}}_k^E = \vec{v}_{k-1}^E \oplus (\vec{v}_{k-1}^E \ominus \vec{v}_{k-1}^S) \tag{9}$$

Speed up: subset surfaces (1000 upper half, 500 lower half)

D. De-skewing

To account for sensor movement

E. Mapping

Map stored in 3D grid structure, each cell is one surface

Update-1: move the 3D point coordinate $\vec{\mathrm{P}}\mathrm{i}$ along the normal vector $\vec{\mathrm{N}}\mathrm{i}$

$$\vec{P}_i'(a) = \vec{P}_i + a \cdot \vec{N}_i \tag{10}$$

until it best represents a plain together with the neighbouring surfaces.

$$\begin{aligned} E_{S_i}(a) &= \sum_{j \in \mathrm{kNN}(S_i)} w_{ij} (\vec{n}_j^T (\vec{P}_i'(a) - \vec{p}_j))^2 \\ w_{ij} &= C_i c_j \vec{N}_i^T \vec{n}_j \end{aligned} \tag{11}$$

E. Mapping

Update-2: measurement is added to the map:

cell is non-empty, its surface s_i is replaced by the current measurement S_i in case

$$\frac{r_i - R_i}{r_i} + (C_i - c_i) > \theta_4 \qquad (12)$$

preference lies on surfaces that have a **higher normal confidence** and/or points that were captured from **lower distance**.

F. Map Refinement

purpose: to obtain a final map containing more details.

III. Future Work

1. detect and handle loop-closure appropriately as *e.g.* in GraphSLAM.
2. obstacle detection and tracking as the approach is currently limited to (nearly) static scenes.