Calibrating an Outdoor Distributed Camera Network using Laser Range Finder Data

Agustín Ortega, Ernesto Teniente, Juan Andrade-Cetto

Bruno Dias, Alexandre Bernardino, José Gaspar

IROS'09 Saint Louis, MO, USA, October 2009
Motivation

What are SLAM maps useful for?
Barcelona Robot Lab

- outdoor pedestrian area
- 10,000 sq m

- limited gps coverage
- moderate vegetation and intense cast shadows
Camera Network

- 21 cameras
- full coverage of wifi and mica devices
Barcelona Robot Lab GUI

- Distributed camera network
- Non-overlapping field of view
Robot Task Execution
Tracking Activities on Camera Network
Problem

How to calibrate such large camera network?

• Pattern-based calibration with known metric structure (Tsai, 1987; Zisserman, 2004).
  • For large outdoor camera networks, calibration patterns of reasonable sizes project on images with very small resolution.
  • A secondary process is needed to relate all camera coordinate systems to a global reference frame.

• For planar scenarios, a Direct Linear Transformation (DLT) suffices to estimate image to plane homographies (Okuma et al 2004).
  • The planar scenario assumption is too restrictive: our scenario has multiple unparallel locally planar surfaces such as ramps.
  • A secondary process is needed to relate all homographies to a 2D map.
Tracking points or robots…

- Use a bright moving spot [Svoboda et al 2005].
  - assumes overlapping fields of view to estimate the epipolar geometry of the camera network,
  - in our case the cameras’ fields of view seldom overlap, and the visibility of the bright spot does not always hold at sunlight.

- Track a moving object assuming a constant velocity model for the target (Rahimi et al 2004).

- Place the led light on a moving robot and have a secondary robot equipped with a laser sensor tracking the first one, relating their position estimates to the camera network (Yokoya et al 2008).
  - Deployment for recalibration is costly.
Our solution

How to calibrate such large camera network?

Answer:

A semi-automatic method with the aid of our SLAM-made dense range map.
Camera Network

LRF Data

Coarse estimation of camera parameters

3D to 2D feature matching with nonlinear optimization of camera parameters

Plane and line segmentation

Computation of homographies of the walking areas
Coarse estimation of camera parameters

3D to 2D feature matching with nonlinear optimization of camera parameters

Plane and line segmentation

Computation of homographies of the walking areas

Camera Network

LRF Data
Coarse estimation of camera parameters

1. Nominal Calibration
Coarse estimation of camera parameters

• Intrinsic parameters are set to default values
  • 40 deg fov for an 8mm lens in a ¼ in CCD.

• Adjust extrinsic parameters manually
  • Initial position: 6m above the floor.
  • Initial elevation: 17 deg, imaging objects at 20m on ground plane.
Plane segmentation

3D laser map
Plane segmentation

3D laser map

Search for kNNs to each point using ANN library (Mount 97) $O(N \log n)$
Plane segmentation

Search for kNNs to each point using ANN library (Mount 97) $O(N \log n)$

Fit normals to local planar patches $O(N)$

$$
\epsilon = \sum_{i \in K} (\mathbf{p}_i^T \mathbf{n} - d)^2
$$

$$
\epsilon = n^T \left( \sum_{i \in K} p_i p_i^T \right) n - 2d \left( \sum_{i \in K} p_i^T \right) n + |K|^2 d^2
$$

$$
\left( Q - \frac{q q^T}{|K|^2} \right) n = \lambda n
$$
Plane segmentation

3D laser map

Search for kNNs to each point using ANN library (Mount 97) $O(N \log N)$

Fit normals to local planar patches $O(N)$

Sort inter-point distances in increasing order $O(N \log N)$

__
__
__
__
...

Plane segmentation

3D laser map

- Search for kNNs to each point using ANN library (Mount 97) $O(N \log N)$
- Fit normals to local planar patches $O(N)$
- Sort inter-point distances in increasing order $O(N \log N)$

Grow forest of disjoint trees
Plane segmentation

Search for kNNs to each point using ANN library (Mount 97) $O(N \log N)$

Fit normals to local planar patches $O(N)$

Sort inter-point distances in increasing order $O(N \log N)$

Grow forest of disjoint trees
Plane segmentation

Merging criteria:

- Inter point distance
  \[ \frac{k_1 d_1 + k_2 d_2}{k_1 + k_2} < t_d, \]

  \[ d_1 = (p_1 - p_2)^T n_2, \]
  \[ d_2 = (p_2 - p_1)^T n_1, \]

- Local curvature
- Segment curvature

  \[ |\cos^{-1}(n_1^T n_2)| < t_c, \]
Plane segmentation

3D laser map

Grow trees using union by rank and path compression in $O(N)$

choosing as tree root the one with larger cardinality

all nodes on a tree point to its parent

(a) before union by rank
(b) after union by rank
(c) path compression
The algorithm is capable of segmenting maps with over 8 million points and with accuracies that range from 5 to 20 cm.
Line extraction

- Extract lines by intersecting all near orthogonal planes.
Coarse estimation of camera parameters

3D to 2D feature matching with nonlinear optimization of camera parameters

Plane and line segmentation

Computation of homographies of the walking areas

Camera Network

LRF Data
Feature matching

- Extract lines intersecting orthogonal planes
- Project 3D lines on images.
- Manually match candidates.
Optimization

- Minimize feature projection error using Levenberg Marquardt over the calibration parameters

\[ \hat{\theta}_j = \arg\min_{\theta_j} \sum_i \| m_i - h(P(\theta_j) \cdot M_i) \|^2 \]

- \( \hat{\theta}_j \) – calibration parameters \( (\alpha_u, \alpha_v, u_0, v_0, R, t) \)
- \( m \) – image points
- \( M \) – 3D points
- \( P \) – perspective transformation matrix
- \( h \) – homogeneous coordinates scaling function
Optimization

- Minimize feature projection error using Levenberg Marquardt over the calibration parameters

\[
\hat{\theta}_j = \arg\min_{\theta_j} \sum_i \| m_i - h(P(\theta_j) \cdot M_i) \|^2
\]

(c) After optimization

(d) After optimization
Other calibration results
When to recalibrate?

• Internal parameters once calibrated seldom change.

• Camera position is also fixed.

• Mostly, only orientation is subject to some variation due to weather conditions.

• Distortion effects not considered.
Camera Network

LRF Data

Coarse estimation of camera parameters

3D to 2D feature matching with nonlinear optimization of camera parameters

Plane and line segmentation

Computation of homographies of the walking areas
Computing homographies

• Use GUI to select planar patches from the 3D map of the walking areas.
• Project these points to images and compute homographies using DLT.
• There might be various planar regions on the same image, compute individual homographies for each.
Computing homographies
Room for improvement

• Match 3D to 2D features fully automatically

• Optimize using line parameters directly
Thanks!