Outdoor SLAM using Visual Appearance and Laser Ranging

P. Newman, D. Cole and K. Ho

ICRA 2006

Jackie Libby
Advisor: George Kantor
Main Idea

• laser and camera for 3d SLAM system
• Laser: builds 3d point cloud map
• Camera: detects loop closure from sequences of images
• First working implementation in outdoor environment
Outline

• Slam framework
• Laser data representation
• How loop is detected with vision
• How loop is closed once detected
• results
SLAM representation: state

\[ x(k + 1|k) = \left[ \begin{array}{c} x(k|k) \\ x_{vn}(k|k) \oplus u(k + 1) \end{array} \right] = \left[ \begin{array}{ccc} x_{v1}^T & \ldots & x_{vn}^T & x_{vn+1}^T \end{array} \right]^T (k + 1|k) \]

\( x(k+1|k) \) = state vector of vehicle poses from \( t = 1, \ldots, k+1 \) given observations from \( t = 1, \ldots, k \)

\( x_{vn} \) = \( n^{th} \) vehicle pose in state vector

\( u(k+1) \) = odometry between \( k \) and \( k+1 \)

\( \oplus \) = \( SE_3 \) transformation composition operator (motion model)
SLAM representation: covariance

\[
P(k + 1 | k) = \begin{bmatrix}
P(k | k) & P_{vp}(k + 1 | k) \\
P_{vp}(k + 1 | k)^T & P_v(k + 1 | k)
\end{bmatrix}
\]

\(P_v = \text{covariance of newly added vehicle state (bottom left)}\)

\(P_{vp} = ? \) (my guess: covariance between new state and all previous states)
Scan-match framework

• Nodding “yes-yes” laser
  – Returns planar scans at different elevations
• each vehicle pose corresponds to a laser scan
  – $x_v(k) \rightarrow S_k$
  – timesteps not constant
• $x_v(i), x_v(j) \rightarrow S_i, S_j \rightarrow T_{i,j}$
• $T_{i,j} =$ rigid transformation
  – Observation $\rightarrow$ EKF equations
  – i, j sequential $\rightarrow$ $T_{i,j}$ can replace u
  – loop closing $\rightarrow$ $j >> i$
  • $T_{i,j}$ used to correct position
State vector with growing uncertainty

- outdoor data set
- x,y,z grid (20m marks)
- $1 \sigma$ x,y,z uncertainty ellipsoids
- Effect of long loops

How to detect loop closure
1) slam pdf, small Mahalanobis distance
   - Ellipsoids overconfident -> this method fails
2) Vision system
   - Matches image sequences
   - similarity in appearance
My work with SLAM:

- **CASC**
  - Advisor: George Kantor
  - Sanjiv Singh
  - Marcel Bergerman
  - Ben Grocholsky
  - Brad Hamner

- Retroreflective cones
- **no-no nodding SICK laser**
Detecting loop closure (high level)

**Assumption:**
2 images look similar -> close in space
Not close in time -> loop detected

**Problem:**
Just one image pair -> false positive
“repetitious low level descriptors”
“common texture”
leaves, bricks on buildings
“background similarity”
“common large scale features”
plants, windows

**Solution:**
sequences of image pairs increases confidence
Detect interest points

SIFT Descriptors

Clustered into words

Vocab = set of all words

Assign weight to each word

Image = vector of weights

Similarity between two images
• Harris Affine detector
• Scale invariant, affine invariant
• Example: corners -> still a corner from any angle or scale

Thanks Martial Hebert!
Detect interest points

SIFT Descriptors \{d_1, \ldots, d_n\}

16x16 pixel window around interest point
Assign each pixel a gradient orientation (out of 8 values)
For each 4x4 window, make histogram of orientations
16 histograms * 8 values = 128 = dimension of SIFT vector

Thanks Martial Hebert!
SIFT Descriptors \( \{d_1, ..., d_n\} \)

Clustered into words, \( \hat{d}_i \)

Vocab = set of all words \( V = \{\hat{d}_1, ..., \hat{d}_n\} \)

- SIFT Descriptors: \( n \) is different for different images
- word, \( \hat{d}_i = \{d_1, ..., d_k\} \)
  - clustering happens in an offline learning process
- Vocabulary, \( V \)
  - future work: different vocabularies for different settings
  - urban, park, indoors
Vocab = set of all words \( V = \{ d_1, ..., d_n \} \)

- The more frequent the word, the less descriptive it is
- inverse weighting frequency scheme *:

\[
\hat{d}_i \rightarrow w_i = \frac{\log N}{n_i}
\]

\( w_i \) = weight for index word, \( \hat{d}_i \)
\( N \) = total # images
\( n_i \) = # images where this word appears

weight for each word

\[ d_i \rightarrow w_i = \frac{\log N}{n_i} \]

Image \rightarrow vector of weights

\[ I_u = [u_1 \ldots u_{|\mathcal{V}|}]^T \]

\[ u_i = \begin{cases} w_i & \text{if for } \hat{d}_i \in I_u \\ 0 & \text{otherwise.} \end{cases} \]

- Image has been transformed into a vector
- vector is long, \(|\mathcal{V}|\), but many elements are zero
Similarity

\[ S(u, v) = \frac{\sum_{i=0}^{\left\lvert V \right\rvert} u_i v_i}{\sqrt{\sum_{i=0}^{\left\lvert V \right\rvert} u_i^2} \sqrt{\sum_{i=0}^{\left\lvert V \right\rvert} v_i^2}} \]

Cosine distance:

\[ \cos \theta = \frac{a \cdot b}{\|a\| \|b\|} \]

\[ \cos(0) = 1 \]

The closer the vectors \(\rightarrow\) the smaller the angle between them the greater the similarity.
Image, $I_u$  
Detect interest points  
SIFT Descriptors $\{d_1, ..., d_n\}$  
Clustered into words, $d_i$  
Vocab = set of all words $V = \{\hat{d}_1, ..., \hat{d}_n\}$  
weight for each word $\hat{d}_i \rightarrow w_i = \frac{\log N}{n_i}$  
Image $\rightarrow$ vector of weights $I_u = [u_1, ..., u_{|V|}]^T$  

Similarity $S(u, v) = \frac{\sum_{i=0}^{|V|} u_i v_i}{\sqrt{\sum_{i=0}^{|V|} u_i^2} \sqrt{\sum_{i=0}^{|V|} v_i^2}}$
Similarity Matrix

- Similarity matrix, $M$
  - $M_{i,j} = S(i,j)$
  - darker means more similar

- axes = timesteps
  - same for $x$ and $y$
  - comparing each image against every other one
  - main diagonal is line of reflection

- Loop closure = off diagonal streaks
  - $a_i \leftrightarrow b_i$
  - $a_{i+1} \leftrightarrow b_{i+1}$

- Boxes = false positives
  - one to many mapping:
    - $a_i \leftrightarrow b_{i+1}, b_{i+2}, b_{i+3} ...$
    - $b_i \leftrightarrow a_{i+1}, a_{i+2}, a_{i+3} ...$
  - causes:
    - vehicle stopped
    - repetitive low-level structure (windows, bricks, leaves)
    - distant images
Sequence extraction (finding streaks)

Maximal cumulative similarity

\[ H_{i,j} = \begin{cases} 
H_{i-1,j-1} + M_{i,j} & \text{if } H_{i-1,j-1} \text{ maximal,} \\
H_{i,j-1} + M_{i,j} - \delta & \text{if } H_{i,j-1} \text{ maximal,} \\
H_{i-1,j} + M_{i,j} - \delta & \text{if } H_{i-1,j} \text{ maximal} \\
\alpha \max(H_{i-1,j-1}, H_{i,j-1}H_{i-1,j}) & \\
0 & \text{otherwise}
\end{cases} \]

- Modified Smith-Waterman algorithm
- dynamic programming
- penalty terms avoid boxes
  - allow for curved lines (i.e. change in velocity)
- \( \alpha \) term allows gaps
- Maximum \( H_{i,j} = \eta_{A,B} \)
Removing “common mode similarity” (finding boxes)

Decompose $M$ into sum of outer products

$$M = \sum_{i=1}^{N} v_i \lambda_i v_i^T$$

“Dominant structure” = repetitive structure

dominant structure $\rightarrow$ largest eigenvalues/vectors

Eigenface:

More repetition $\rightarrow$ more range in eigenvalues

Relative significance:

$$\rho(r) = \frac{\lambda_r}{\sum_{k=r}^{n} \lambda_k}$$

Maximize entropy:

$$H(M, r) = -\sum_{k=r}^{n} \rho(r) \log(\rho(r))$$
Sequence Significance

• problem:
  Maximum $H_{i,j} = \eta_{A,B} \rightarrow$ this doesn’t mean there’s a loop

• solution:
  • randomly shuffle rows and columns of $M$, recompute $\eta_{A,B}$
  • look at distribution:

Extreme value distribution (EVD)

$$p(\eta_{A,B}) = \frac{1}{\beta} \exp^{-z} \exp^{-\exp^{-z}}, \quad z = \frac{\eta_{A,B} - \mu}{\beta}$$

Probability that sequence could be random

$$P(\eta \geq \eta_{A,B} \mid M) = 1 - \exp^{-\exp^z}$$

$\eta_{A,B} = \text{real score}$

$\eta = \text{random score}$

threshold at 0.5%
Estimating Loop Closure Geometry

• We have detected a loop (with image sequence)
• Now how do we close it? (how do we find $T_{ij}$?)

• One solution: iterative scan matching
  – $\eta_{A,B} \rightarrow a_i, b_j \rightarrow x_{vi}, x_{vj} \rightarrow S_i, S_j \rightarrow T_{ij}$
  – Problem: local minima

• Better solution: projective model
  – Essential matrix
  – 5 point algorithm with Ransac loop
  – User lasers to:
    • Remove scale ambiguity
    • Fine-tune with iterative scan matching
      – quality of final scan match is another quality check
Enforcing loop closure

- Naïve method: single EKF update step
  - only works for small errors, because of linear approximation

- Better method:
  - constrained non-linear optimization
  - incremental changes

\[ [x_{v1}, \ldots, x_{vn}] \rightarrow [T_{1,2}, \ldots T_{n-1,n}, T_{n,1}] \]
\[ [\Sigma_{1,2}, \ldots, \Sigma_{n,1}] \text{ (from scan-matching)} \]

Want new poses, \([T^*_{1,2}, \ldots T^*_{n-1,n}, T^*_{n,1}]\]

Minimize:

\[
C(T^*) = \sum_{i=1}^{n+1} (T^*_i - T_i)^T \Sigma_n^{-1} (T^*_i - T_i)
\]

Subject to constraint:

\[
T_1 \oplus T_2 \oplus \cdots T_n \oplus T_{n+1} = 0
\]
Results

• successfully applied to several data sets
• 98% runtime spent on laser registration -> bottleneck
• 1/3 real time
• most expensive part of vision subsystem is feature detector/descriptor
  • (Harris Affine/ SIFT)
Conclusions and future work

• Conclusions
  – SLAM system for outdoor applications
  – Works for challenging urban environment
  – Complementary vision laser system
    • Vision for loop closing
    • Laser data for geometry map building
  – First working implementation

• Future work
  – SLAM formulation not efficient
  – Laser scan matching is bottle neck
  – Learning vocabularies for distinct domains
    • (urban, park, indoors)
    • Different similarity matrix if domain switches