RGBD object SLAM using quadrics for indoor environments

Thursday, January 7, 2021 2:27 PM

Introduction

- We propose an object-level SLAM using quadrics, and propose two RGB-D observation models.
- We propose a method to extract a complete ellipsoid from a single RGB-D frame
- We introduce a nonparametric pose graph to the quadrics model to solve the semantic data associations

Quadrics

- A quadrics can be expressed by a dual form Q^*, for any plane tangent to it, the relationship can be expressed compactly as a plane constraint formula:

$$\pi^T Q^* \pi = 0$$

- The dual quadric surface Q^* is projected under the camera observation P to obtain the curve C^*,

$$C^* = P Q^* P^T$$

where $P = K[R|t]$, when Q^* is an ellipsoid, C^* is an ellipse,

- As one bounding box can only constrain 4 dof, and an ellipsoid has 9 dof, at least 3 observations to constrain an ellipsoid.
- When the point cloud quality is high, the algorithm applies the complete constraint model to recover a complete ellipsoid from a single RGB-D data.
- If there is significant occlusion, the partial constraint model will be used, this model makes full use of the constrained planes generated by the bounding box

Ellipsoid generation:

A cuboid is generated from the point cloud, and the cuboid pose is

$$T_c = \begin{bmatrix} R_c & t_c \\ 0 & 1 \end{bmatrix}$$

Define Q as the inscribed ellipsoid of cuboid C.
Let $D = diag(s)$ be the diagonal matrix generated by vector s, we obtain the dual parameter Q^*:

$$Q^* = T_c \begin{bmatrix} DD^T & 0 \\ 0 & -1 \end{bmatrix} T_c^T$$

Partial constraint model

bounding box $\{b_i\}$, where $b = (x_{min}, y_{min}, x_{max}, y_{max})$
$P = K[R|t]$, each line of the bounding box can form a plane by back-projecting from the camera center.
If the line is l, the generated plane π can be obtained by

$$\pi = P^T l.$$

The bounding box's four edges can form four constraint planes S_p.
If the edge is too close to image boundary, it is removed. Afterwards, a tangent plane cluster $\bar{S}_p$ can be formed.
Assuming L contains the Q^* and object label l, a tangent constraint of plane π to L is

$$f_p(\pi, L) = \| \pi^T Q^* \pi \|_{\Sigma_d},$$

Σ_d is the constraint variance. The partial constraint function is

$$F_p(\bar{S}_p, L) = \Sigma_\pi f_p(\pi, L), \ \pi \in \bar{S}_p.$$

Nonparametric pose graph.

$$\max_{X_{0:T}, L_{0:M}, y_{0:T}} \log p(O_{1:T}, Z_{0:T}, U_{0:T}; X_{0:T}, L_{0:M}, y_{0:T})$$

$U_{0:T}$ are observations of semantic labels
$y_{0:T}$ is data association
$L_i = \{Q_i^*, l_i\}$ is object map, Q_i^* is ellipsoid parameter, l_i is semantic label.
$O_{1:T}$ is odometry
$Z_{0:T}$ is sensor observation
$X_{0:T}$ are camera poses.

The Dirichlet Process (DP) means algorithm has two steps:

1. fixing the data association y, maximize the likelihood to solve X, L.
2. fix X, L to solve y according to maximum likelihood.

In detail, its divided into following steps:

1. Initialization:
   According to odometry $O_{1:T}$, initialize poses X, initialize all objects L by considering all observations to be new objects, and initialize the object label distribution β to the initial value β_0.

2. Optimize data association:
   fix X, L, β, and update the posterior of each data association y_t^k by

   $$P_i \propto DP(i) P(U_t^k; l_i) p(Z_t^k; X_t, L_i)$$

   That is, the product of DP prior and likelihood of observation (U_t^k, Z_t^k),
   $P(U_t^k; l_i)$ is given by the mathematical model of the semantic label.
   $P(Z_t^k; X_t, L_i)$ is given by RGBD observation model of quadrics.
   Then assign y_t^k objects with maximum likelihood

   $$y_t^k = \arg\max_i P_i.$$

3. Update the object label:
   fix the data association y_t^k, update the posterior parameters of the Dirichlet distribution of the object label. Intuitively, the category with the most observations will be the category of the object.

4. Optimize poses and object parameters:
   fix data association to solve the maximum likelihood estimation of robot poses X and objects L. This is a standard SLAM problem.

5. Filter out false positives:
   treat these objects with observation number less than n_0 as false positives and filter out them.