Algorithms for Mobile Robot Localization and Mapping,

Incorporating Detailed Noise Modeling and Multi-scale Feature Extraction

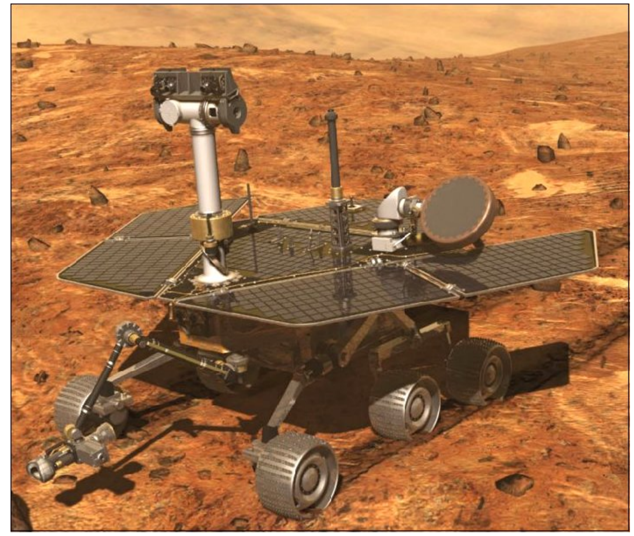
Samuel T. Pfister

April 14, 2006

Mobile Robot Navigation

Navigation Applications

- Unmanned exploration
- Convoys for military supplies
- Autonomous highway driving



Robot localization is critical for:

- Effective path planning
- Accurate construction and use of global maps

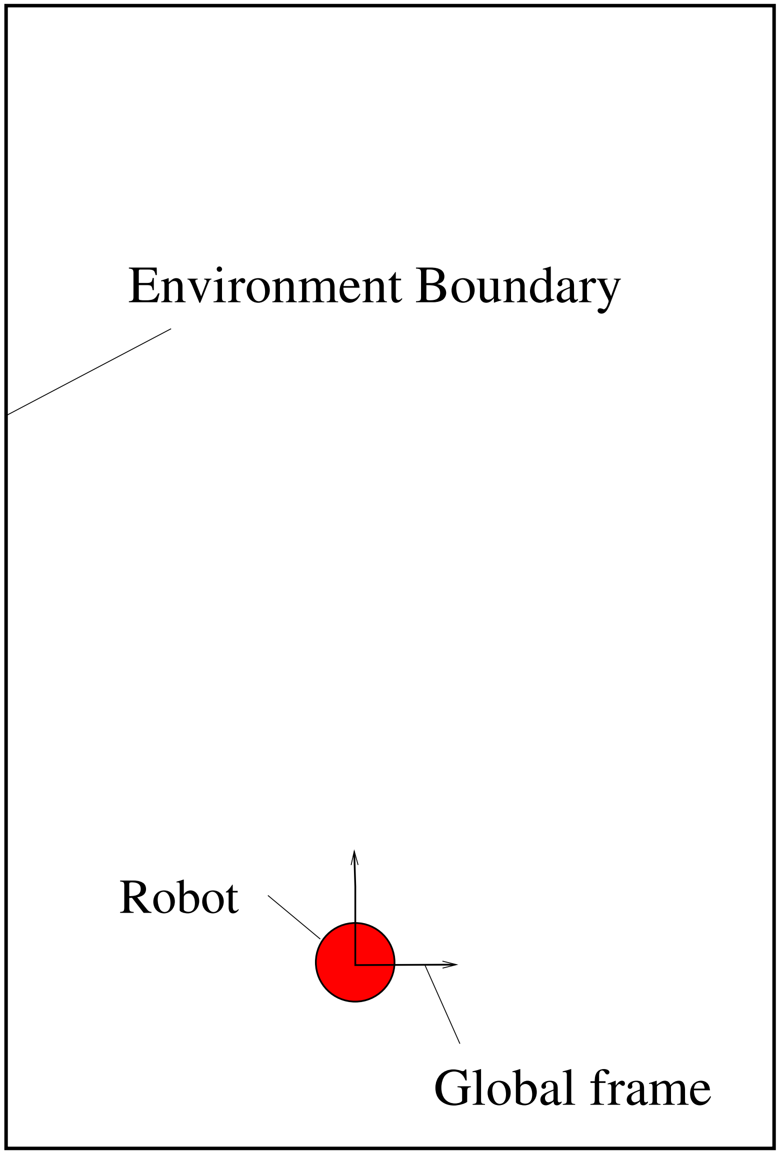
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



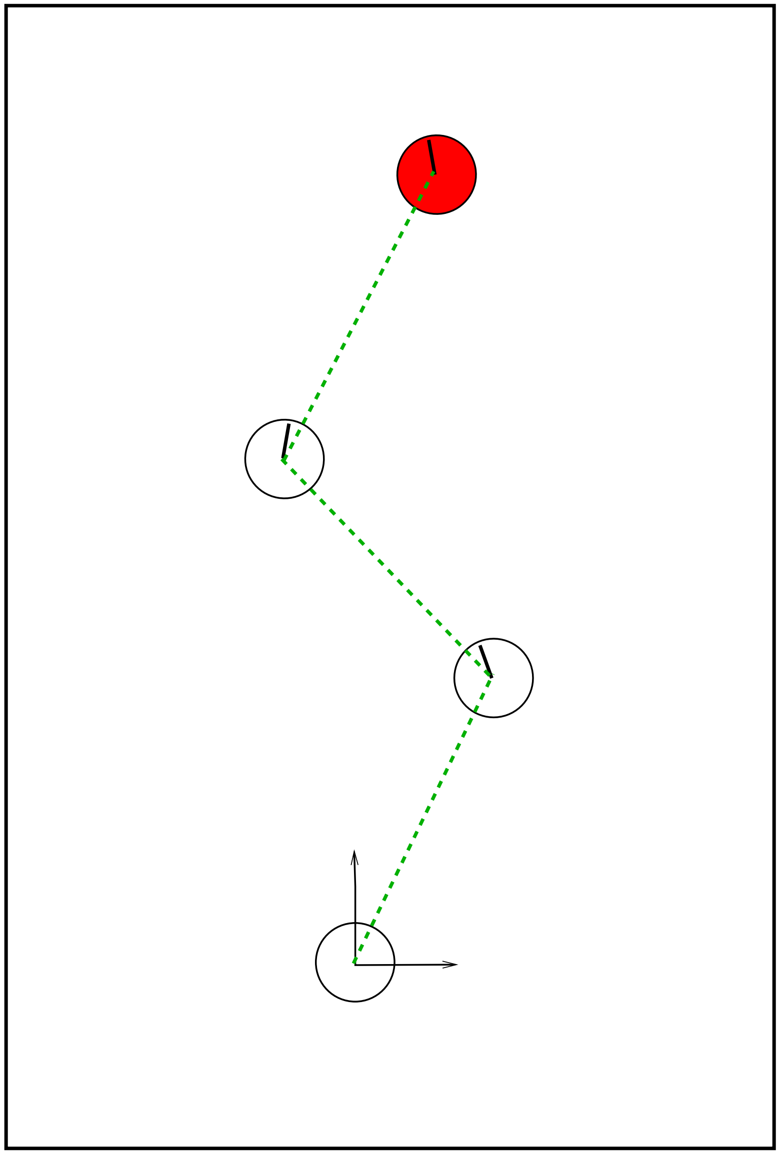
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



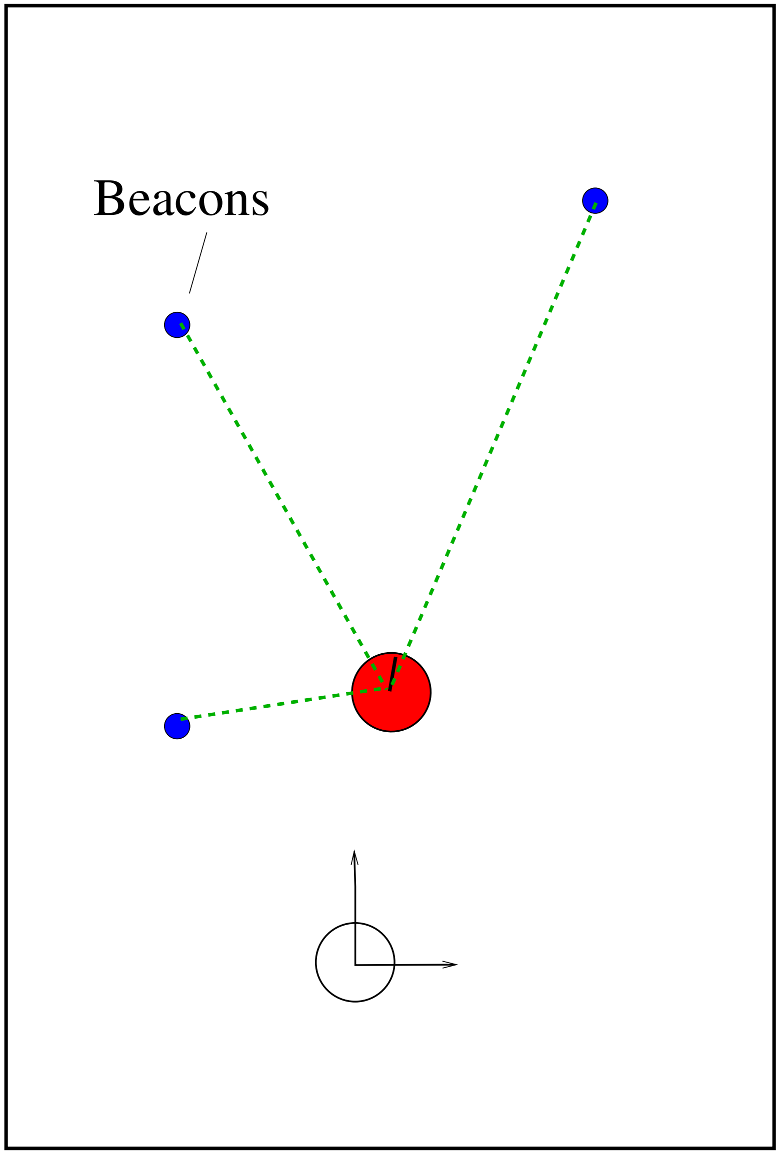
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



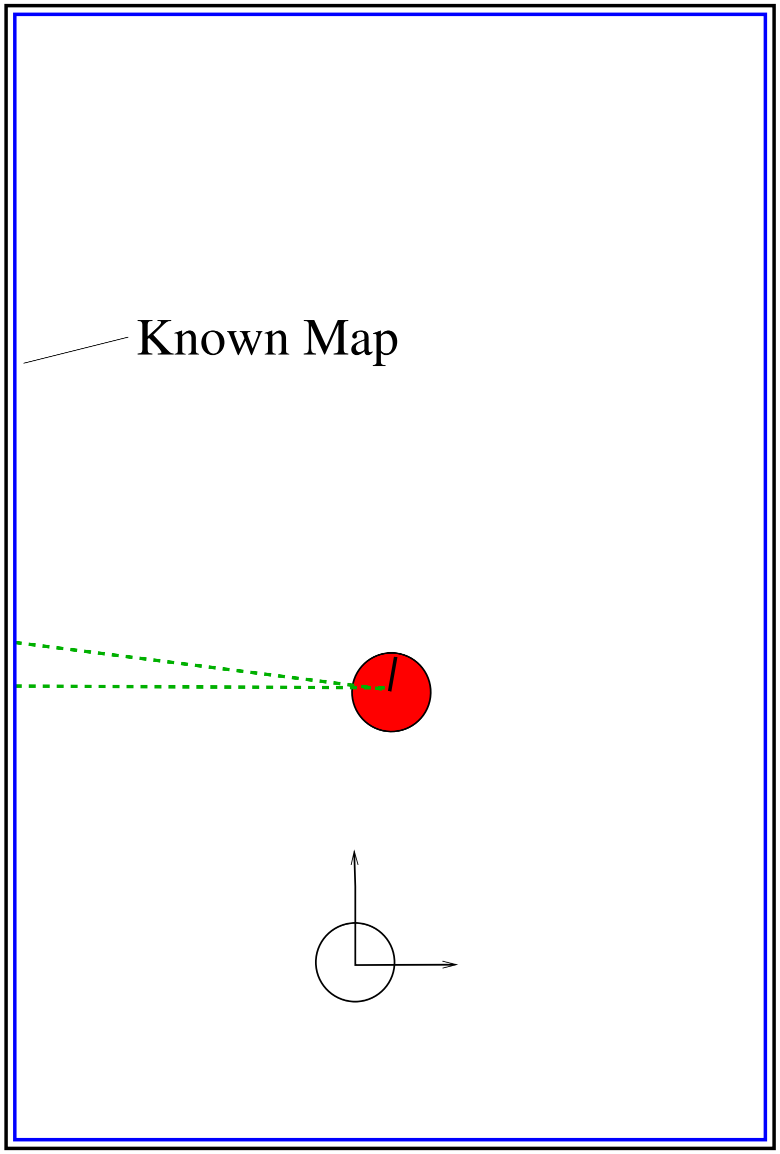
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



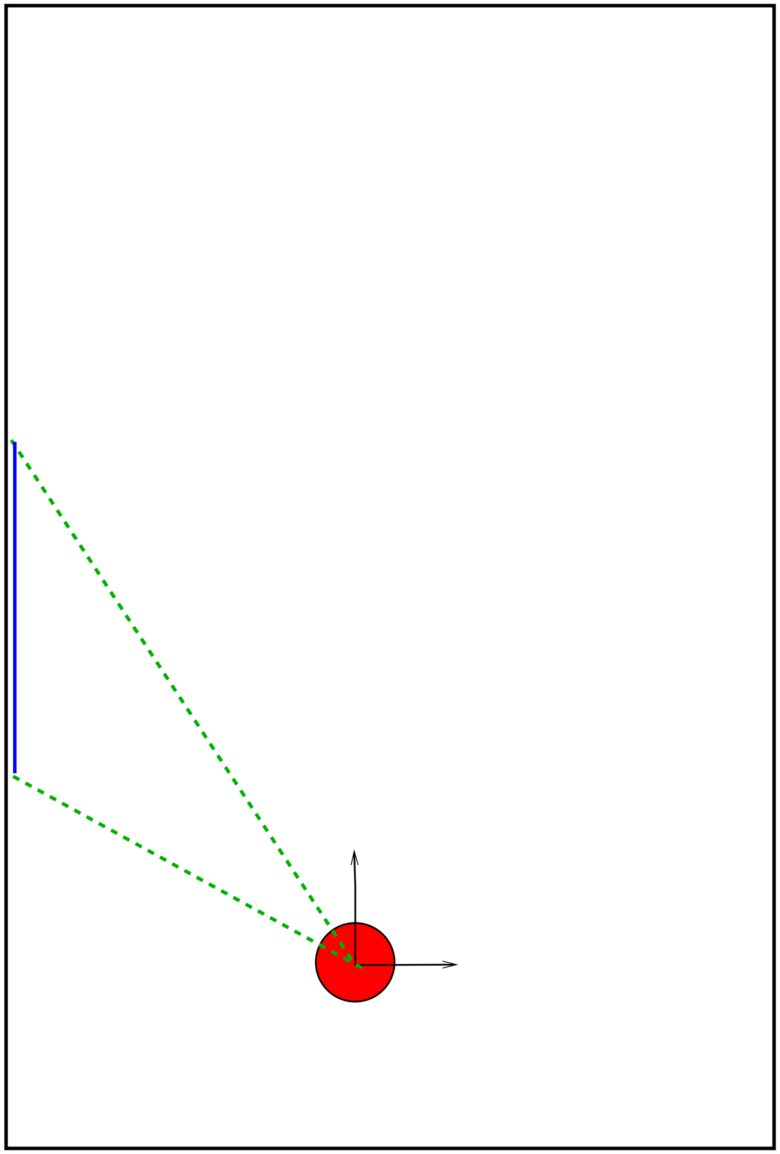
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

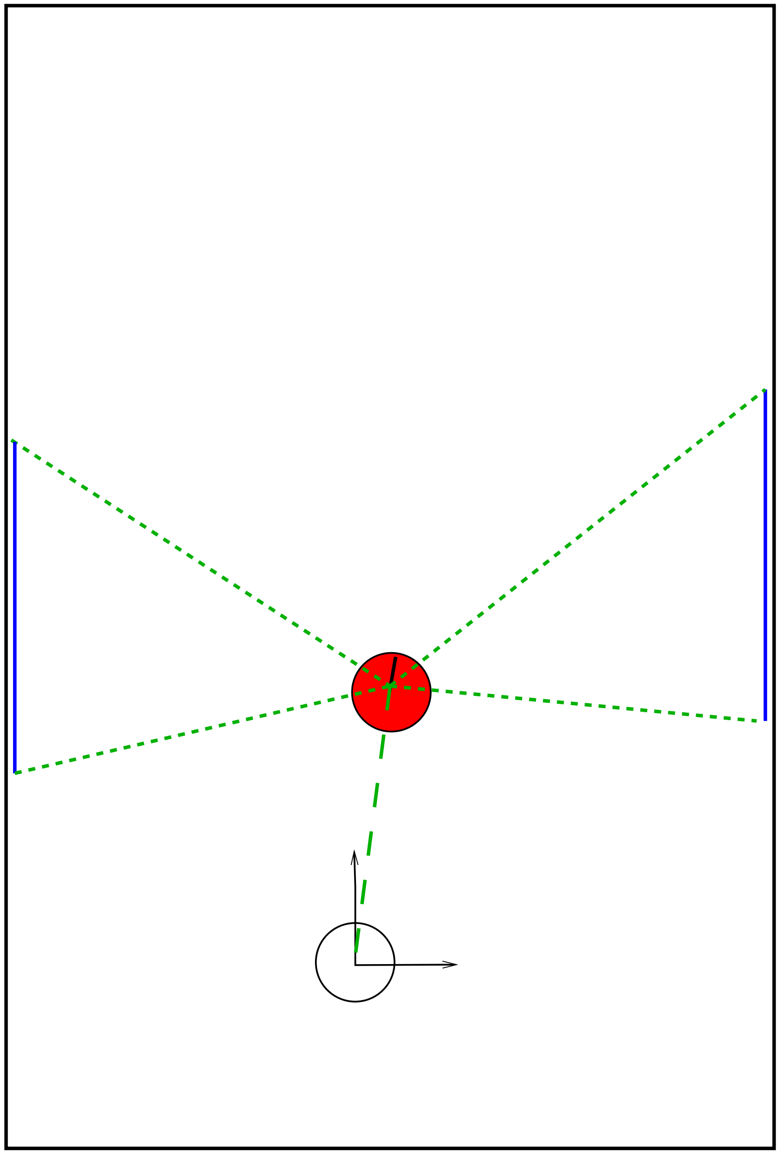
- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping



Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]

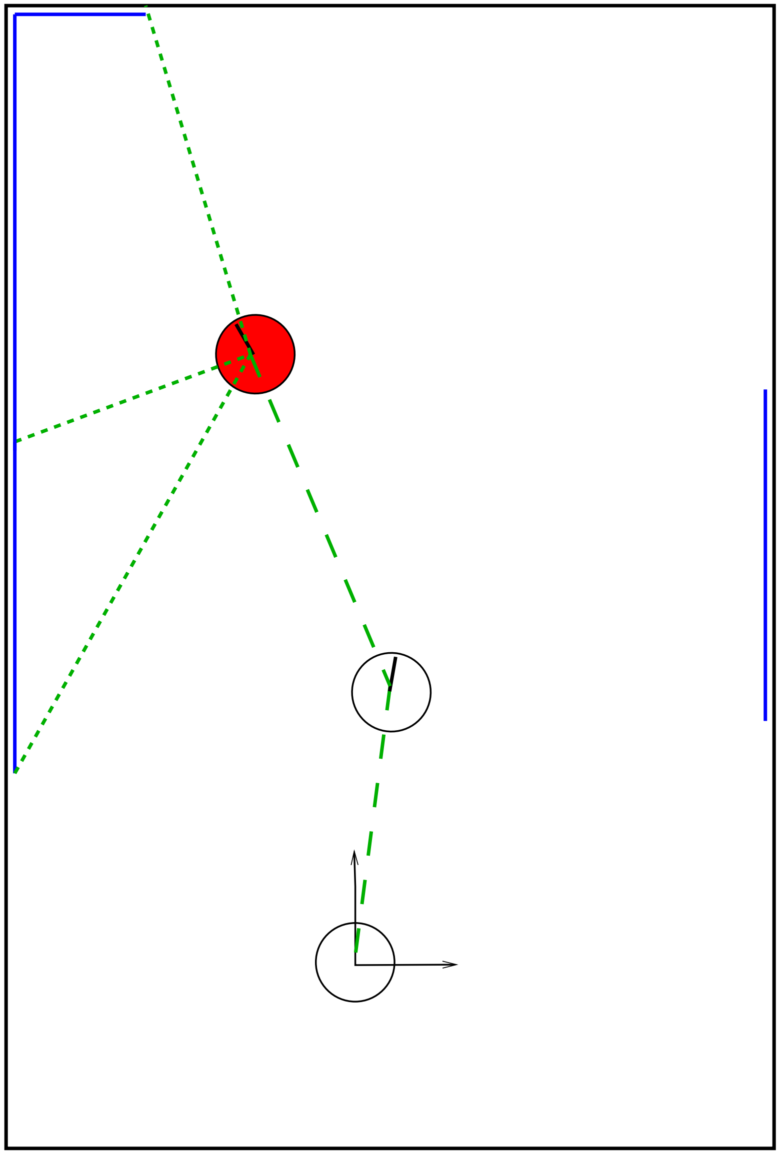
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



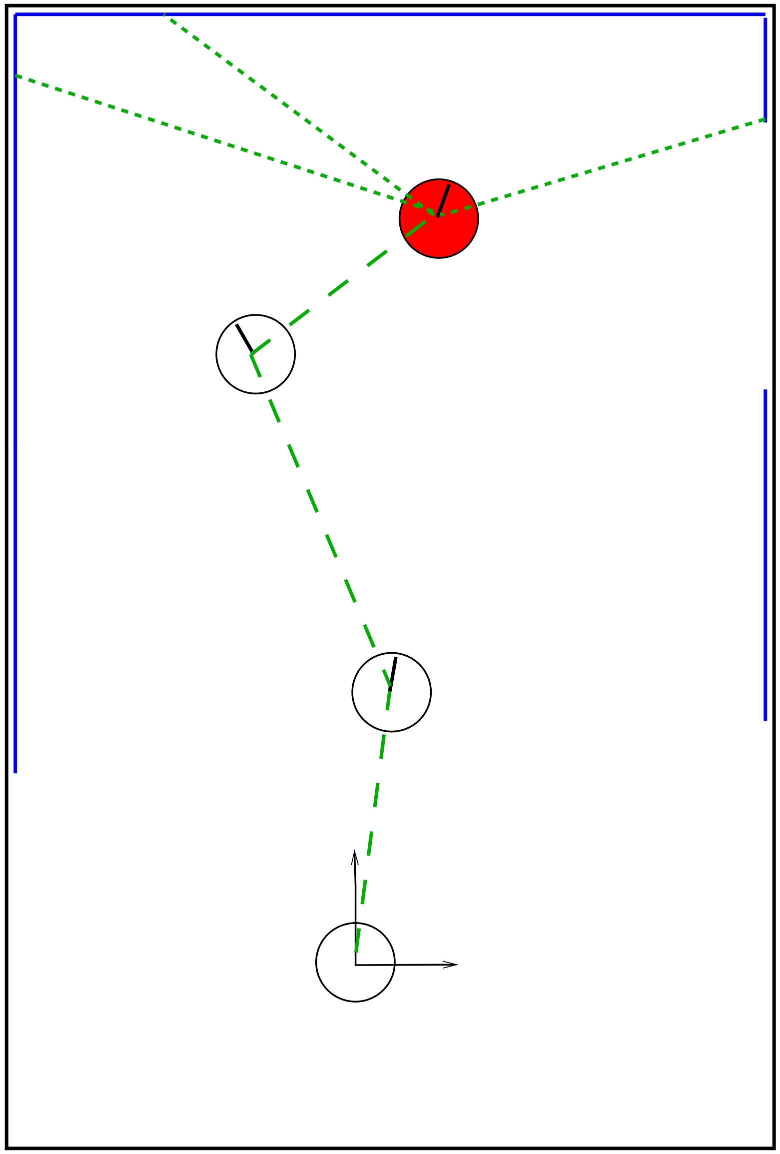
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



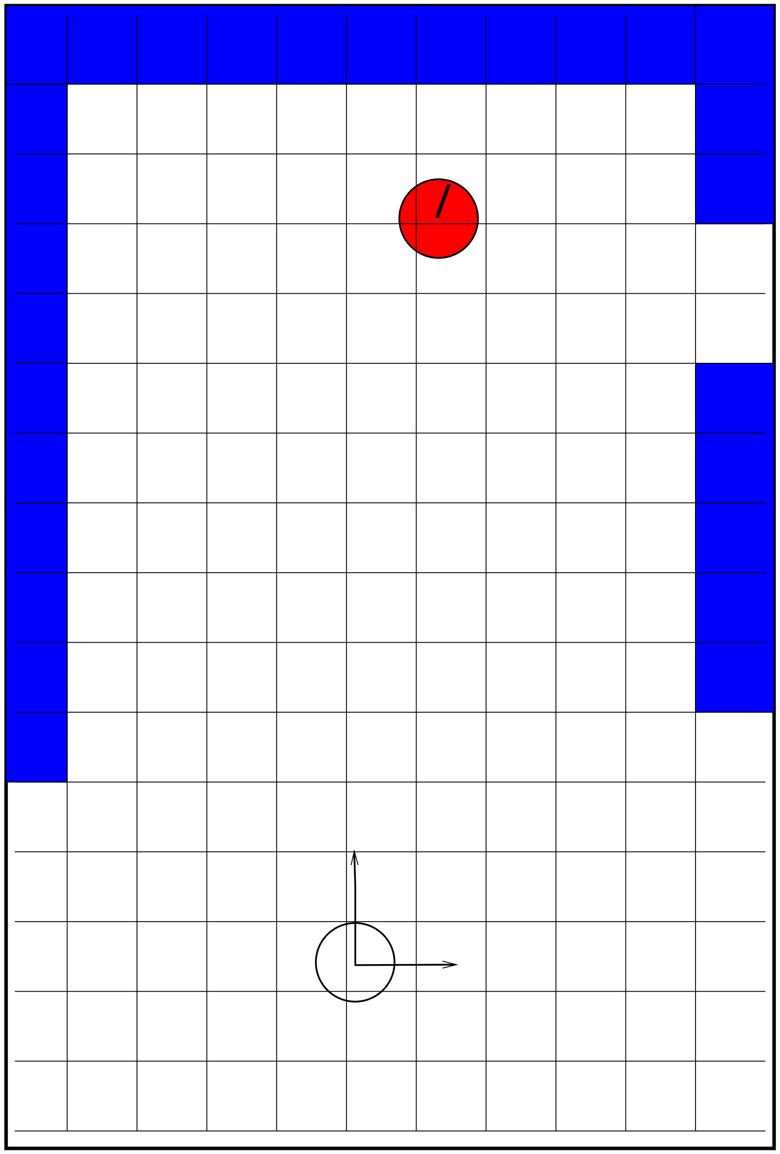
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



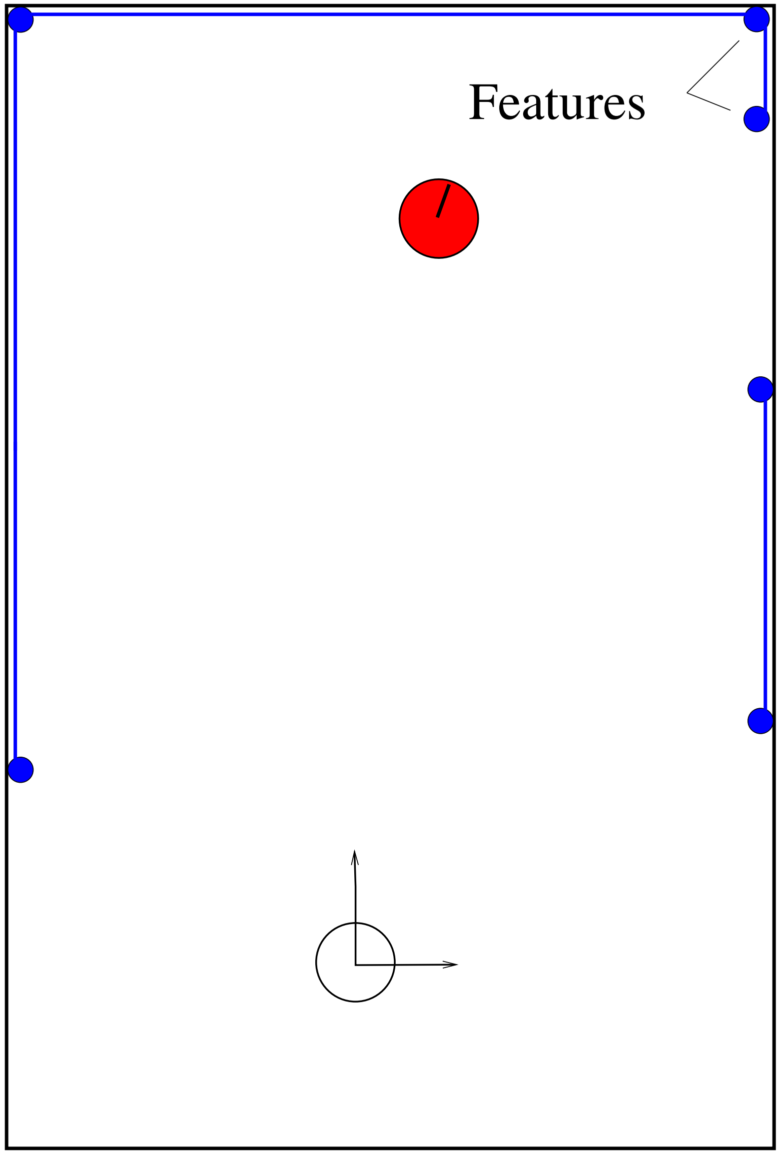
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



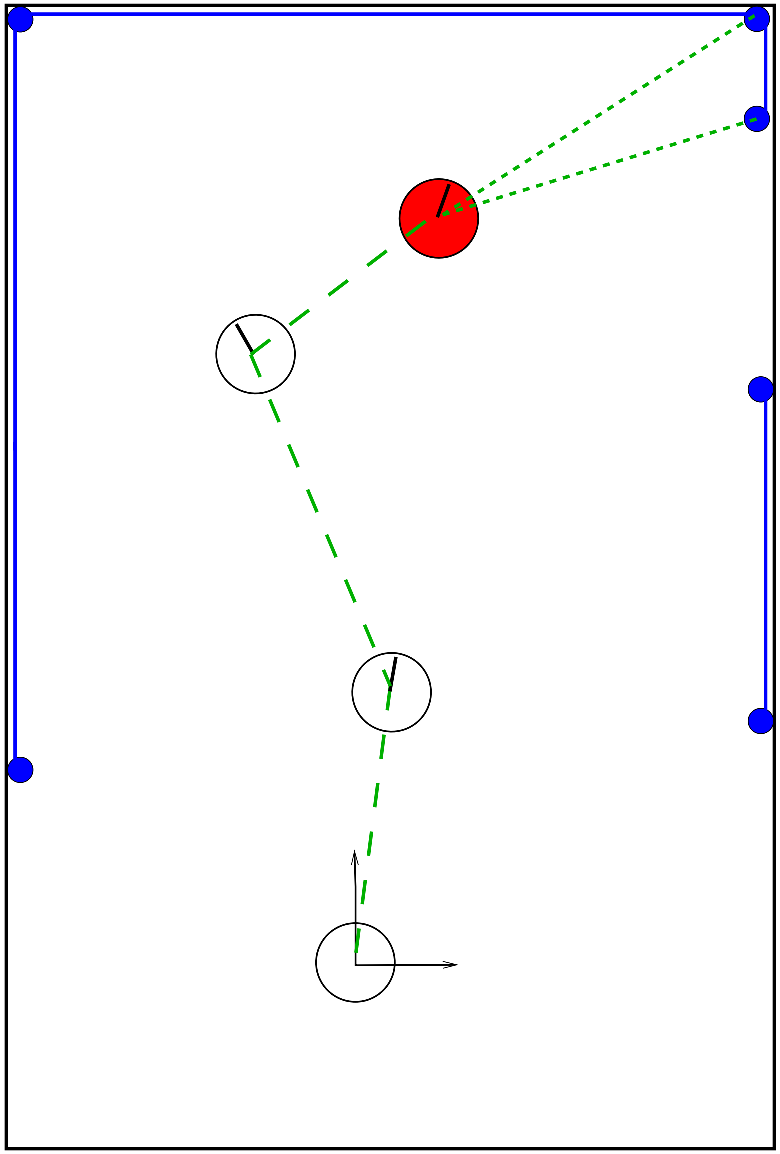
Sensor Based Localization and Mapping

Localization Methods

- Dead reckoning [Lu&Milios]
- Beacon based localization (GPS)
- Localization using known maps [Borenstein]
- Localization with no prior knowledge of environment
  - Requires sensor based mapping

Mapping Methods

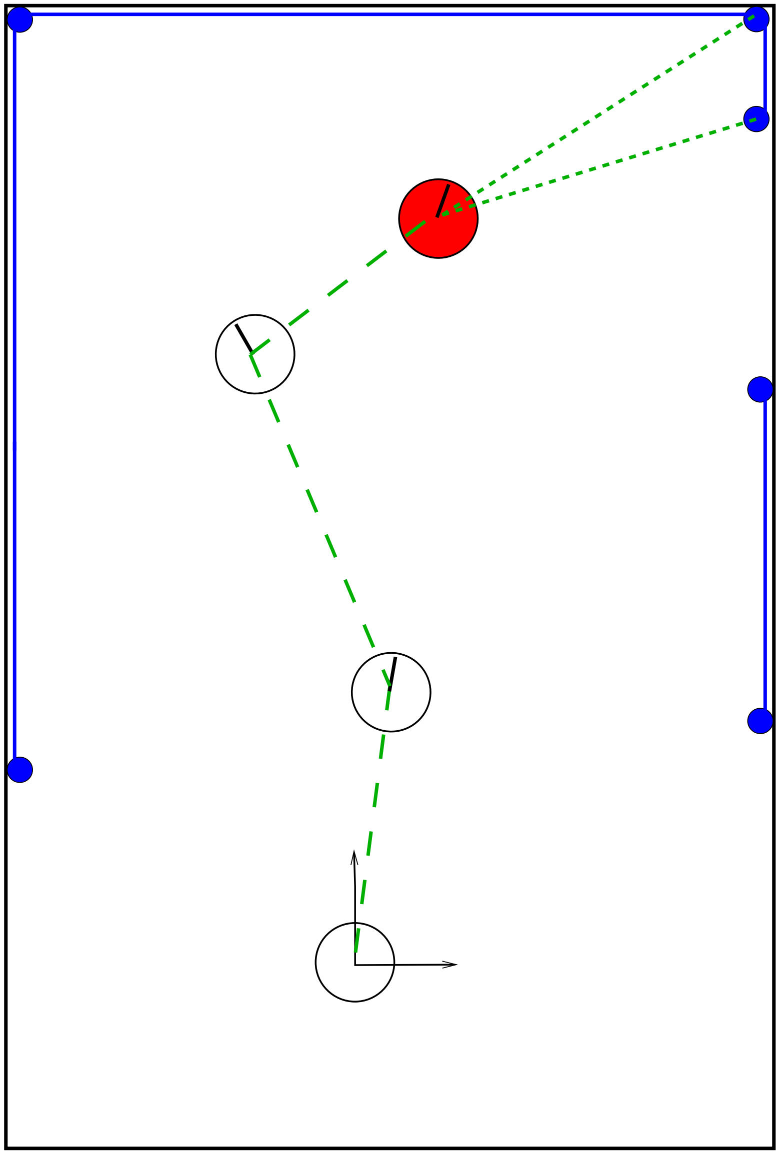
- Grid based mapping - [Elfes]
- Feature based mapping - [Chatila & Laumond]



Sensor Based Localization and Mapping

Critical Goals

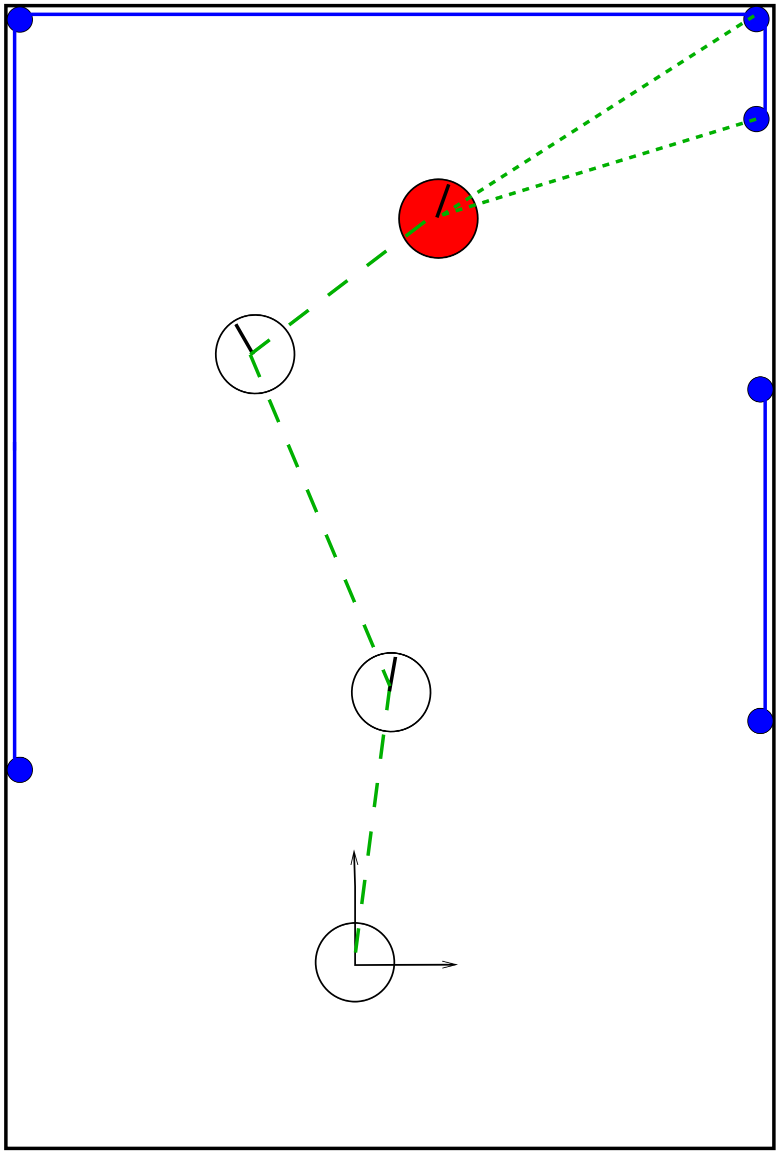
- Accurate estimates of
  - robot position
  - map feature position
  - measurement uncertainty
- Robustness for long term operation
- Computational efficiency



Sensor Based Localization and Mapping

Critical Challenges

- Data association accuracy and efficiency
  - Feature correspondence
- Sensor noise compensation
- Unmodeled errors and effects
  - Changing environment
  - Bad data

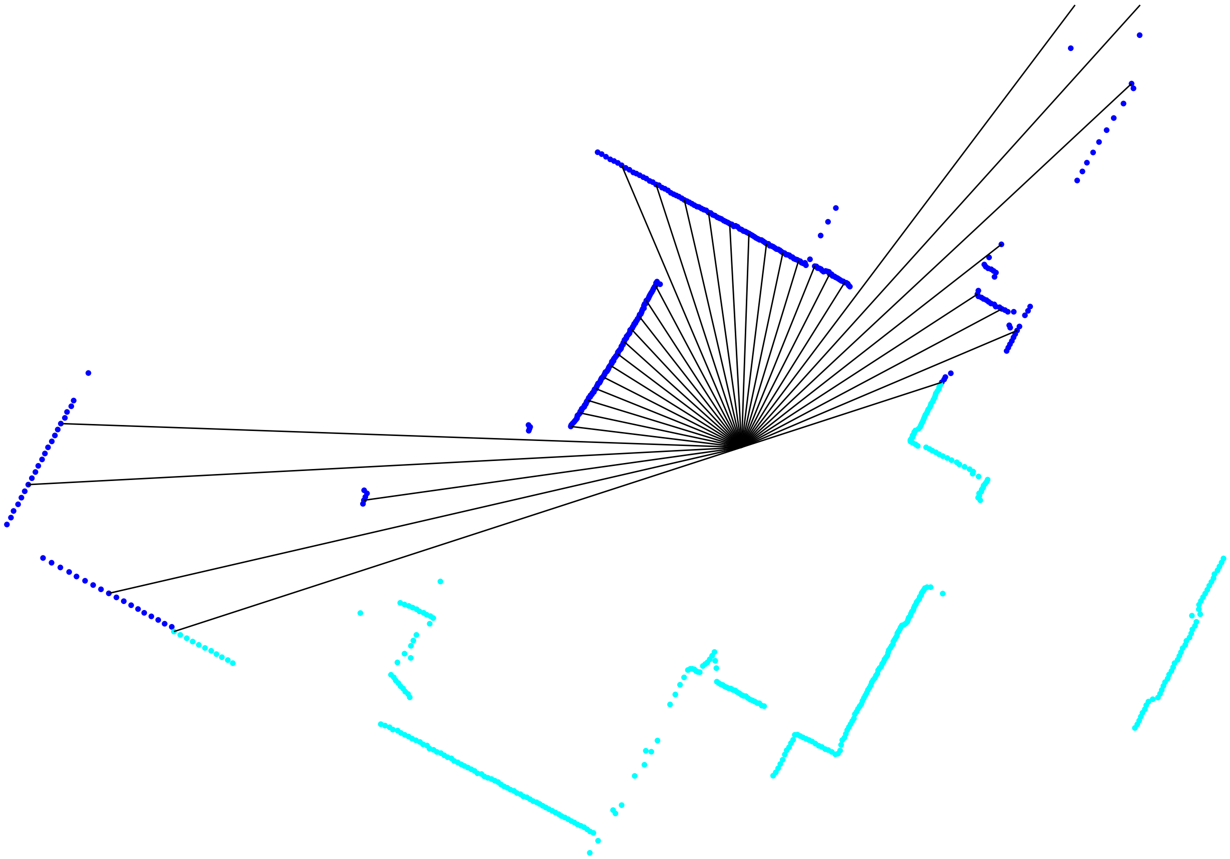


Overview

Three localization and mapping methods are presented

Assumptions: Planar robot motion in SE(2)

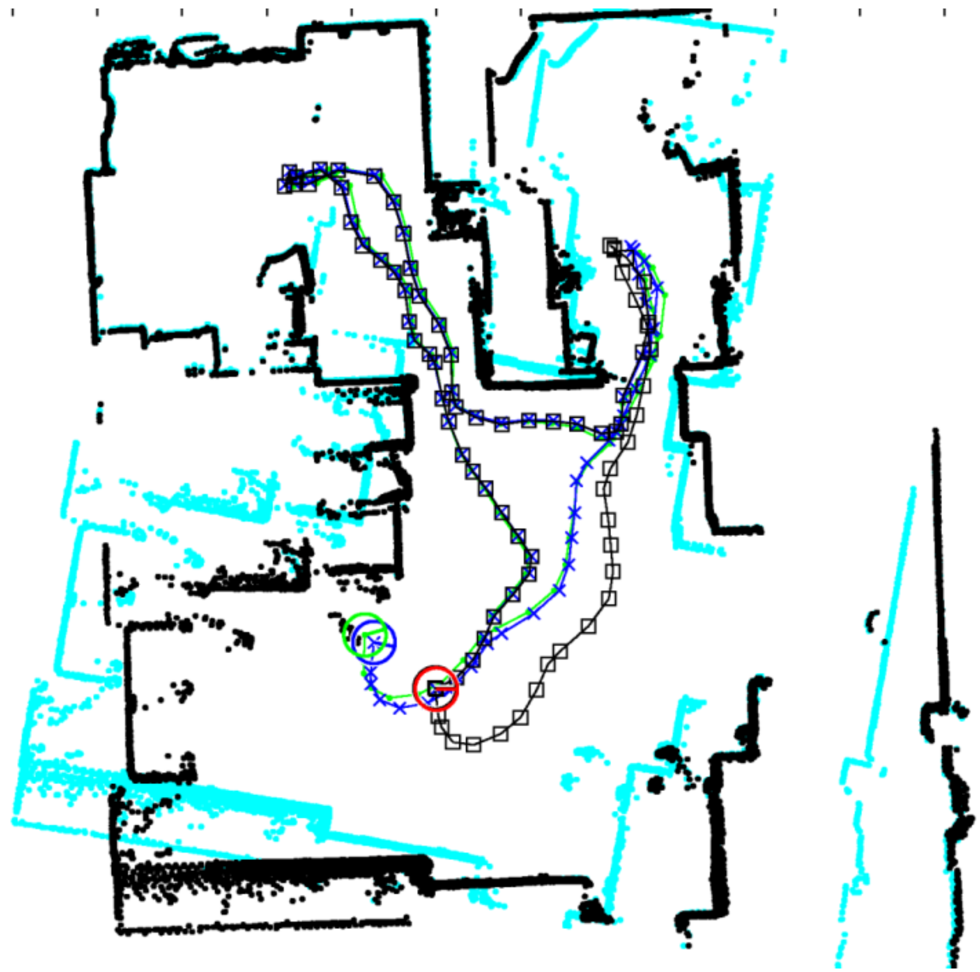
Sensors: Dense planar range scanner, Simple odometry



Overview

Three localization and mapping methods are presented

1. Range point based dead reckoning: scan matching
2. Line feature based mapping and global localization
3. Multi-scale feature based mapping and global localization



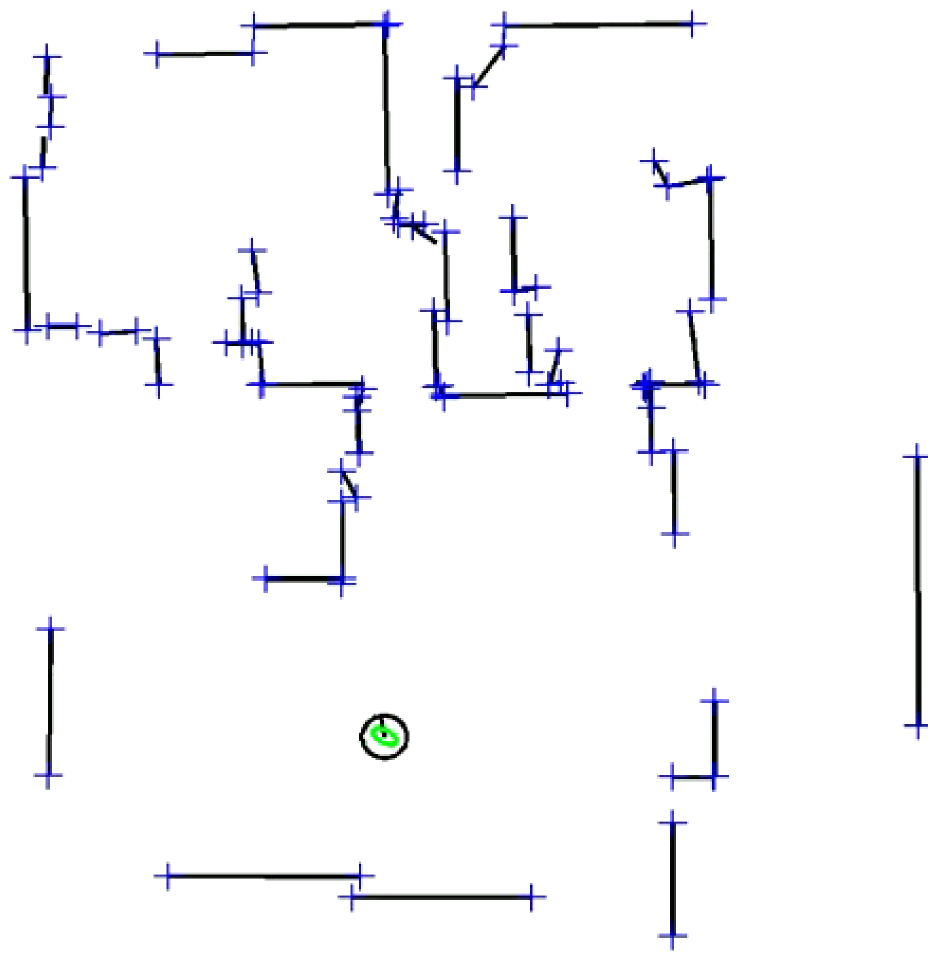
Overview

Three localization and mapping methods are presented

1. Range point based dead reckoning: scan matching

2. Line feature based mapping and global localization

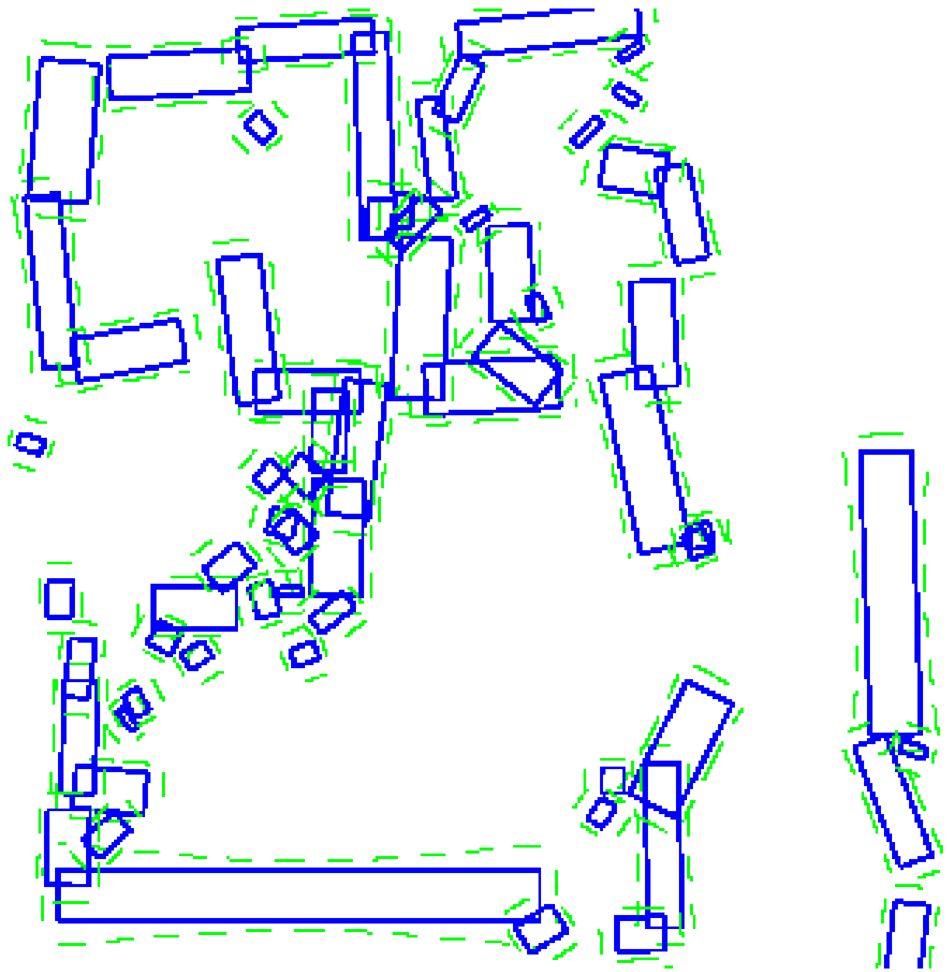
3. Multi-scale feature based mapping and global localization



Overview

Three localization and mapping methods are presented

1. Range point based dead reckoning: scan matching
2. Line feature based mapping and global localization
3. Multi-scale feature based mapping and global localization



Overview

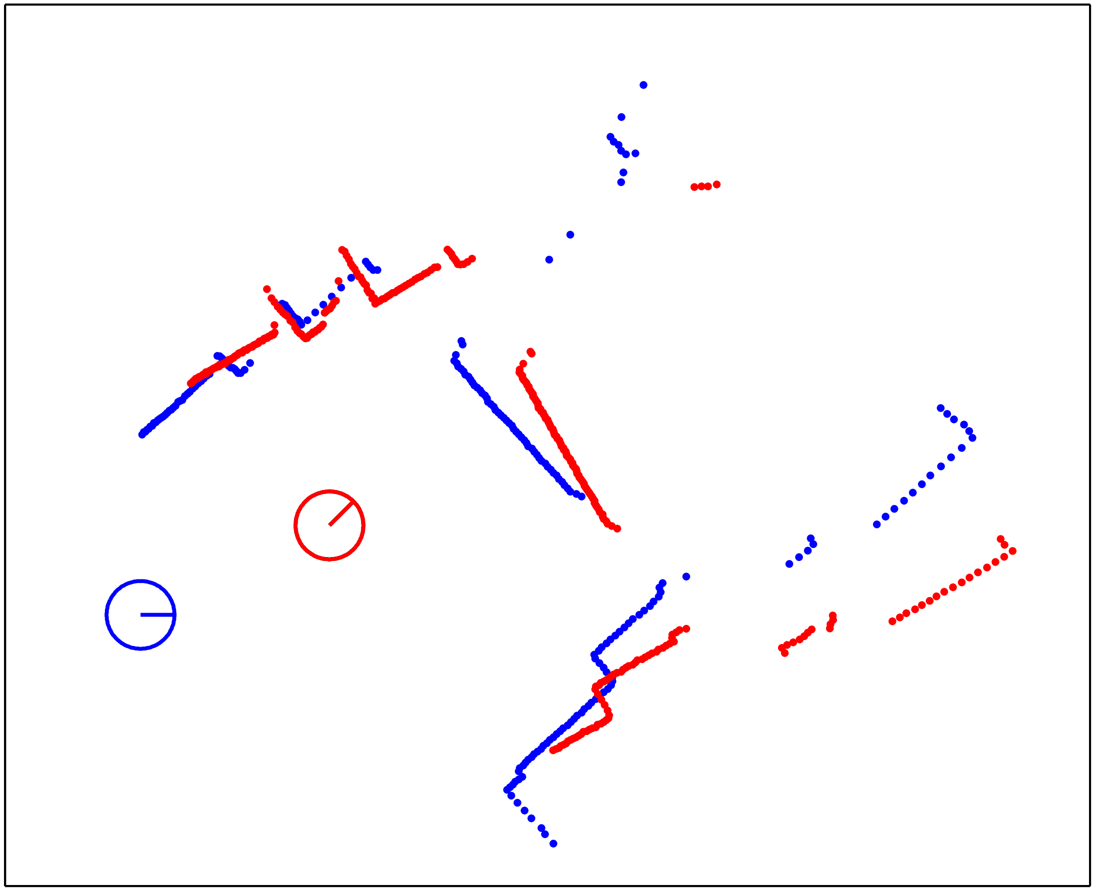
Three localization and mapping methods are presented

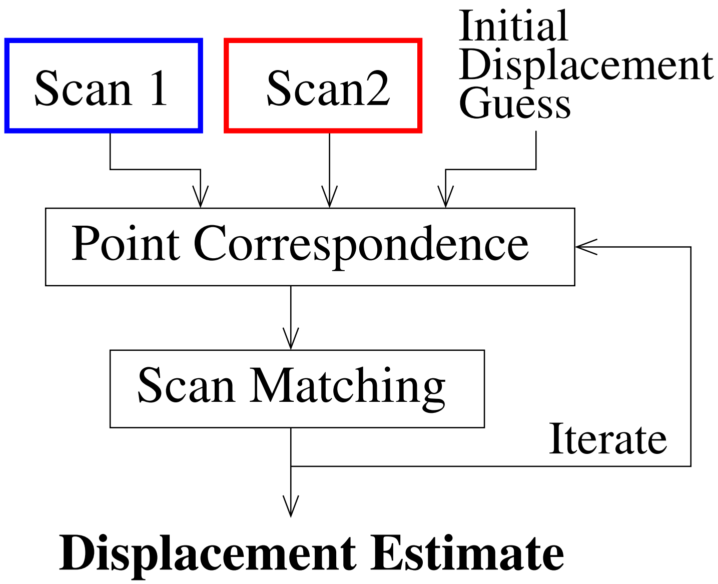
1. Range point based dead reckoning: scan matching
2. Line feature based mapping and global localization
3. Multi-scale feature based mapping and global localization

Critical Goals

- Accurate estimates of
  - robot position
  - map feature position
  - measurement uncertainty
- Robustness for long term operation
- Computational efficiency

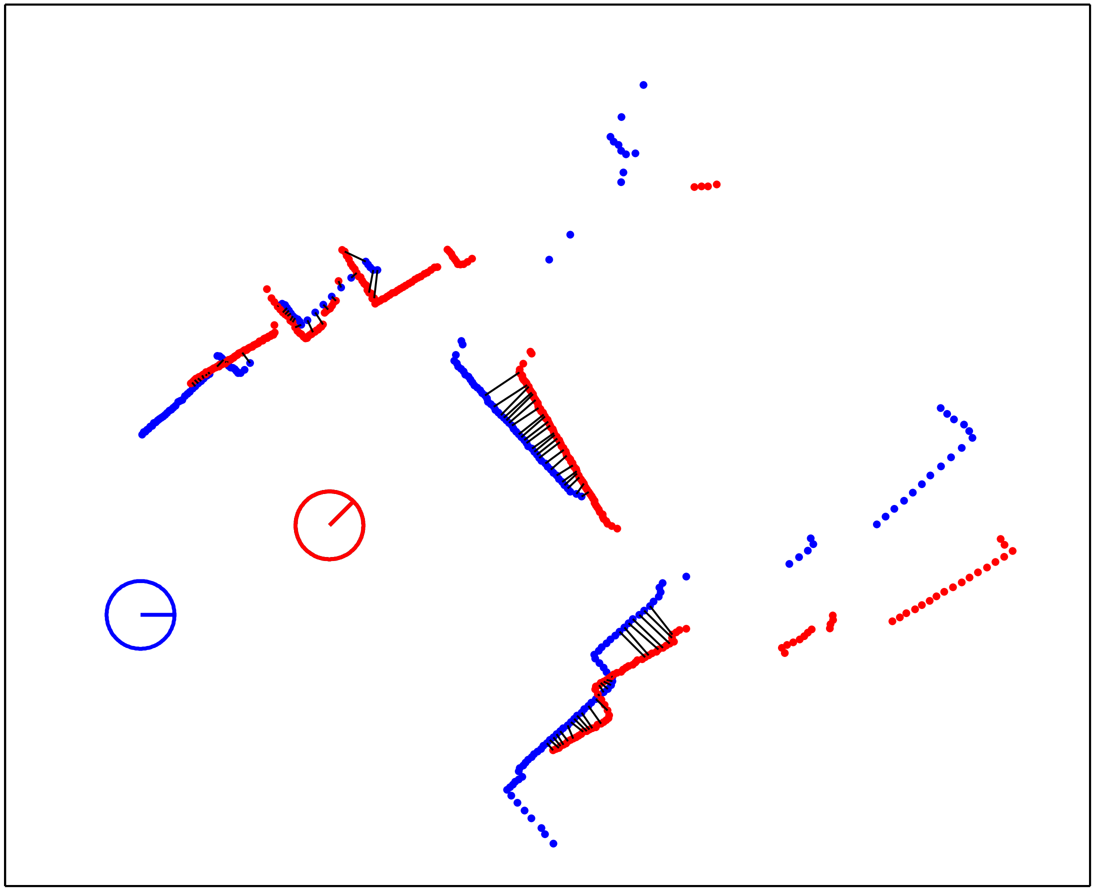
Method 1) Weighted Scan Matching

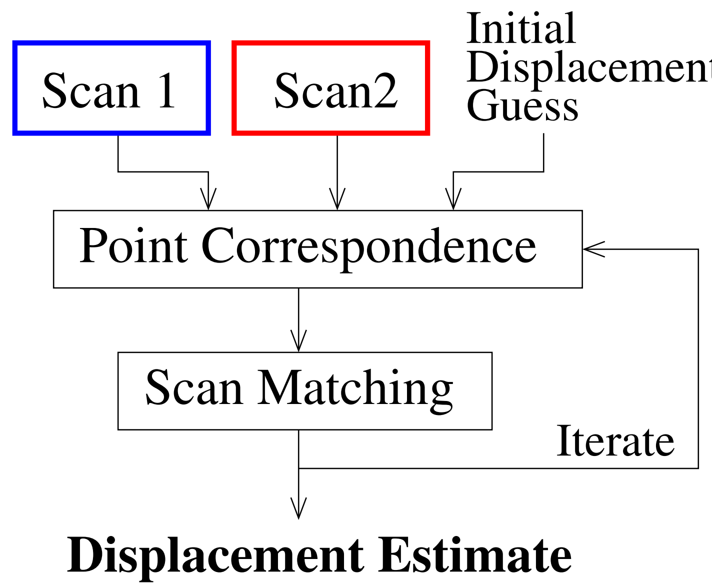




- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios '97

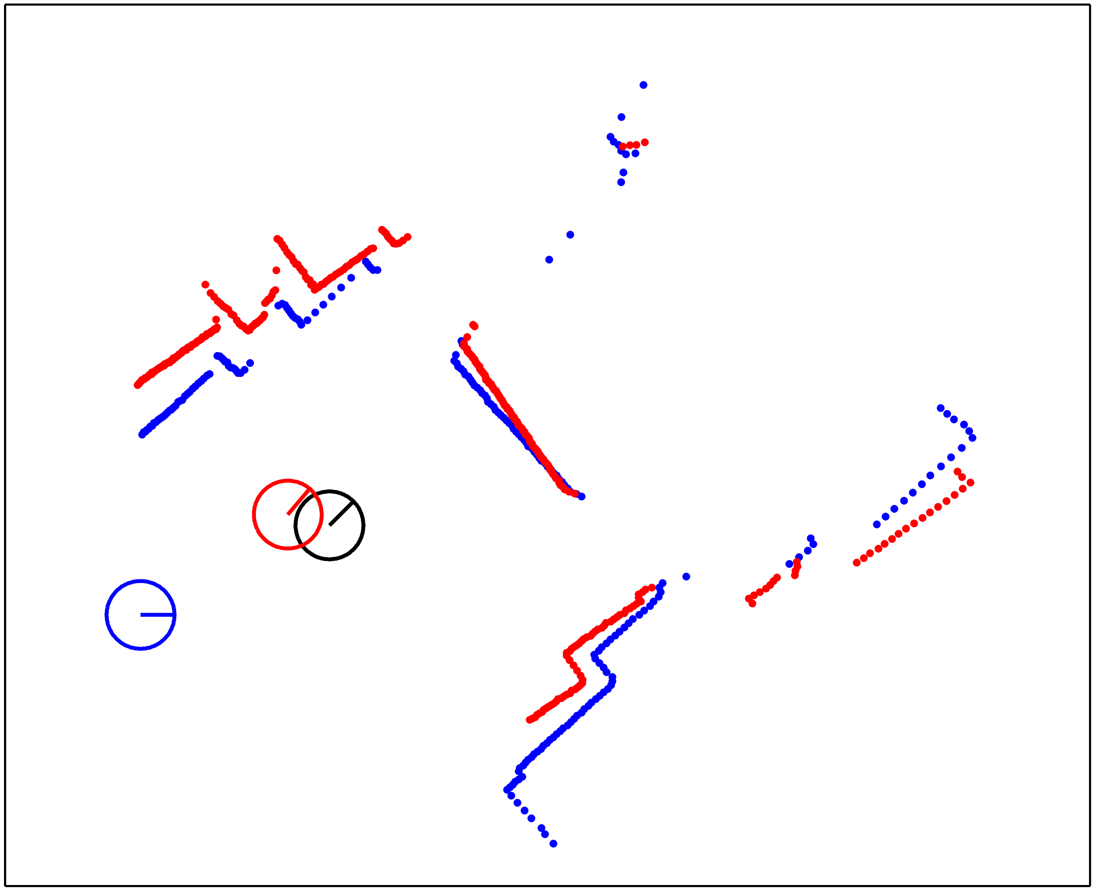
Method 1) Weighted Scan Matching

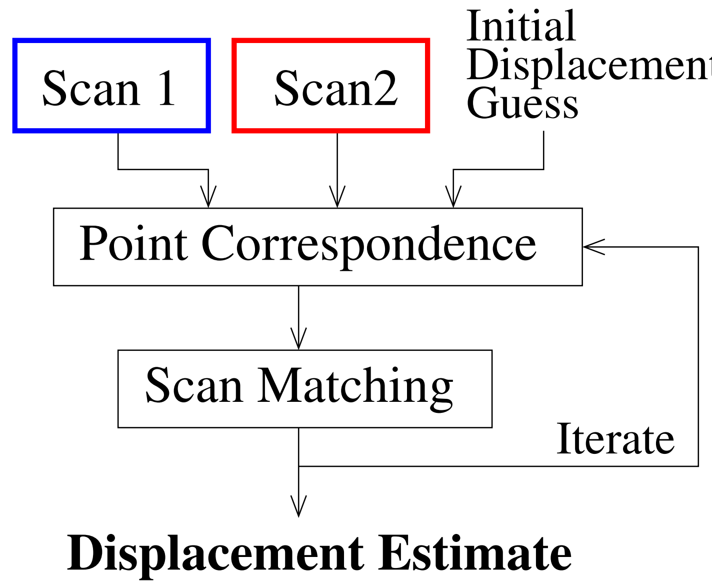




- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios '97

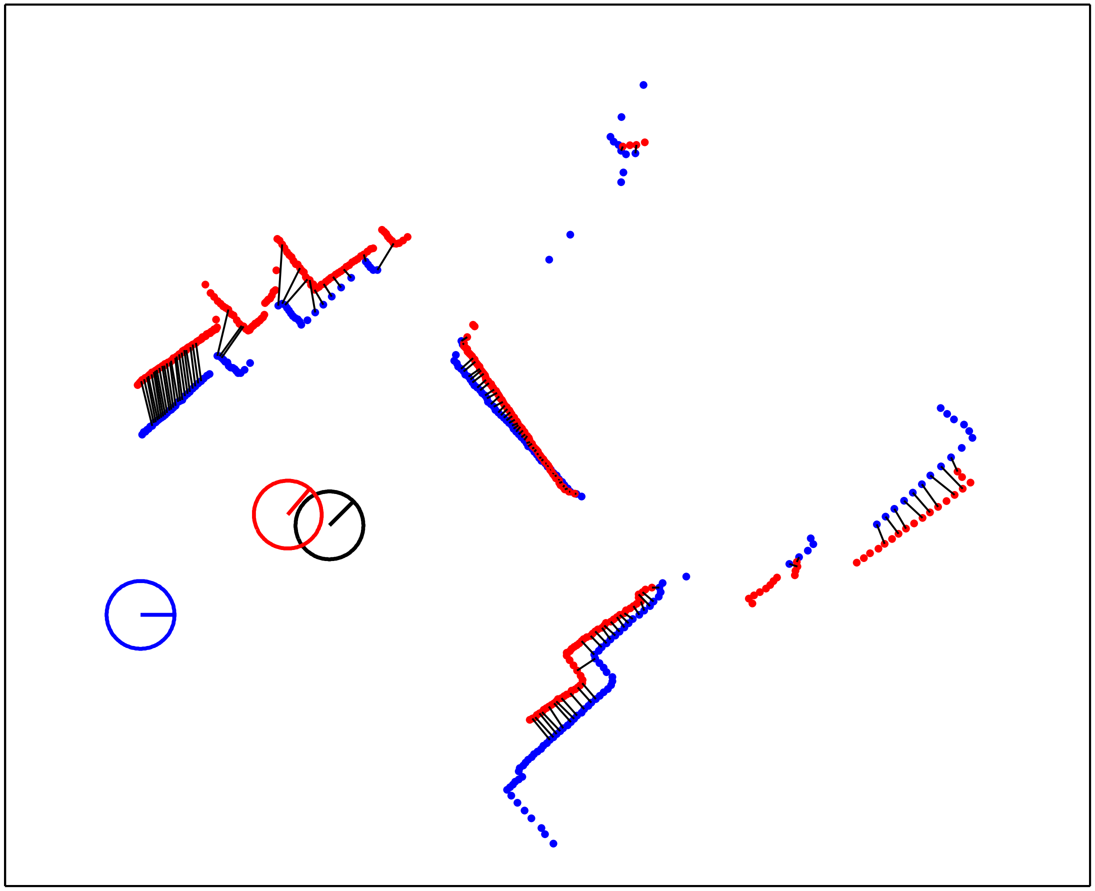
Method 1) Weighted Scan Matching

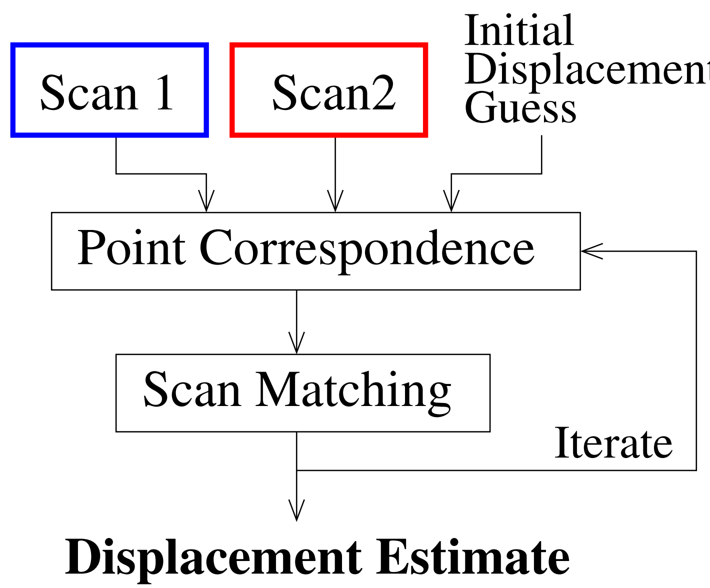




- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios '97

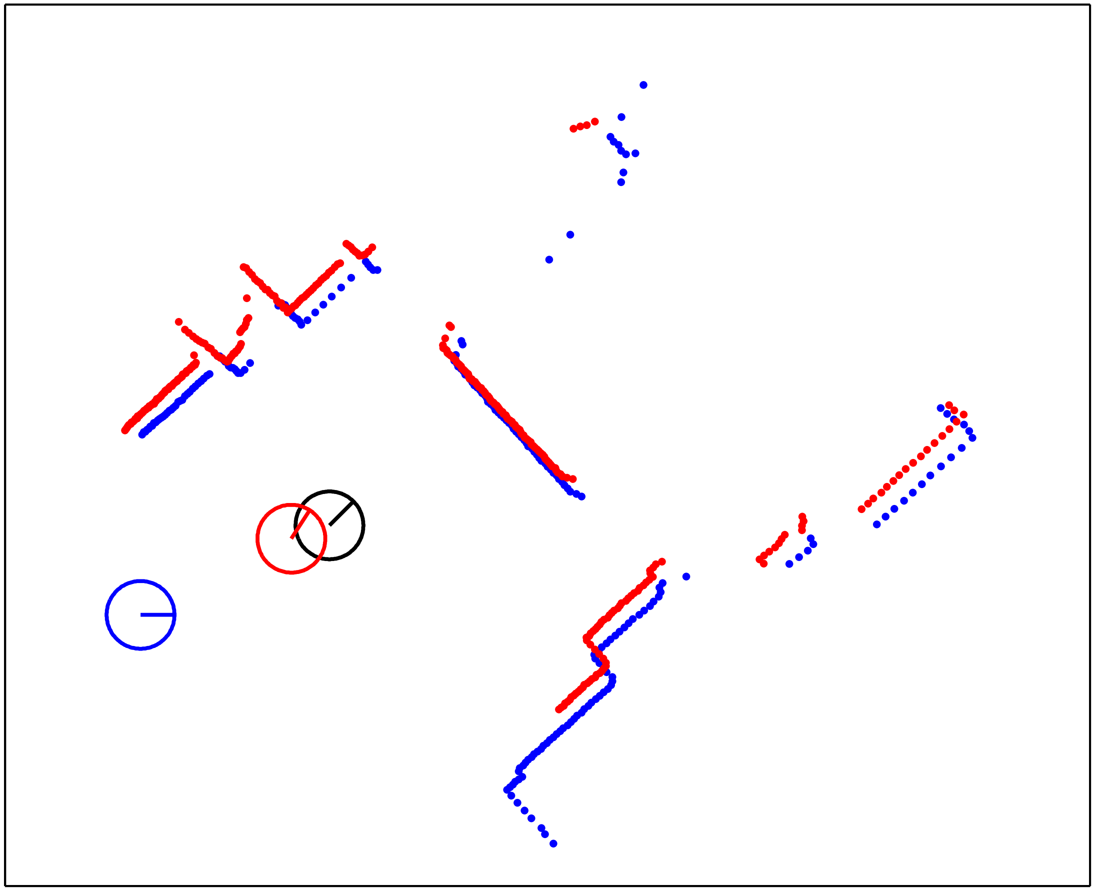
Method 1) Weighted Scan Matching

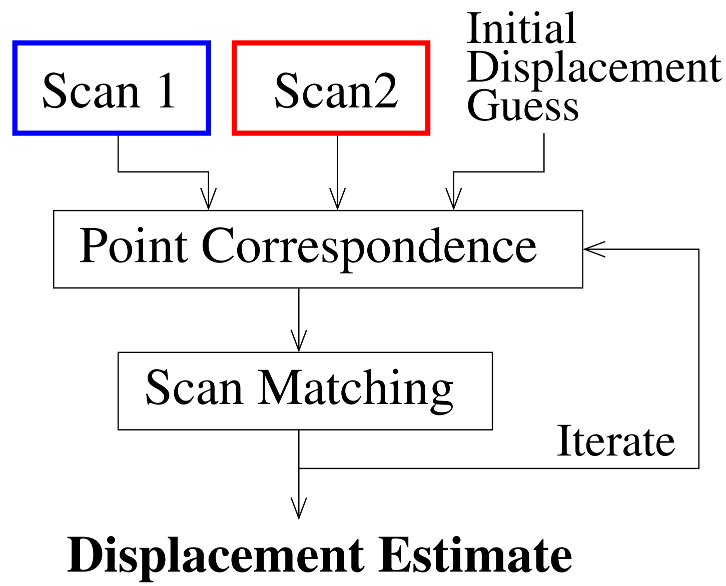




- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios ’97

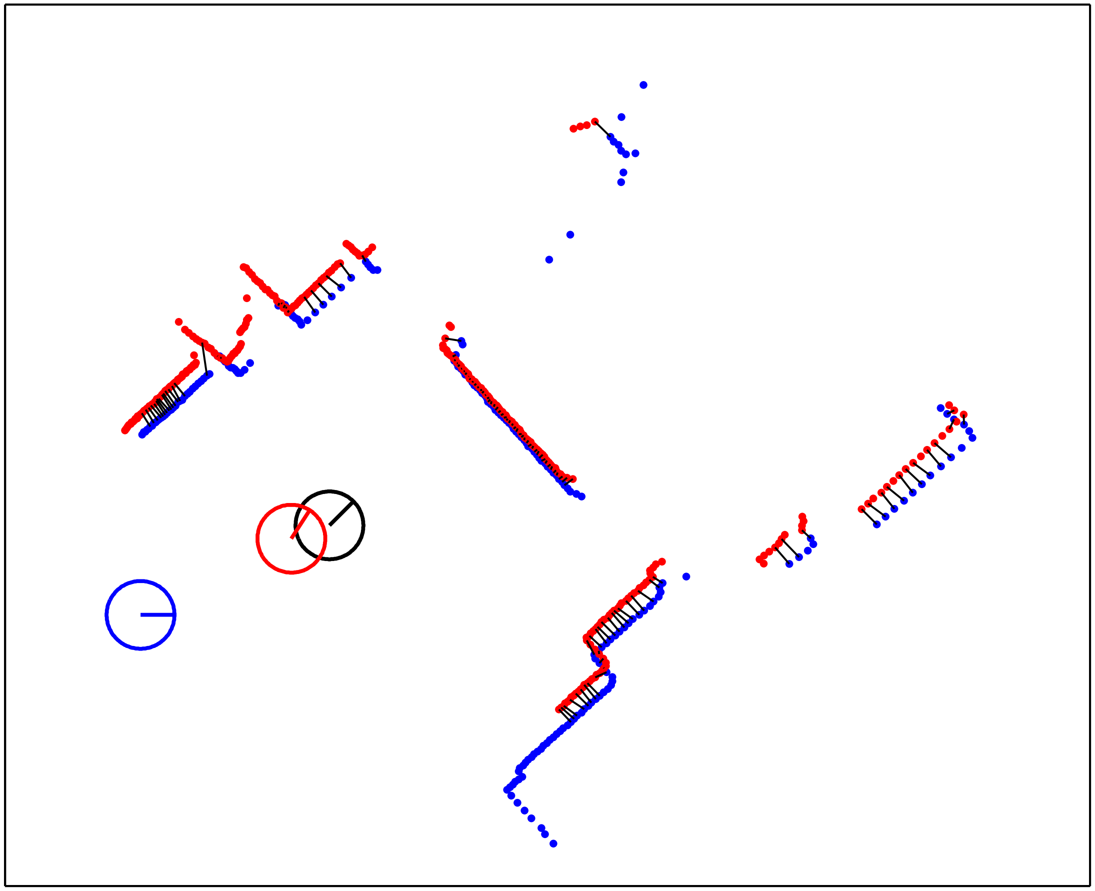
Method 1) Weighted Scan Matching

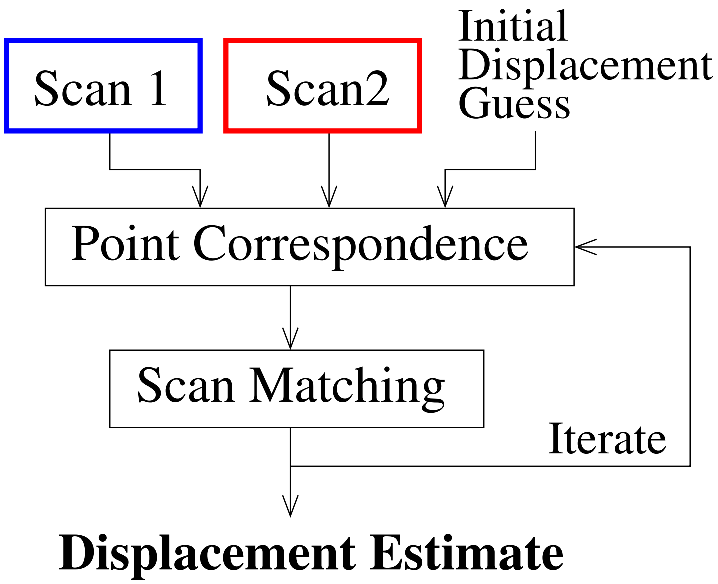




- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios ’97

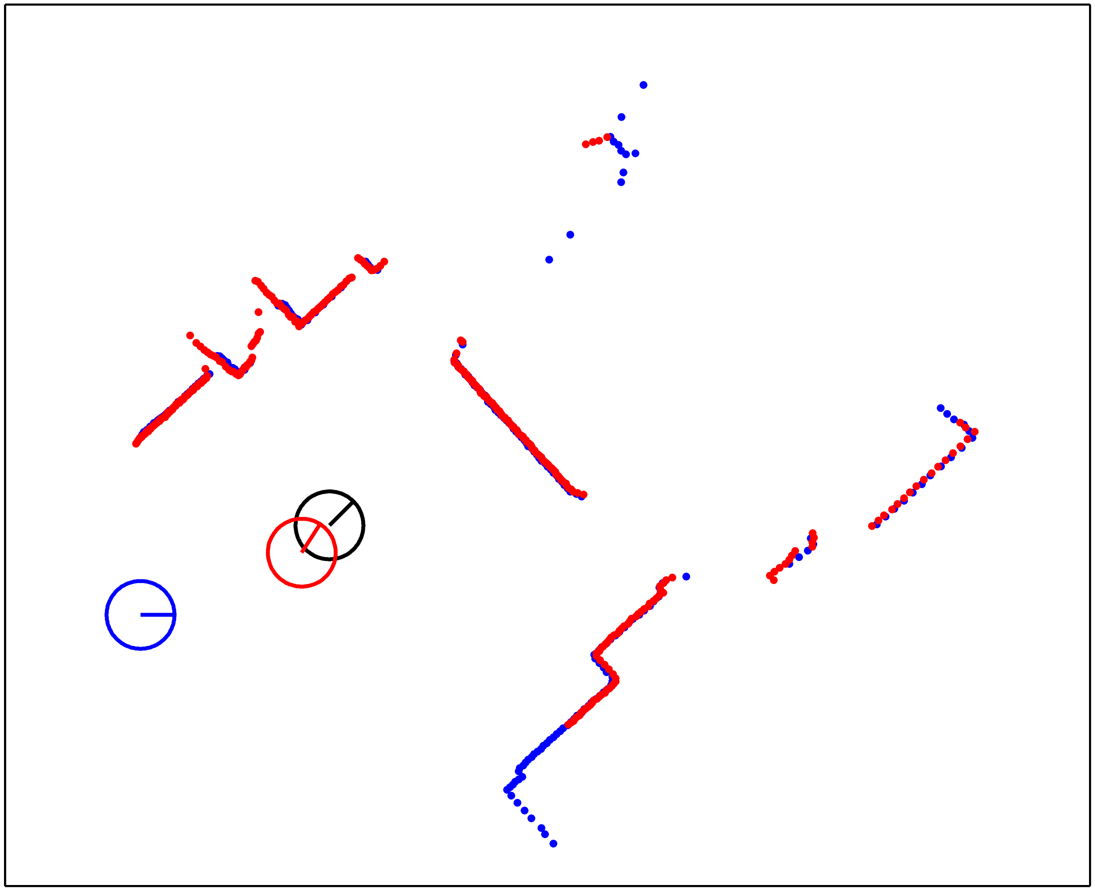
Method 1) Weighted Scan Matching

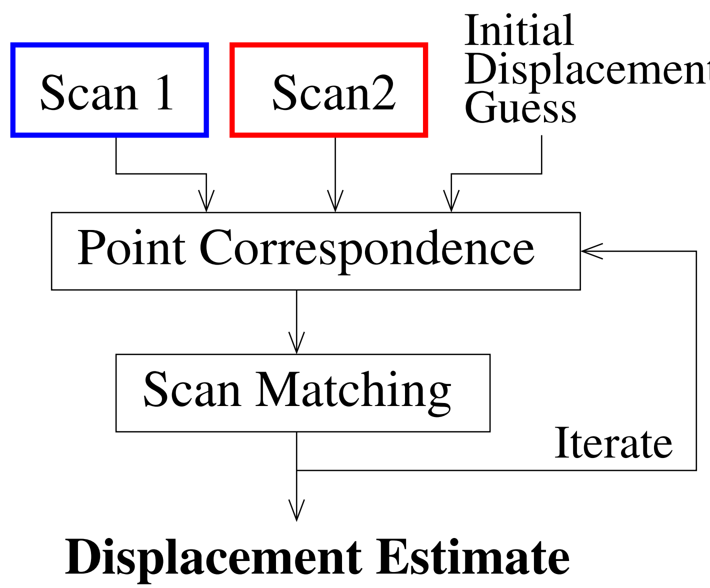




- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios '97

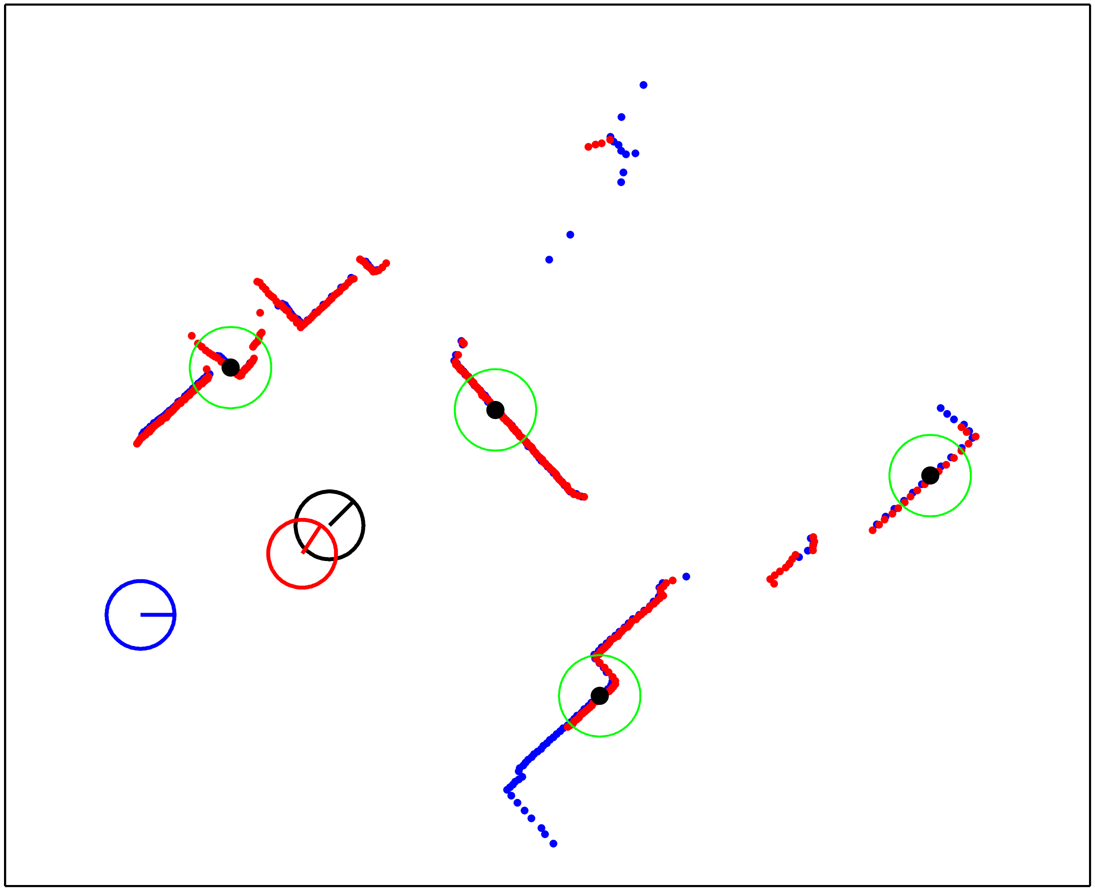
Method 1) Weighted Scan Matching

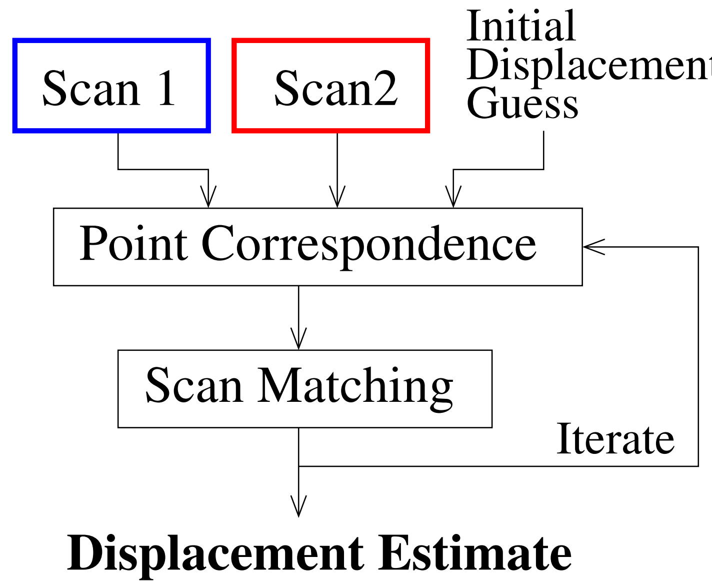




- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios ’97

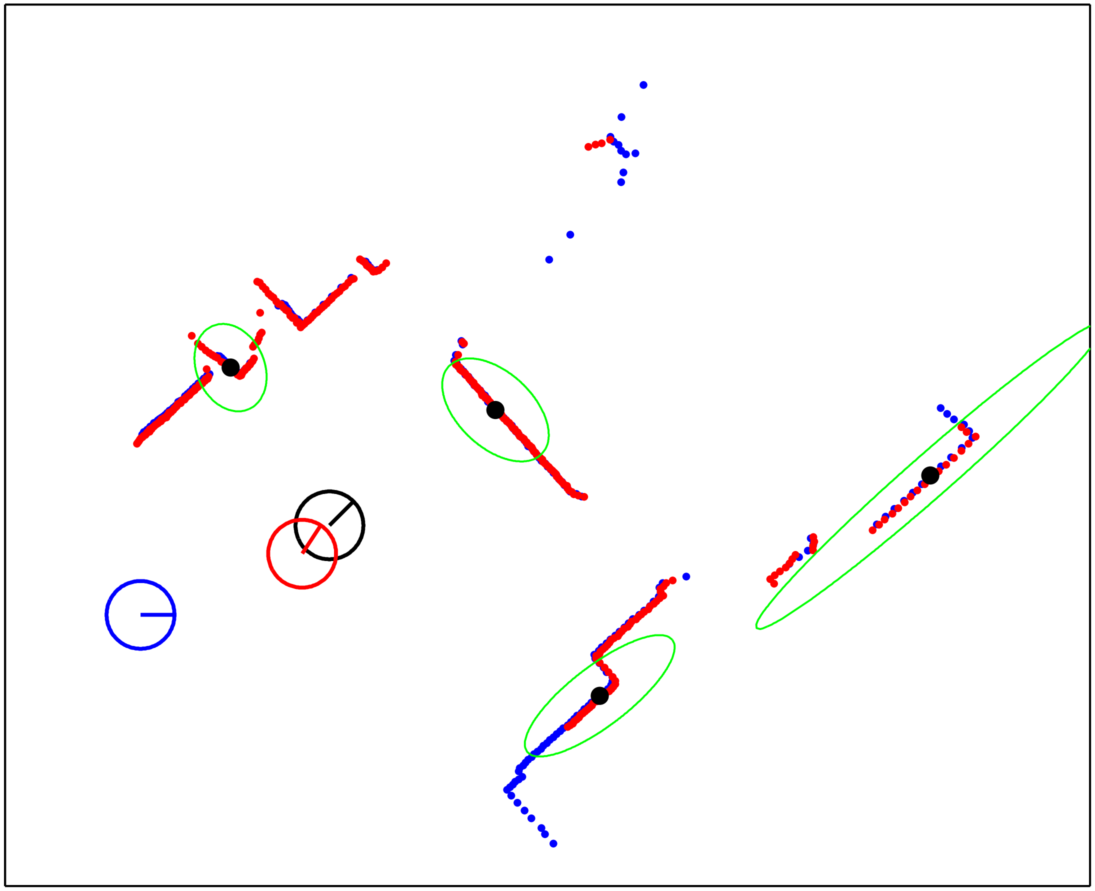
Method 1) Weighted Scan Matching

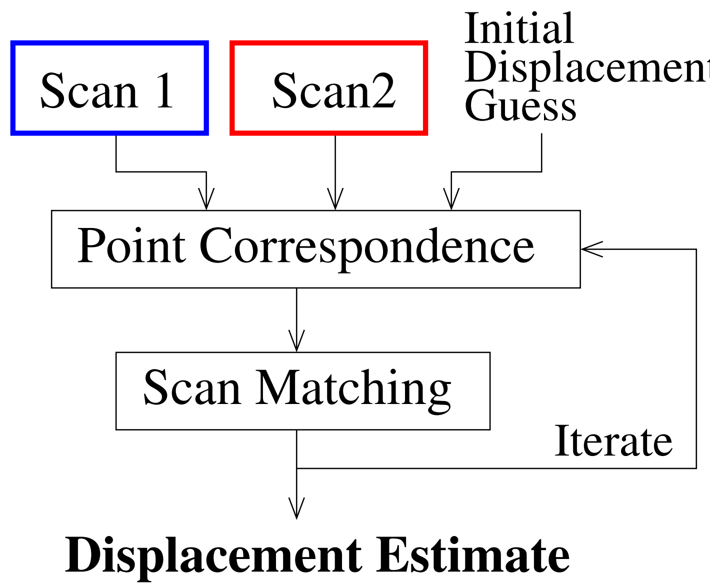




- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios ’97

Method 1) Weighted Scan Matching



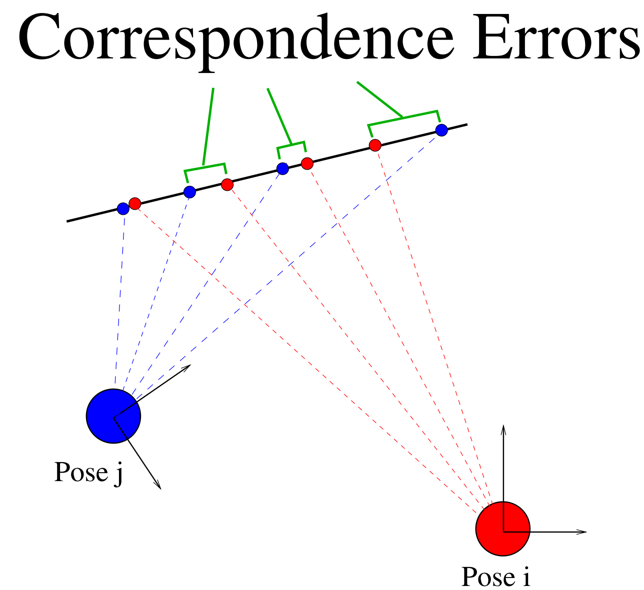


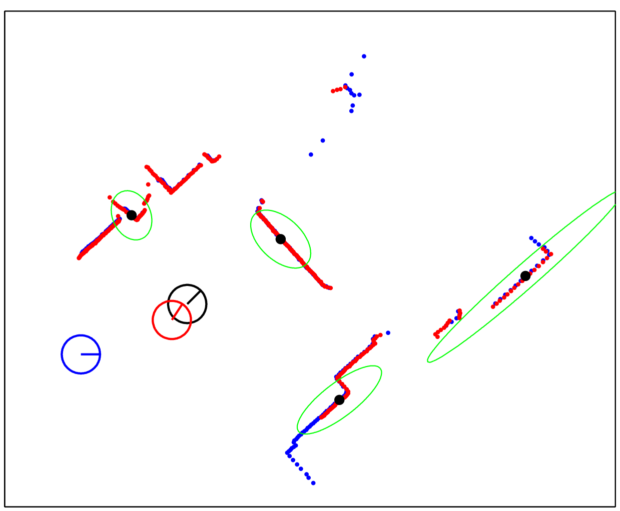
- Correlate range measurements to estimate displacement
- Can improve (or even replace) odometry - [Roumeliotis]
- Previous Work - Vision community and Lu & Milios '97

Weighted Approach

Explicit models of uncertainty
& noise sources for each point pair:

- Sensor noise & errors
  - Range noise
  - Scan angle uncertainty
- Point correspondence uncertainty
  - Due to a geometric effect





Improvement vs. unweighted method:

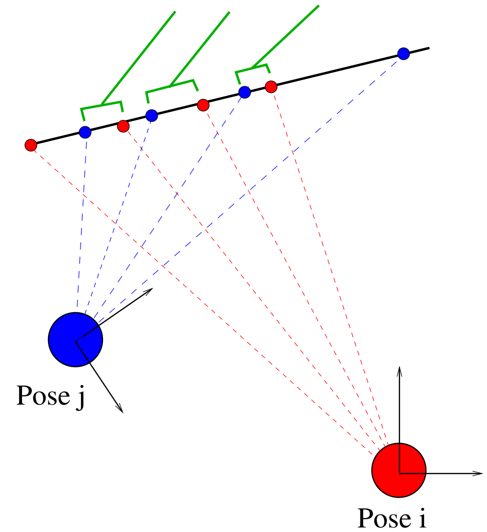
- More accurate displacement estimate
- More realistic covariance estimate
- Increased robustness to initial conditions
- Improved convergence

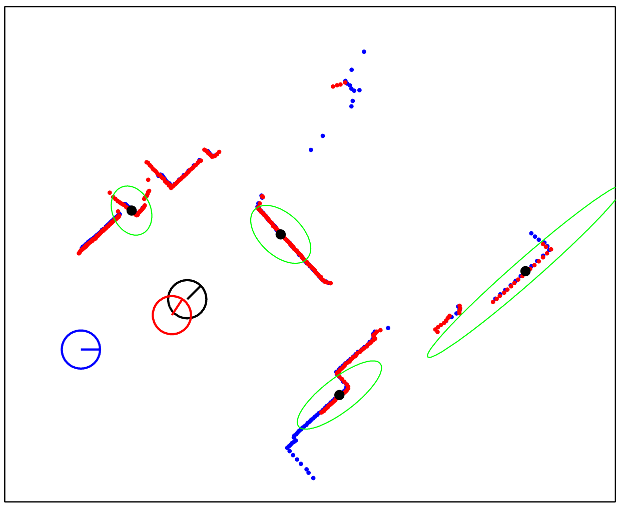
Weighted Approach

Explicit models of uncertainty
& noise sources for each point pair:

- Sensor noise & errors
  - Range noise
  - Scan angle uncertainty
- Point correspondence uncertainty
  - Due to a geometric effect

Correspondence Errors





Improvement vs. unweighted method:

- More accurate displacement estimate
- More realistic covariance estimate
- Increased robustness to initial conditions
- Improved convergence

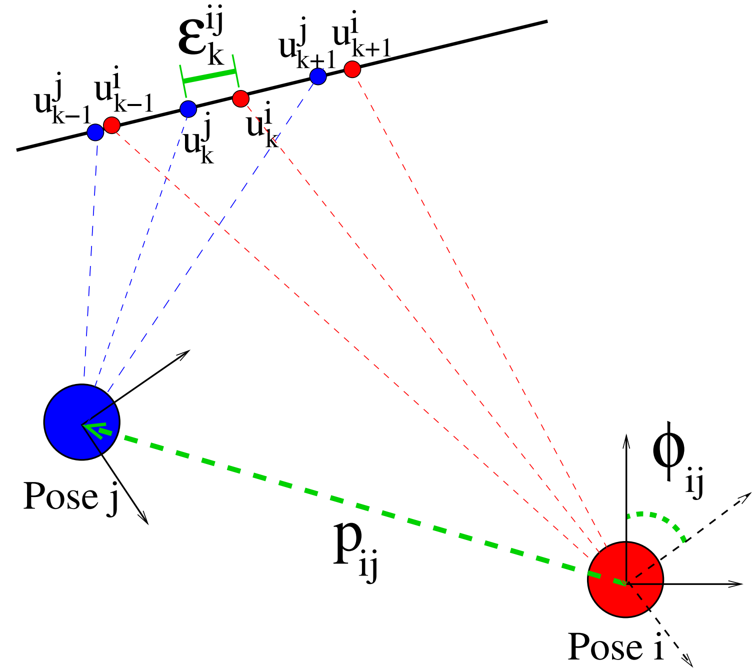
Weighted Formulation

Goal: Estimate displacement (p_{ij}, ϕ_{ij})

Measured range data
from poses i and j

$$\begin{aligned} u_k^i &= \delta u_k^i + b_k^i + r_k^i \\ u_k^j &= \delta u_k^j + b_k^j + r_k^j \end{aligned}$$

sensor noise | bias | true range



Error ε_k^{ij} between k^{th} scan point pair

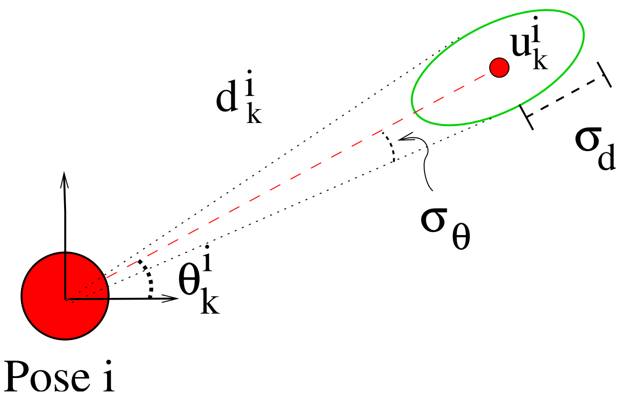
$$\varepsilon_k^{ij} = \hat{\vec{u}}_k^i - R_{ij}\hat{\vec{u}}_k^j - p_{ij} \quad R_{ij} = rotation\ through\ \phi_{ij}$$

$$\varepsilon_k^{ij} = \underbrace{(\vec{u}_k^i - R_{ij}\vec{u}_k^j - p_{ij})}_{(Noise\ Error)} + \underbrace{(\vec{b}_k^i - R_{ij}\vec{b}_k^j)}_{(Bias\ Error)} + \underbrace{(\delta\vec{u}_k^i - R_{ij}\delta\vec{u}_k^j)}_{(Correspondence\ Error)}$$

Covariance of Error Estimate

$$
\begin{aligned}
P_k^{ij} &\triangleq E\left[\varepsilon_k^{ij}(\varepsilon_k^{ij})^T\right] \\
&= \underbrace{{}^N P_k^i + R_{ij}\,{}^N P_k^j R_{ij}^T}_{(Sensor\ Noise)} + \underbrace{{}^B P_k^i + R_{ij}\,{}^B P_k^j R_{ij}^T}_{(Sensor\ Bias)} + \underbrace{{}^C P_k^{ij}}_{(Correspondence)}
\end{aligned}
$$

1) Sensor Noise



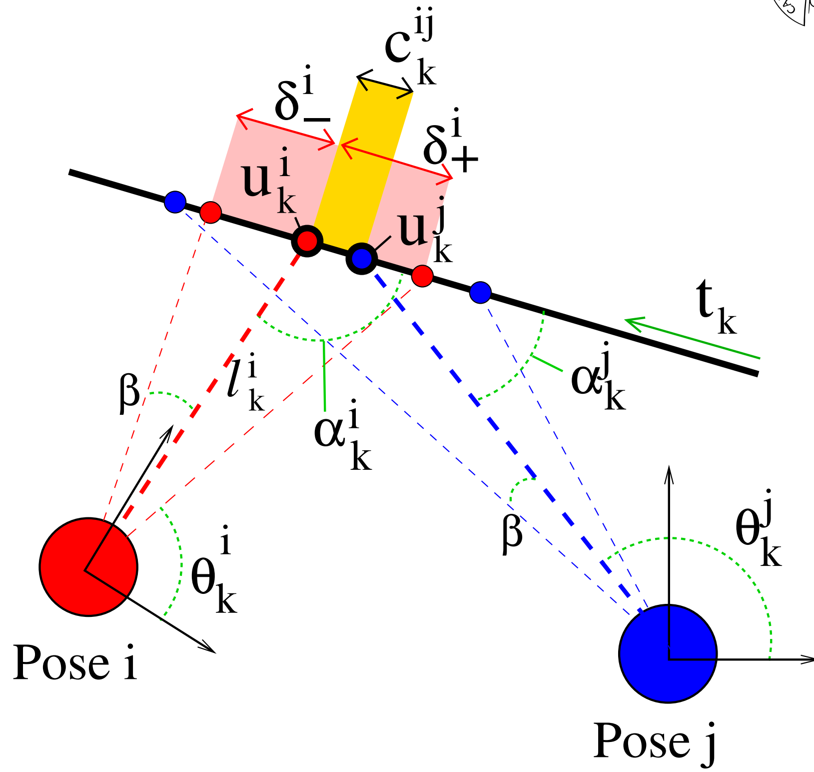
$$
\begin{aligned}
{}^N P_k^i &= E\left[\delta\vec{u}_k^i(\delta\vec{u}_k^i)^T\right] \\
{}^N P_k^i &= \frac{(d_k^i)^2\sigma_\theta^2}{2}\begin{bmatrix} 2\sin^2\theta_k^i & -\sin 2\theta_k^i \\ -\sin 2\theta_k^i & 2\cos^2\theta_k^i \end{bmatrix} + \frac{\sigma_d^2}{2}\begin{bmatrix} 2\cos^2\theta_k^i & \sin 2\theta_k^i \\ \sin 2\theta_k^i & 2\sin^2\theta_k^i \end{bmatrix}
\end{aligned}
$$

2) Sensor Bias

Neglect for now - more details in dissertation

3) Correspondence Error $= c_k^{ij}$

Estimate bounds of c_k^{ij} from the geometry of the boundary and robot poses



Max error $= \frac{1}{4}(\delta_+^i + \delta_-^i)$

$$= \frac{l_k^i \sin\beta}{2}\left[\frac{\sin\alpha_k^i \cos\beta}{\sin^2\alpha_k^i - \sin^2\beta}\right]$$

Assume uniform distribution

$$E[(\mu_k^{ij})^2] = \frac{(\delta_+^i)^3 + (\delta_-^i)^3}{3(\delta_+^i + \delta_-^i)} \quad where\ \mu_k^{ij} = c_k^{ij} t_k$$

$$\begin{aligned} {}^C P_k^i &= E[c_k^{ij}(c_k^{ij})^T] = E[(\mu_k^{ij})^2] t_k t_k^T \\ &= \frac{(\delta_+^i)^3 + (\delta_-^i)^3}{3(\delta_+^i + \delta_-^i)} \begin{bmatrix} \cos^2\eta_k^i & \cos\eta_k^i \sin\eta_k^i \\ \cos\eta_k^i \sin\eta_k^i & \sin^2\eta_k^i \end{bmatrix} \end{aligned}$$

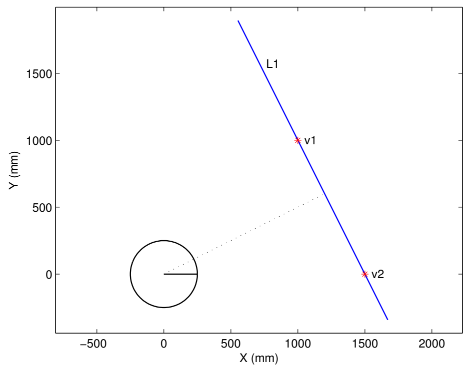
Determination of Incidence Angles

Goal : Find incidence angles α_k^i and α_k^j

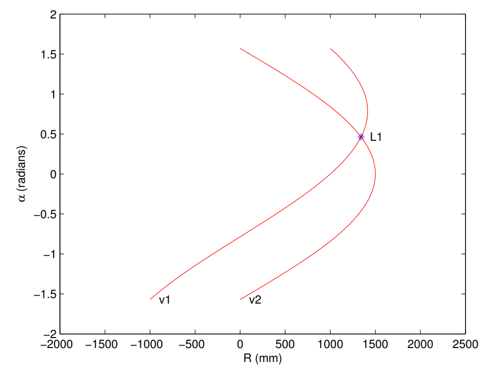
Approach : Use the Hough transform to extract underlying lines

Hough Transform

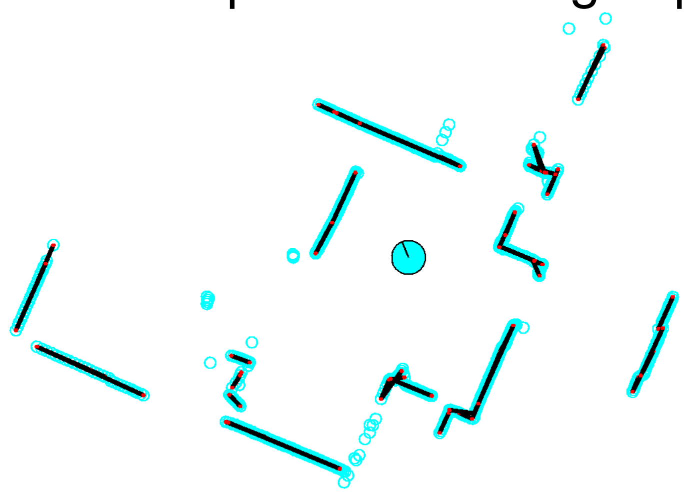
- General pattern detection method
- Fits lines to range data
- Local incidence angle estimated from line tangent and scan angle
- Common technique in vision community



Real space



Hough space



Maximum Likelihood Estimation

Likelihood of obtaining errors ε_k^{ij} given displacement g_{ij}

$$\mathcal{L}(\{\varepsilon_k^{ij}\}|g_{ij}) = \prod_{k=1}^{n_{ij}} \frac{e^{-\frac{1}{2}(\varepsilon_k^{ij})^T (P_k^{ij})^{-1} \varepsilon_k^{ij}}}{2\pi\sqrt{\det P_k^{ij}}} \qquad g_{ij} = \begin{bmatrix} p_{ij} \\ \phi_{ij} \end{bmatrix}$$

Non-linear Optimization Problem $\nabla(\mathcal{L}) = 0$

- Position displacement estimate obtained in closed form

$$\hat{p}_{ij} = P_{pp} \sum_{k=1}^{n_{ij}} \left((P_k^{ij})^{-1} (\hat{\vec{u}}_k^i - \hat{R}_{ij} \hat{\vec{u}}_k^j) \right) \qquad P_{pp} = \left(\sum_{k=1}^{n_{ij}} (P_k^{ij})^{-1} \right)^{-1}$$

- Orientation estimate found using 1-D numerical optimization, or series expansion methods

$$\delta\hat{\phi}_{ij} \simeq -\frac{\sum_{i=1}^{n_{ij}} p_k^T (P_k^{ij})^{-1} J q_k}{\sum_{k=1}^{n_{ij}} q_k^T J (P_k^{ij})^{-1} J q_k}, \quad J = \left[\begin{array}{cc} 0 & -1 \\ 1 & 0 \end{array} \right], \quad \begin{array}{rcl} q_k & = & \hat{R}_{ij} \hat{\vec{u}}_k^j \\ p_k & = & \hat{\vec{u}}_k^i - \hat{p}_{ij} - \hat{R}_{ij} \hat{\vec{u}}_k^j \end{array}$$

Experimental Results: Robustness Testing

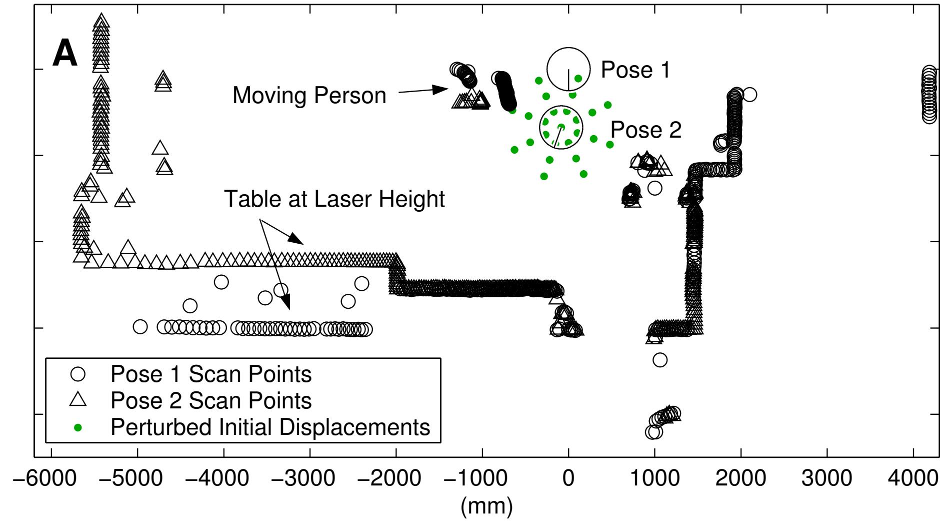
- 1525 trials with different initial displacement guesses
- Max initial error = 600mm, 0.6 radians
- Successful convergence defined by covariance

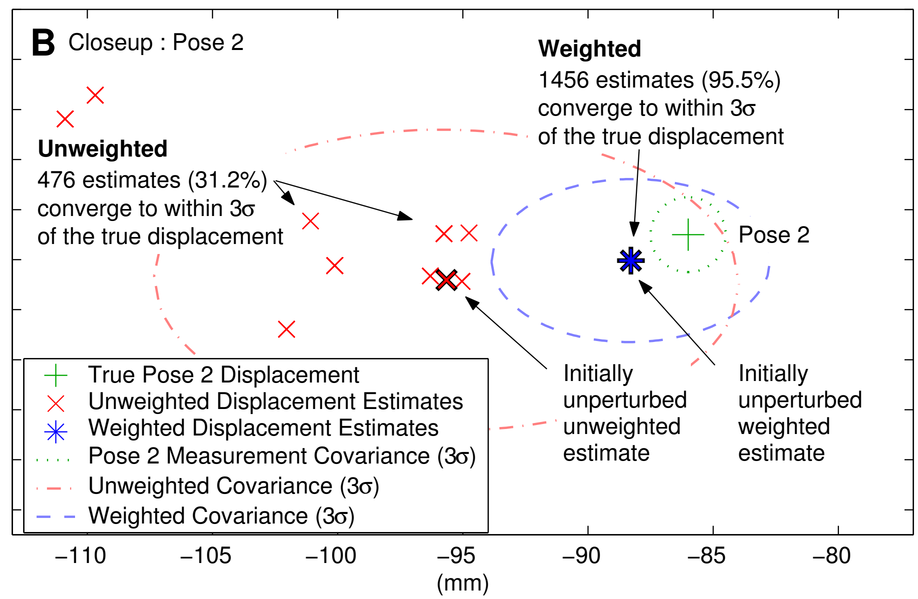
Weighted Results

- 95.5% converge
- Average error = 2.5mm, .57 mrad

Unweighted Results

- 31.2% converge
- Average error = 11.1mm, 16 mrad





Experimental Results: Robustness Testing

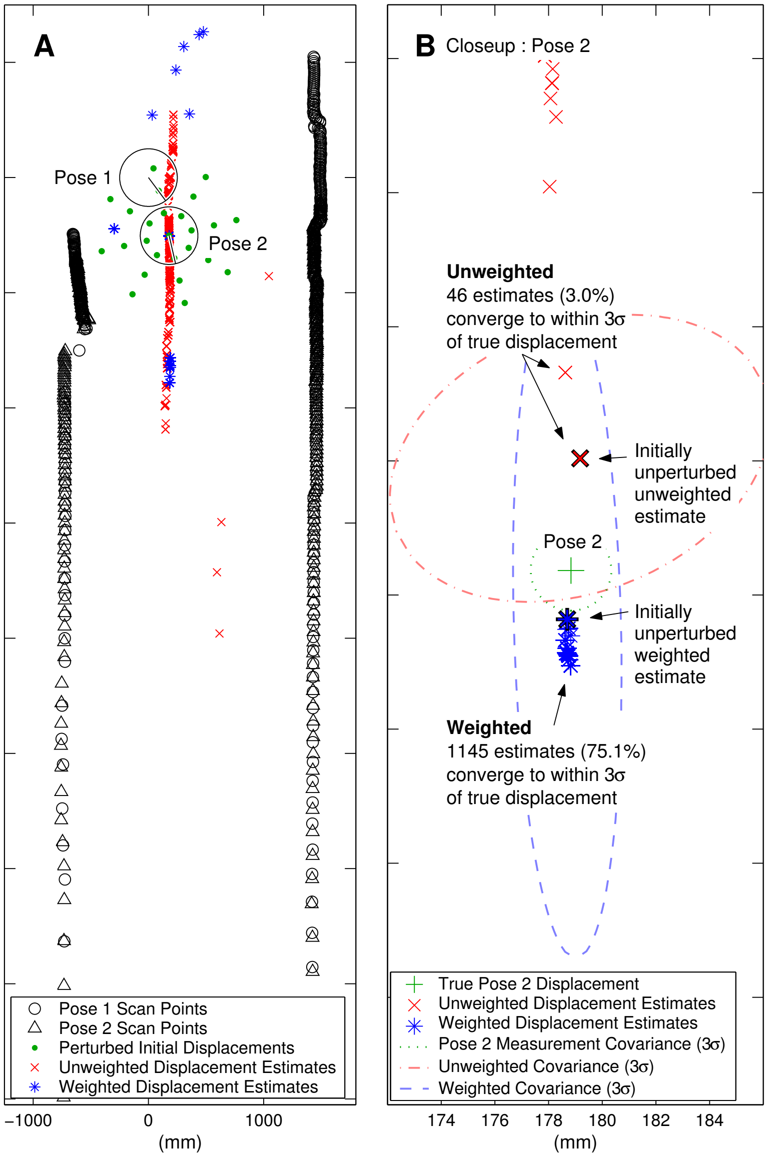
- 1525 trials with different initial displacement guesses
- Max initial error = 600mm, 0.6 radians
- Successful convergence defined by covariance

Weighted Results

- 75.1% converge
- Average error = 3.1mm, 0.04 mrad

Unweighted Results

- 3.0% converge
- Average error = 14.5mm, .47 mrad



Experimental Results : Long Run

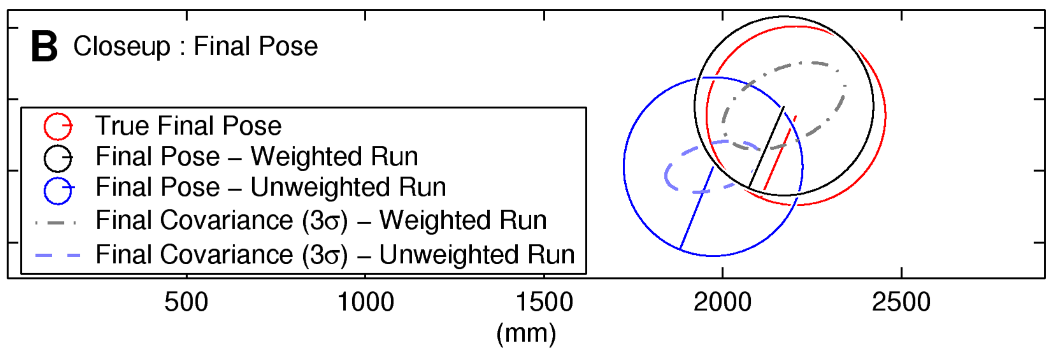
32.8 meter, 109 step loop path

Weighted Results

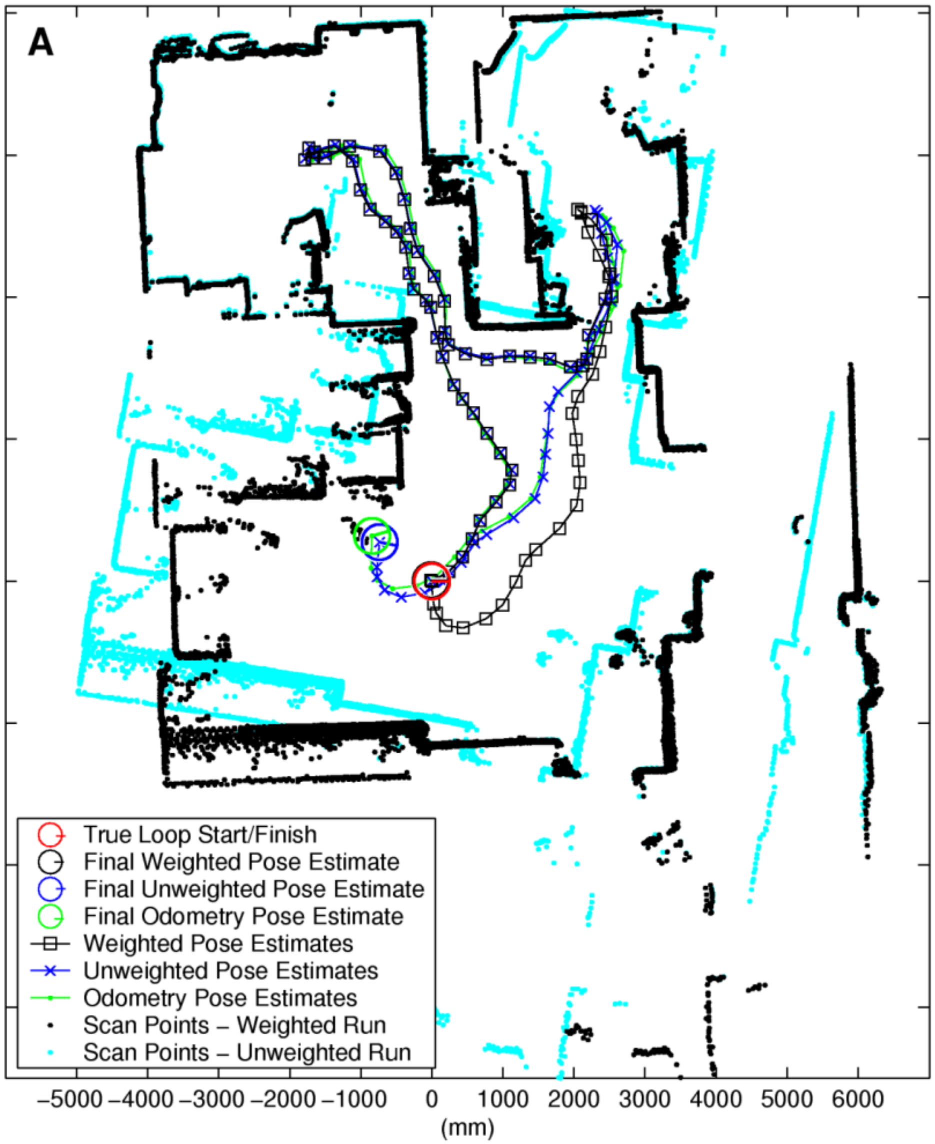
- Final error = 43mm, 2.9 mrad

Unweighted Results

- Final error = 271mm, 21 mrad



Experimental Results : Long Run



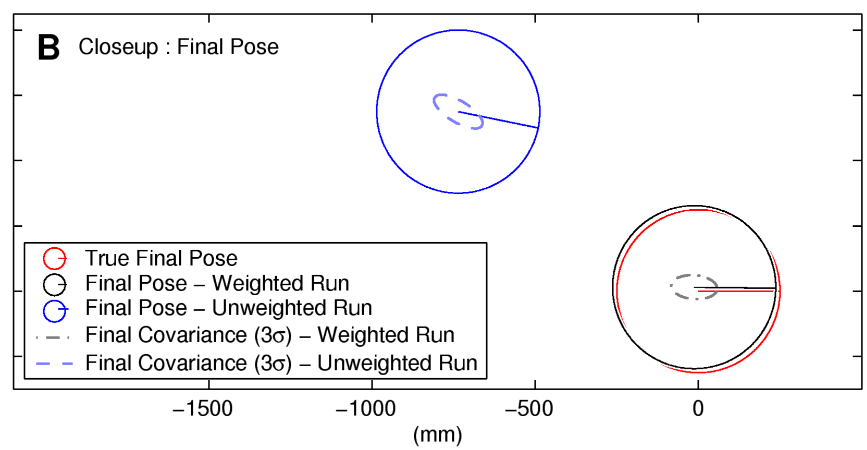
24.2 meter, 83 step loop path

Weighted Results

- Final error = 18mm, 13 mrad

Unweighted Results

- Final error = 919mm, 200 mrad



WLSM Conclusions

Contributions:

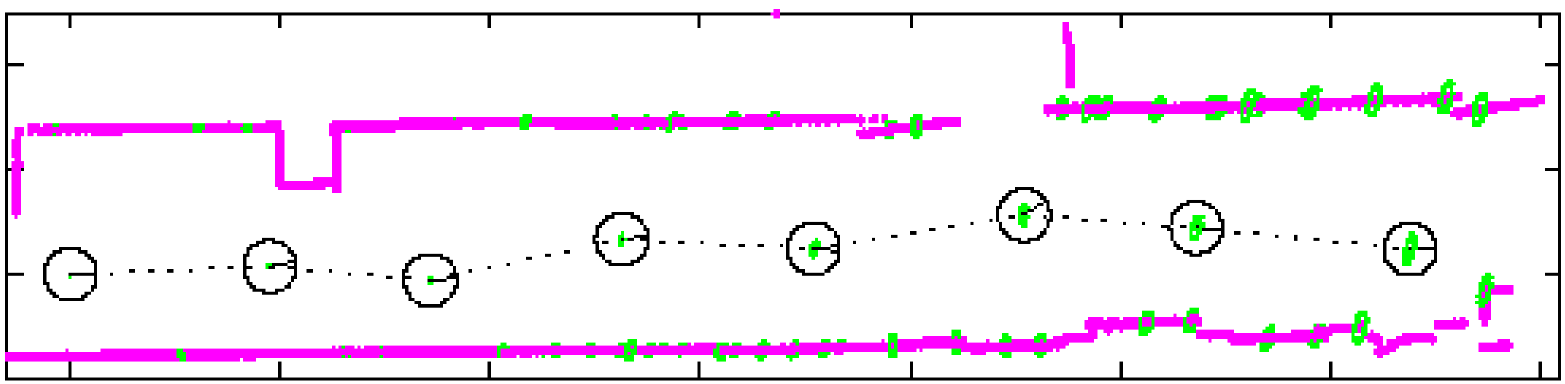
- A method of point correspondence error compensation through modeling
- A general approach to incorporate uncertainty into scan match displacement estimates

Results:

- More accurate relative position estimation
- More accurate covariance
- More robust to poor initial guess
- More efficient in the case of poor initial guess

Method 2) Line Segment Feature Based Localization and Mapping

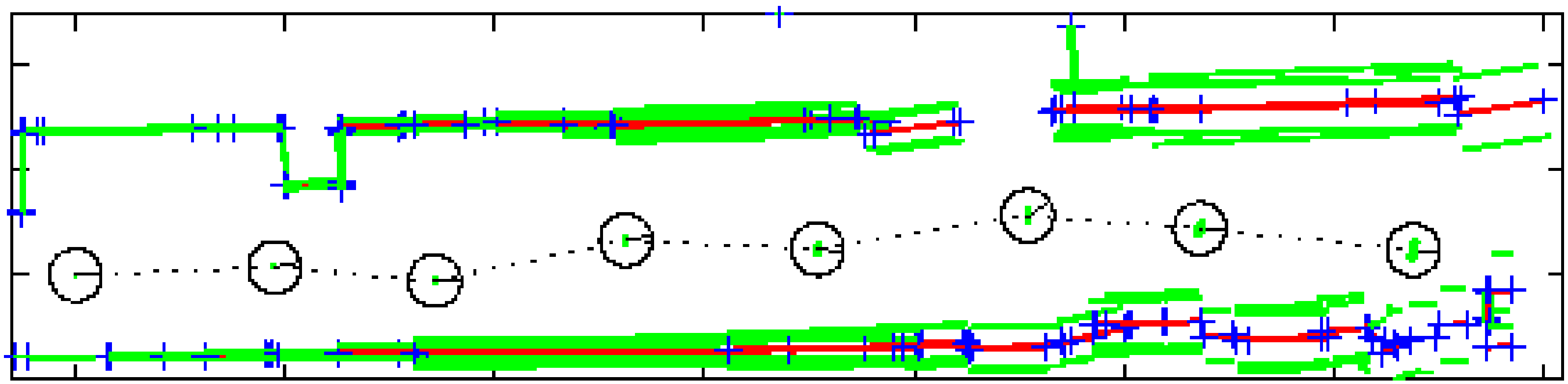
1. Define and extract features from the raw data
2. Compare and align features across data sets
3. Use assembled feature based maps for localization



Raw point data

Method 2) Line Segment Feature Based Localization and Mapping

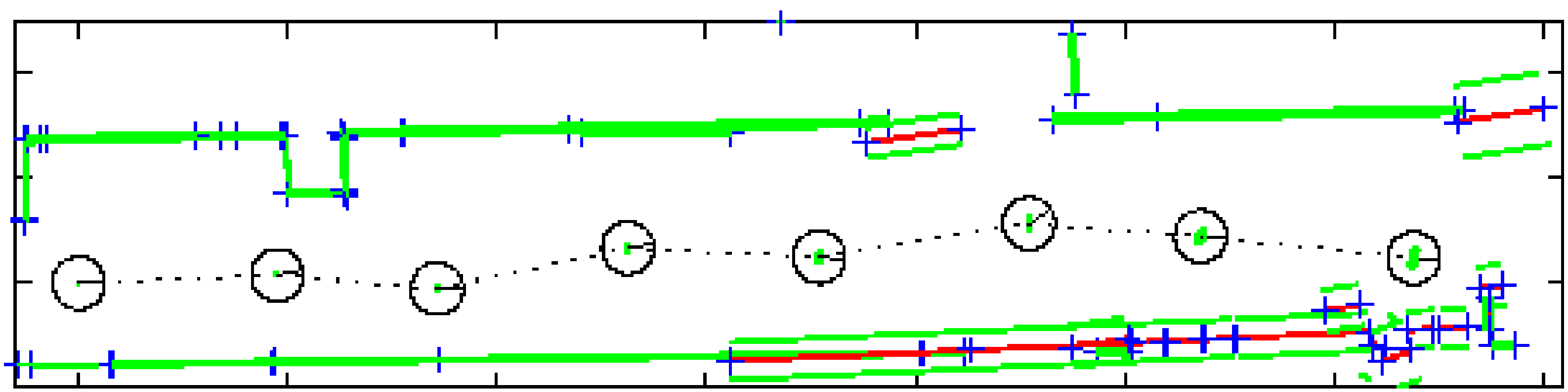
1. Define and extract features from the raw data
2. Compare and align features across data sets
3. Use assembled feature based maps for localization



Extracted line segment features

Method 2) Line Segment Feature Based Localization and Mapping

1. Define and extract features from the raw data
2. Compare and align features across data sets
3. Use assembled feature based maps for localization



Merged feature based map

Method 2) Line Segment Feature Based Localization and Mapping

1. Define and extract features from the raw data
2. Compare and align features across data sets
3. Use assembled feature based maps for localization

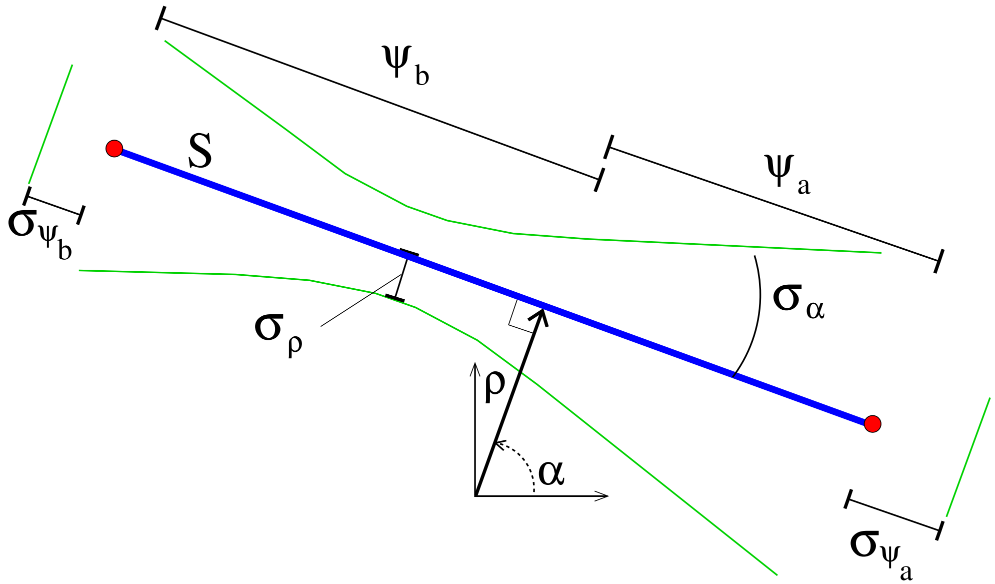
Benefits vs. point based methods

- More efficient data representation for reduced storage
- More efficient localization and mapping algorithms
- More discerning data association

Background

- Fitting lines to range data has been done
  [Ayache, Faugeras, Castellanos]
- I introduce rigorous noise modeling, and novel feature correspondence methods

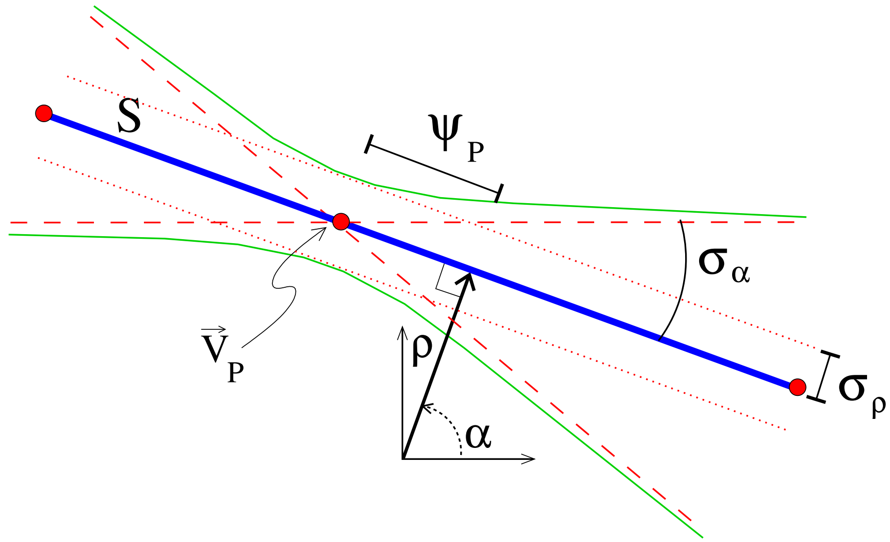
Line Segment Feature Representation



$$S = \begin{bmatrix} \alpha \\ \rho \\ \psi_a \\ \psi_b \end{bmatrix} \quad P_S = \begin{bmatrix} P_{\alpha\alpha} & P_{\alpha\rho} & P_{\alpha\psi_a} & P_{\alpha\psi_b} \\ P_{\rho\alpha} & P_{\rho\rho} & P_{\rho\psi_a} & P_{\rho\psi_b} \\ P_{\psi_a\alpha} & P_{\psi_a\rho} & P_{\psi_a\psi_a} & P_{\psi_a\psi_b} \\ P_{\psi_b\alpha} & P_{\psi_b\rho} & P_{\psi_b\psi_a} & P_{\psi_b\psi_b} \end{bmatrix}$$

$$L = \begin{bmatrix} \alpha \\ \rho \end{bmatrix} \quad P_L = \begin{bmatrix} P_{\alpha\alpha} & P_{\alpha\rho} \\ P_{\rho\alpha} & P_{\rho\rho} \end{bmatrix}.$$

Center of Rotational Uncertainty



$$\begin{bmatrix} \sigma_\alpha^2 & 0 \\ 0 & \sigma_\rho^2 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ -\delta\psi_P & 1 \end{bmatrix} P_L \begin{bmatrix} 1 & -\delta\psi_P \\ 0 & 1 \end{bmatrix}, \quad \delta\psi_P = -P_{\rho\alpha}/P_{\alpha\alpha}$$

Center point : $$\vec{V}_P = \begin{bmatrix} x_P \\ y_P \end{bmatrix} = \begin{bmatrix} \rho\cos(\alpha) - \psi_P \sin(\alpha) \\ \rho\sin(\alpha) + \psi_P\cos(\alpha) \end{bmatrix}$$

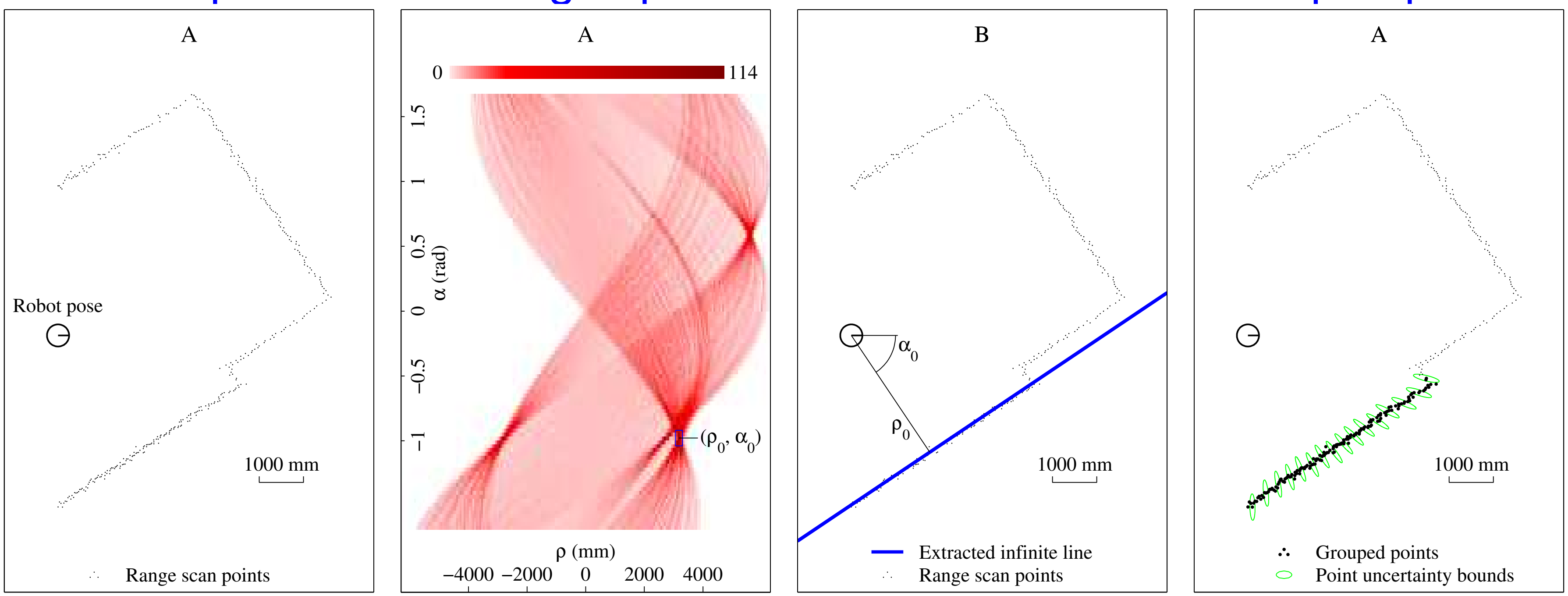
$$\begin{bmatrix} \sigma_\alpha^2 & 0 & 0 & 0 \\ 0 & \sigma_\rho^2 & 0 & 0 \\ 0 & 0 & \sigma_{\psi_a}^2 & 0 \\ 0 & 0 & 0 & \sigma_{\psi_b}^2 \end{bmatrix} = H_P^{-1} P_S (H_P^{-1})^T, \quad H_P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \psi_P & 1 & 0 & 0 \\ -P_{\psi_a\alpha}/P_{\alpha\alpha} & 0 & 1 & 0 \\ -P_{\psi_b\alpha}/P_{\alpha\alpha} & 0 & 0 & 1 \end{bmatrix}$$

Line Segment Feature Extraction

1. Group colinear points using a Hough transform
2. Fit optimal infinite line using point noise models
3. Extract endpoints and repeat for any unused points

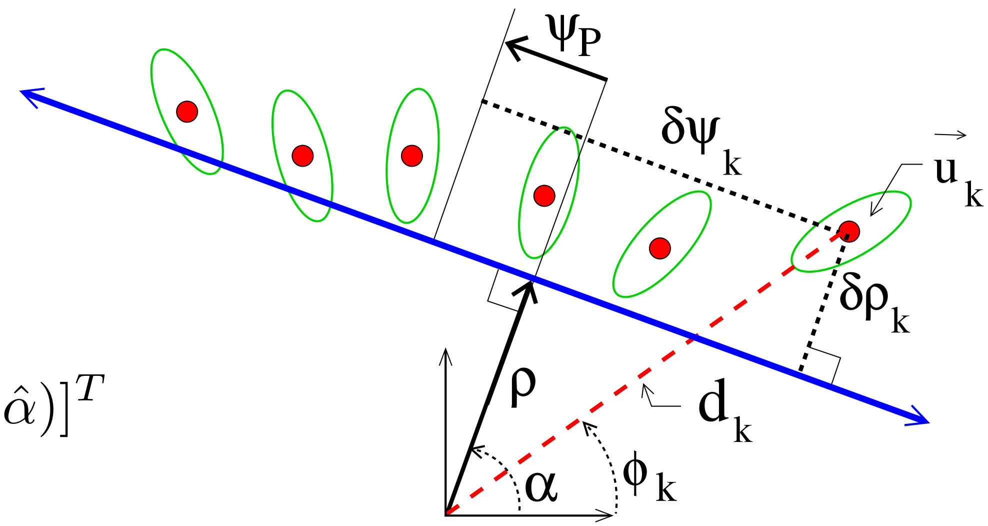
1. Initial grouping using a Hough transform

Raw points | Hough space | Infinite line | Grouped points



Feature Extraction: Weighted Line Fitting

- Find $L = [\alpha, \rho]$ which minimize the set of errors $\delta\rho$



$$\delta\rho_k = \hat{d}_k \cos(\hat{\alpha} - \hat{\theta}_k) - \hat{\rho}$$
$$P_{\delta\rho_k} = [\cos(\hat{\alpha}) \sin(\hat{\alpha})] P_{u_k} [\cos(\hat{\alpha}) \sin(\hat{\alpha})]^T$$

First calculate center of rotational uncertainty position ψ_P:

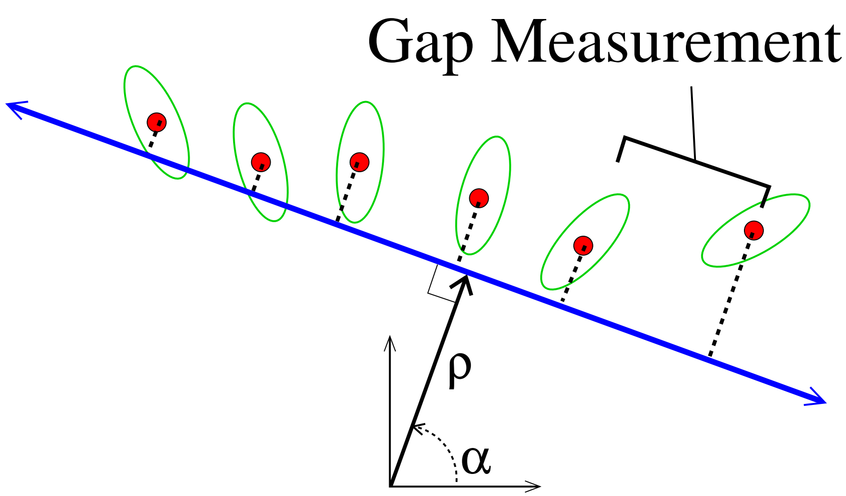
$$\psi_P = \frac{\sum_{k=1}^{n} \frac{\hat{\psi}_k}{P_{\delta\rho_k}}}{\sum_{k=1}^{n} \frac{1}{P_{\delta\rho_k}}}$$

Then use a maximum likelihood approach compute

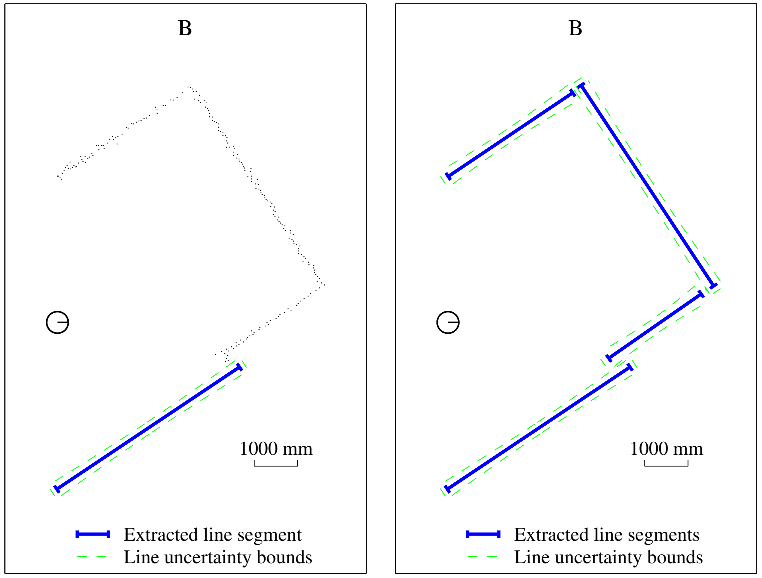
$$\rho = \frac{\sum_{k=1}^{n} \frac{\hat{d}_k \cos(\hat{\alpha} - \hat{\theta}_k)}{P_{\delta\rho_k}}}{\sum_{k=1}^{n} \frac{1}{P_{\delta\rho_k}}}, \quad \delta\alpha = -\frac{\sum_{k=1}^{n} \left(\frac{\delta\rho_k \delta\psi_k}{P_{\delta\rho_k}}\right)}{\sum_{k=1}^{n} \left(\frac{(\delta\psi_k)^2}{P_{\delta\rho_k}}\right)}$$

Feature Extraction: Endpoint Detection

- Split line at large gap
- Determine endpoint covariance



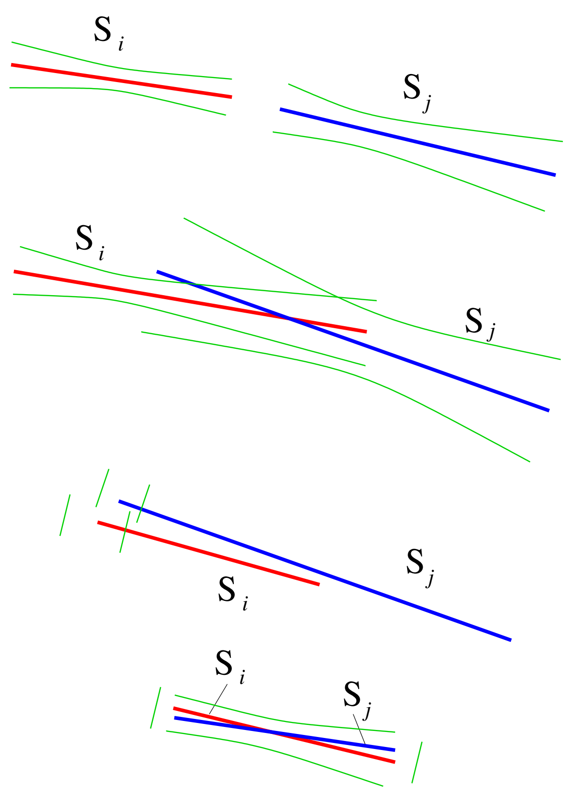
- Repeat to find multiple lines



Line Segment Feature Correspondence

Hypothesis: Feature A from pose i and feature B from pose j represent measurements of the same aspect of the environment

- Type I error : Rejection of a true hypothesis
- Type II error : Acceptance of a false hypothesis

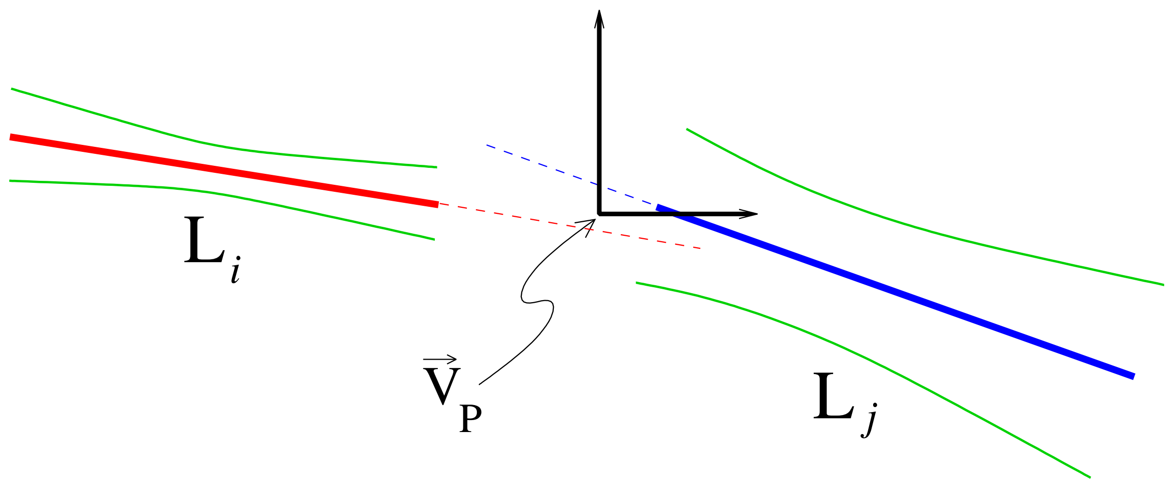


Hypothesis test types:

- Chi-squared hypothesis test - Addresses type I errors
- Probabilistic confidence test - Addresses type II errors

Chi-squared Hypothesis Tests

Can the observed error be reasonably explained by the model?



Underlying line chi-squared test:

- Compute combined center of rotational uncertainty $\vec{V}_P$
- Transform both lines to frame at $\vec{V}_P$
- Calculate Mahalinobis distance

$$D^2 = \begin{bmatrix} \alpha_i - \alpha_j \\ \rho_i - \rho_j \end{bmatrix} (P_{L_i} + P_{L_j})^{-1} \begin{bmatrix} \alpha_i - \alpha_j \\ \rho_i - \rho_j \end{bmatrix}^T$$

The hypothesis is rejected if $D^2 > \chi^2$

-The χ^2 threshold is from a chi-squared distribution at a chosen probability

Feature Correspondence: Overlap Test

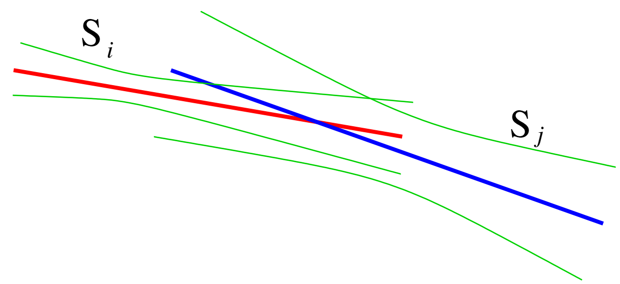
$$\ell^i = \psi_b^i - \psi_a^i \quad \ell^j = \psi_a^j - \psi_b^j \quad \Delta_\psi^{ij} = \frac{\ell^i + \ell^j}{2}$$

Piecewise calculation of D^2

$$if \quad |\psi_c^i - \psi_c^j| \leq \Delta_\psi^{ij} \; then \; D^2 = 0$$

$$if \quad \psi_c^i - \psi_c^j > \Delta_\psi^{ij} \; then \; D^2 = \frac{(\psi_c^i - \psi_c^j - \Delta_\psi^{ij})^2}{P^i_{\psi_a\psi_a} + P^j_{\psi_b\psi_b}}$$

$$if \quad \psi_c^i - \psi_c^j < -\Delta_\psi^{ij} \; then \; D^2 = \frac{(\psi_c^i - \psi_c^j + \Delta_\psi^{ij})^2}{P^i_{\psi_b\psi_b} + P^j_{\psi_a\psi_a}}$$



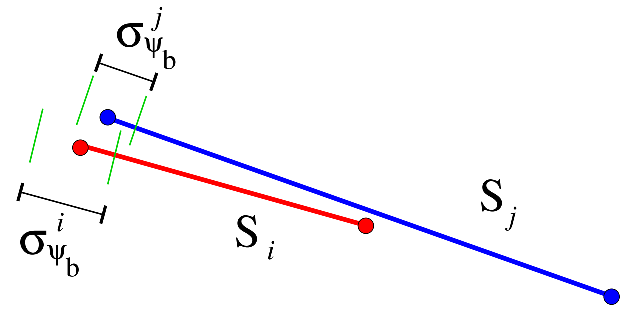
The hypothesis is rejected if $D^2 > \chi^2$

Feature Correspondence: Endpoint Test

- Endpoint Mahalinobis distance calculation

$$D_a^2 = \frac{(\psi_a^i - \tilde{\psi}_a^j)^2}{P^i_{\psi_a\psi_a} + P^j_{\psi_a\psi_a}}$$

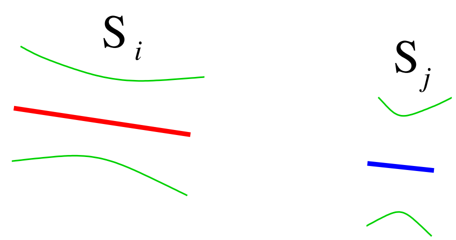
$$D_b^2 = \frac{(\psi_b^i - \tilde{\psi}_b^j)^2}{P^i_{\psi_b\psi_b} + P^j_{\psi_b\psi_b}}$$



- Only matching aspects of a feature are later merged
- Chi-squared test not effective at detecting false positives

Probabilistic Confidence Test

What is the likelihood of a false positive?



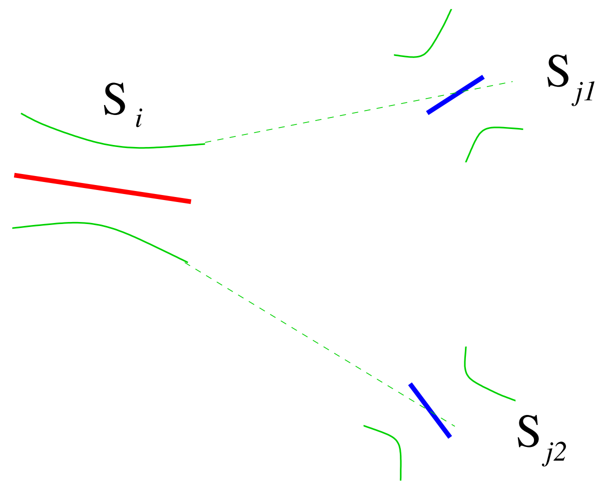
$$
\begin{aligned}
\chi^2 &= \frac{(|\alpha^i - \alpha^j|}{P^i_{\alpha\alpha} + P^j_{\alpha\alpha}} \\
|\alpha^i - \alpha^j| &= \bar{\Delta}^{ij}_{\alpha} = \sqrt{\chi^2(P^i_{\alpha\alpha} + P^j_{\alpha\alpha})}
\end{aligned}
$$

With a similar calculation for ρ, the probability of a random match is:

$$
\mathcal{P}(M_{ij}) = \frac{\bar{\Delta}^{ij}_{\alpha}}{2\pi}\left(\frac{\bar{\Delta}^{ij}_{\rho}}{2d_{max}}\right)
$$

Probabilistic Confidence Test

What is the likelihood of a false positive?



$$\chi^2 = \frac{(|\alpha^i - \alpha^j|}{P^i_{\alpha\alpha} + P^j_{\alpha\alpha}}$$

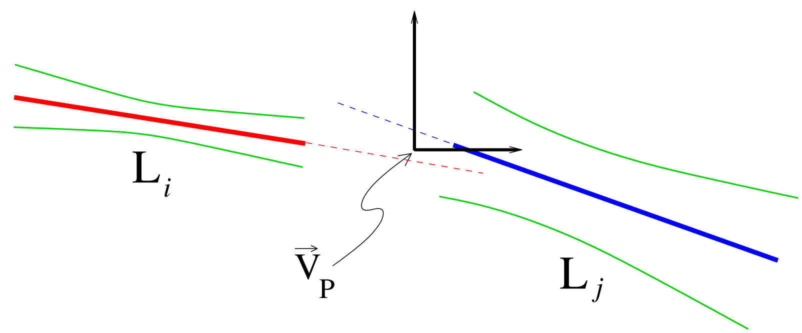
$$|\alpha^i - \alpha^j| = \bar{\Delta}^{ij}_{\alpha} = \sqrt{\chi^2 (P^i_{\alpha\alpha} + P^j_{\alpha\alpha})}$$

With a similar calculation for ρ, the probability of a random match is:

$$\mathcal{P}(M_{ij}) = \frac{\bar{\Delta}^{ij}_{\alpha}}{2\pi} \left(\frac{\bar{\Delta}^{ij}_{\rho}}{2d_{max}} \right)$$

Line Merge

- Transform both features to frame at $\vec{V}_P$
- Calculate hypothesis tests
- Merge line feature portions which correspond



Full segment merge:

$$S_m^i = P_{S_m}^i\left((P_{S_i})^{-1}S_i + (P_{S_j})^{-1}S_j\right) \quad P_{S_m}^i = \left((P_{S_i})^{-1} + (P_{S_j})^{-1}\right)^{-1}$$

Underlying line only merge:

$$L_m^i = P_{L_m}^i\left((P_{L_i})^{-1}L_i + (P_{L_j})^{-1}L_j\right) \quad P_{L_m}^i = \left((P_{L_i})^{-1} + (P_{L_j})^{-1}\right)^{-1}$$

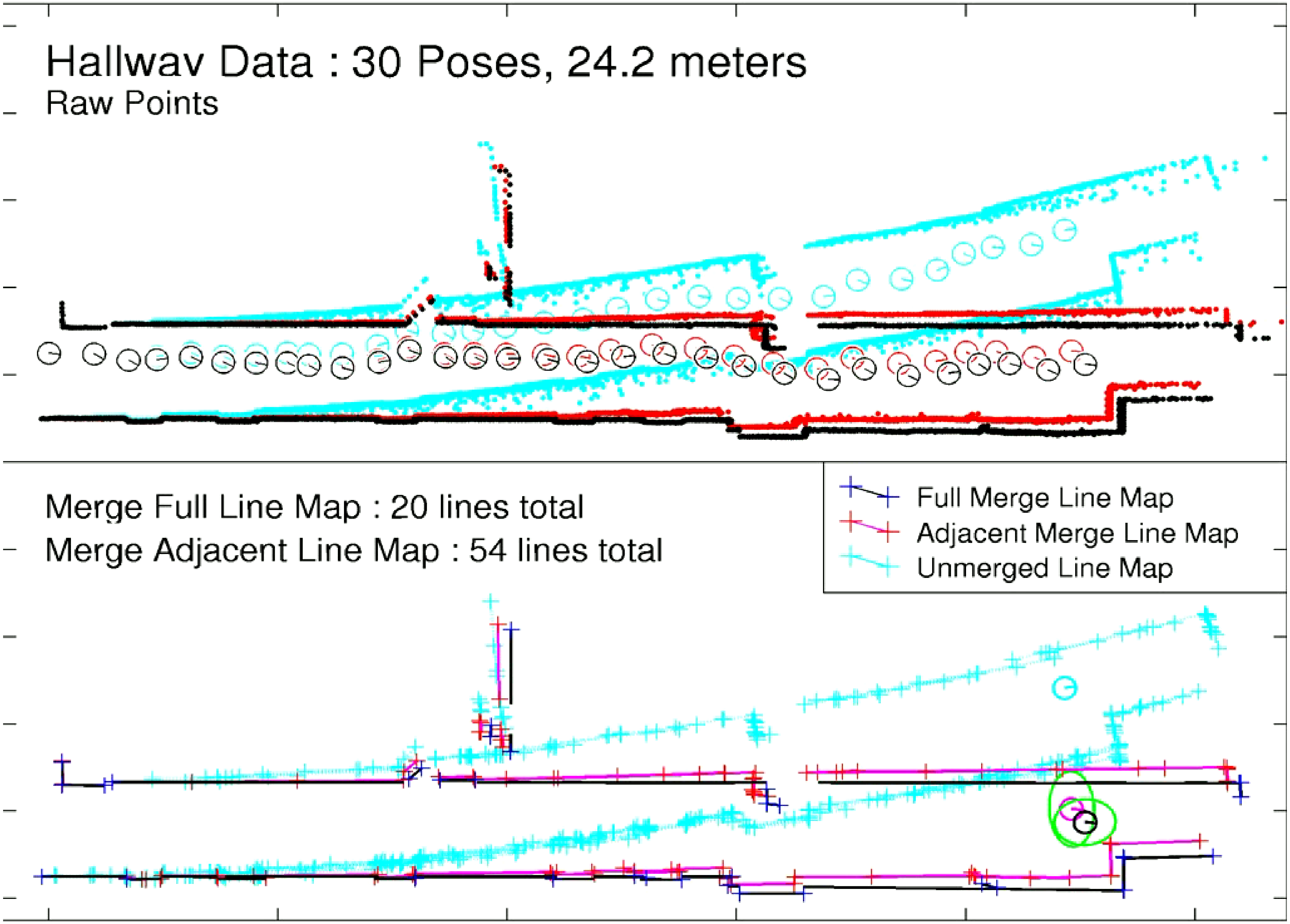
Unmerged ends are updated as follows: $\psi_a^m = \min(\psi_a^i, \psi_a^j)$, $\psi_b^m = \max(\psi_b^i, \psi_b^j)$

Kalman Filter Based SLAM

- Robot state at timestep k : $X_k = \begin{bmatrix} x & y & \phi & S_1 & \dots & S_n \end{bmatrix}_k^T$
- State covariance matrix at timestep k : P_{X_k}
- Propagation step : Integrates odometry
- Update step : Incorporates sensed features
  - Updates both robot position and feature coordinates

$$
\begin{aligned}
K_k &= P_{k|k-1} H_k^T \left(H_k P_{k|k-1} H_k^T + V_k P_{\bar{S}} V_k^T \right)^{-1} \\
\hat{X}_k &= \hat{X}_{k|k-1} + K_k (\bar{S} - h(\hat{X}_{k|k-1}, 0)) \\
P_k &= (I - K_k H_k) P_{k|k-1}
\end{aligned}
$$

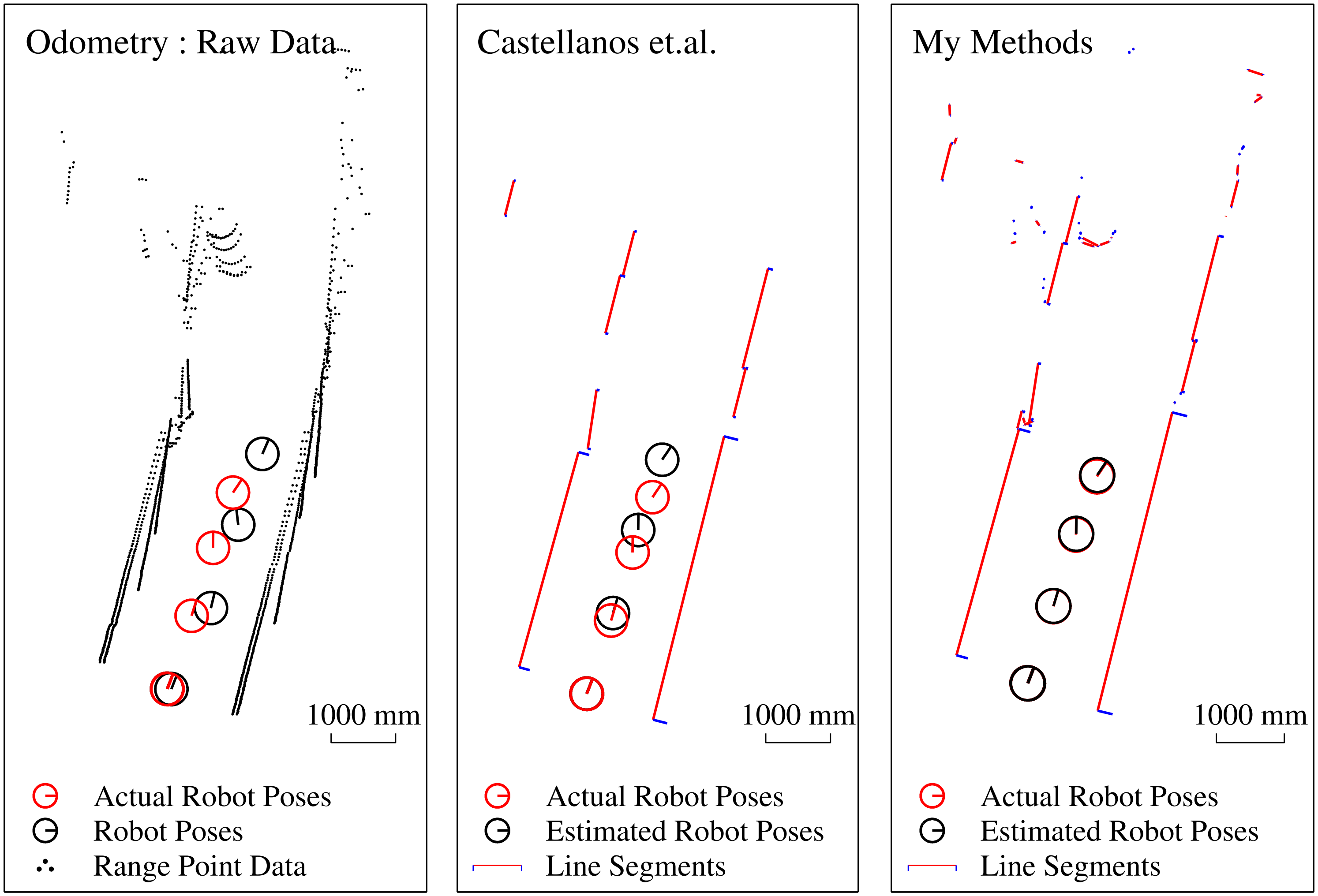
Matrix H_k depends on which aspects of the line segments were determined to match



Hallwav Data : 30 Poses, 24.2 meters
Raw Points
Merge Full Line Map : 20 lines total
Merge Adjacent Line Map : 54 lines total
Full Merge Line Map
Adjacent Merge Line Map
Unmerged Line Map

Line Segment Feature Based Mapping

Localization comparison : Errors due to lost data



Line Based Approach Conclusions

Contributions:

- An improved method of line feature extraction with individual point noise modeling
- An effective approach to feature correspondence :
  - Tests for partial feature matching
  - A method of estimating confidence of a feature pair match
- Improved compensation for non-linear effects
- A more flexible line feature :
  - Allows for comparison of very short line segments
  - Allows for effective merging of long line segments across gaps

Results:

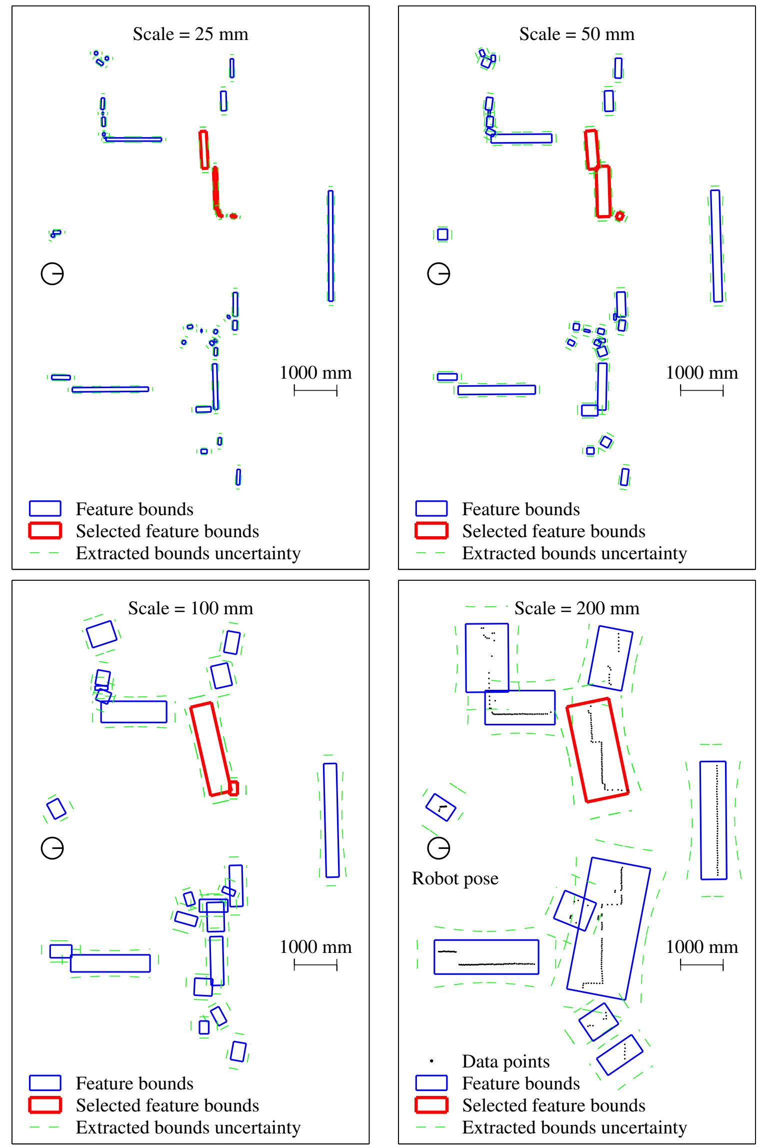
- Improved accuracy in localization and mapping
- more robust feature correspondence
- More efficient map representation without data loss

Method 3) A Multi-scale Approach

- Introduces a *block* feature
  - Extends the line segment with a notion of width
- Introduces a multi-scale tree structure
  - The data is represented at multiple scales
  - Related data is connected in the tree

Motivation:

- Computational efficiency



Multi-scale Background

Prior approaches to address computational complexity

- Sparsification of the information matrix [Leonard, Thrun]
- Selectively reduce the feature set [Newman]
- Rao Blackwellization for particle filtering based SLAM algorithms [Thrun]

Multi-scale approaches in robotics

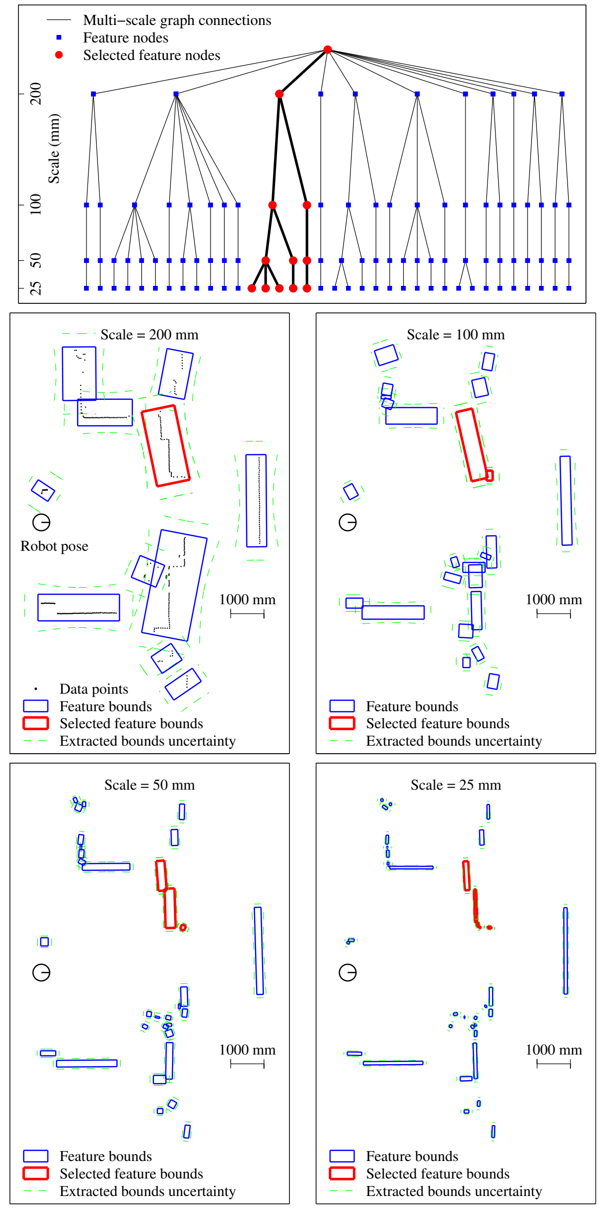
- Efficient data processing [Madhavan]
- Efficient representations [Theocharous, Thrun]

Multi-scale approaches in vision

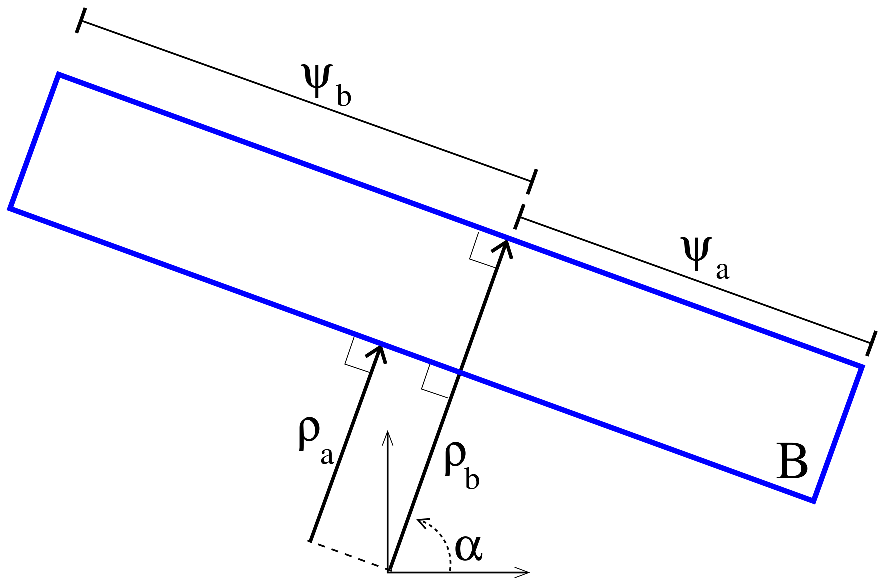
- Multi-scale features for object recognition [Lowe, Kadir]
- Multi-scale edge detection and filtering [Perona, Weickert]

Multi-scale Overview:

- Feature extraction
  - Multi-scale Hough transform
- Feature correspondence
  - Scale compensation
  - Partial feature matching
- Multi-scale tree structure
- Experimental results
  - Correspondence benefits
  - Robustness benefits
  - SLAM
  - Kidnapped robot problem

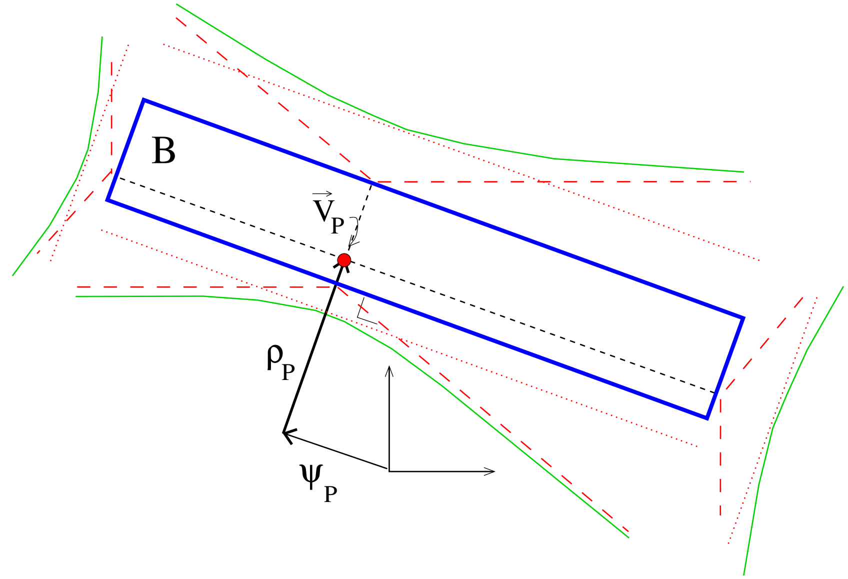


Block Feature Representation



$$B = \begin{bmatrix} \alpha \\ \rho_a \\ \rho_b \\ \psi_a \\ \psi_b \end{bmatrix}, \quad P_B = \begin{bmatrix} P_{\alpha\alpha} & P_{\alpha\rho_a} & P_{\alpha\rho_b} & P_{\alpha\psi_a} & P_{\alpha\psi_b} \\ P_{\rho_a\alpha} & P_{\rho_a\rho_a} & P_{\rho_a\rho_b} & P_{\rho_a\psi_a} & P_{\rho_a\psi_b} \\ P_{\rho_b\alpha} & P_{\rho_b\rho_a} & P_{\rho_b\rho_b} & P_{\rho_b\psi_a} & P_{\rho_b\psi_b} \\ P_{\psi_a\alpha} & P_{\psi_a\rho_a} & P_{\psi_a\rho_b} & P_{\psi_a\psi_a} & P_{\psi_a\psi_b} \\ P_{\psi_b\alpha} & P_{\psi_b\rho_a} & P_{\psi_b\rho_b} & P_{\psi_b\psi_a} & P_{\psi_b\psi_b} \end{bmatrix}$$

Center of Rotational Uncertainty



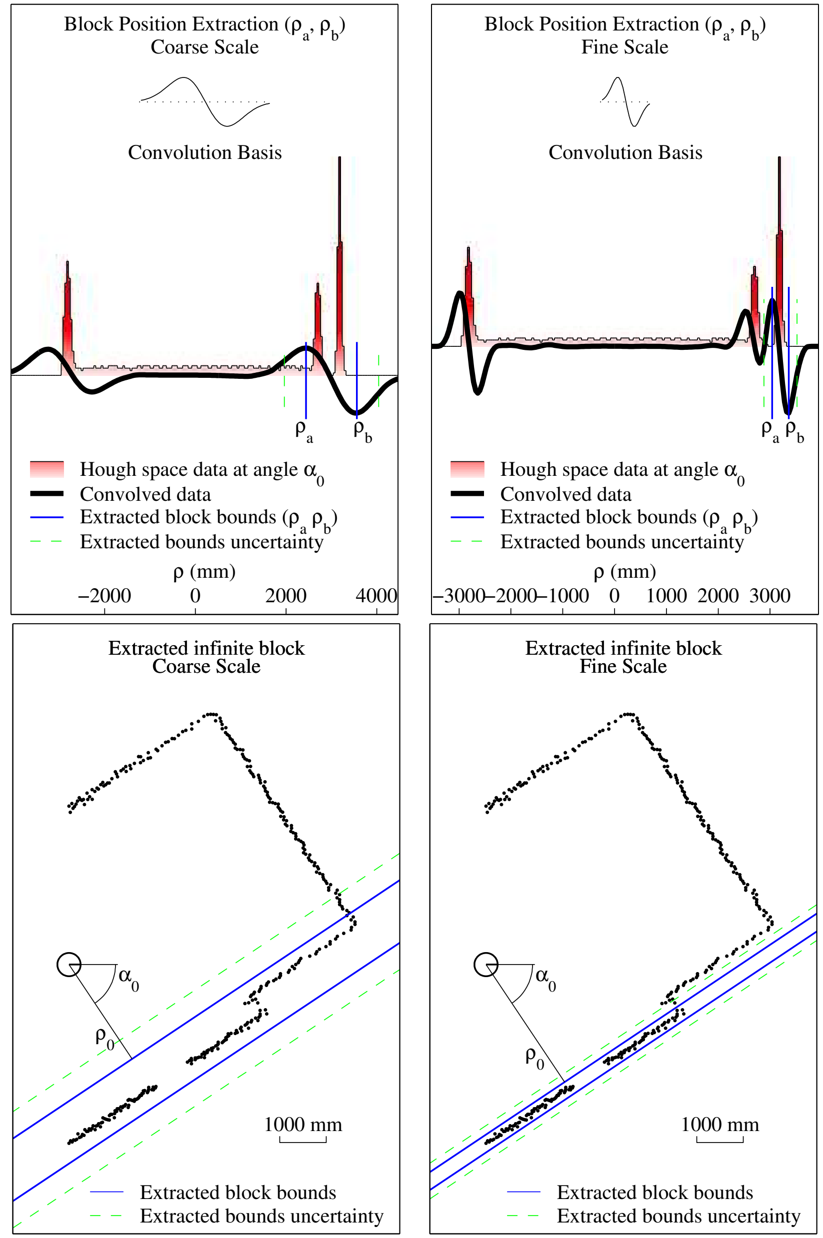
$$\begin{bmatrix} \sigma_\alpha^2 & 0 & 0 & 0 & 0 \\ 0 & \sigma_{\rho_a}^2 & 0 & 0 & 0 \\ 0 & 0 & \sigma_{\rho_b}^2 & 0 & 0 \\ 0 & 0 & 0 & \sigma_{\psi_a}^2 & 0 \\ 0 & 0 & 0 & 0 & \sigma_{\psi_b}^2 \end{bmatrix} = H_{P_B}^{-1} P_B (H_{P_B}^{-1})^T, \quad H_{P_B}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ -\psi_P & 1 & 0 & 0 & 0 \\ -\psi_P & 0 & 1 & 0 & 0 \\ -\rho_P & 0 & 0 & 1 & 0 \\ -\rho_P & 0 & 0 & 0 & 1 \end{bmatrix}$$

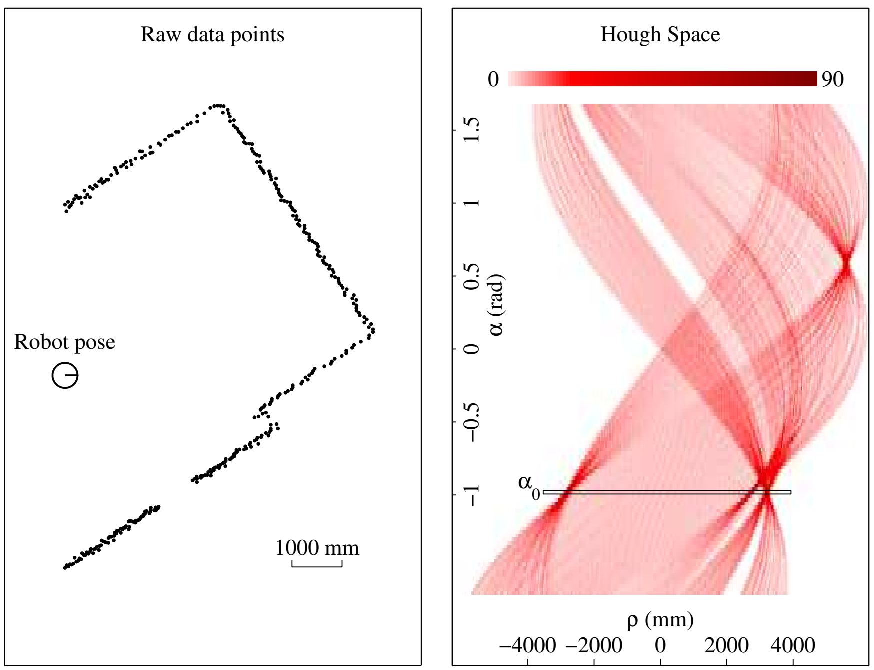
Center point :

$$\vec{V}_P = \begin{bmatrix} x_P \\ y_P \end{bmatrix} = \begin{bmatrix} \rho_P \cos(\alpha) - \psi_P \sin(\alpha) \\ \rho_P \sin(\alpha) + \psi_P \cos(\alpha) \end{bmatrix}$$

Block Feature Extraction

- Features are extracted sequentially using a multi-scale approach based on the Hough transform



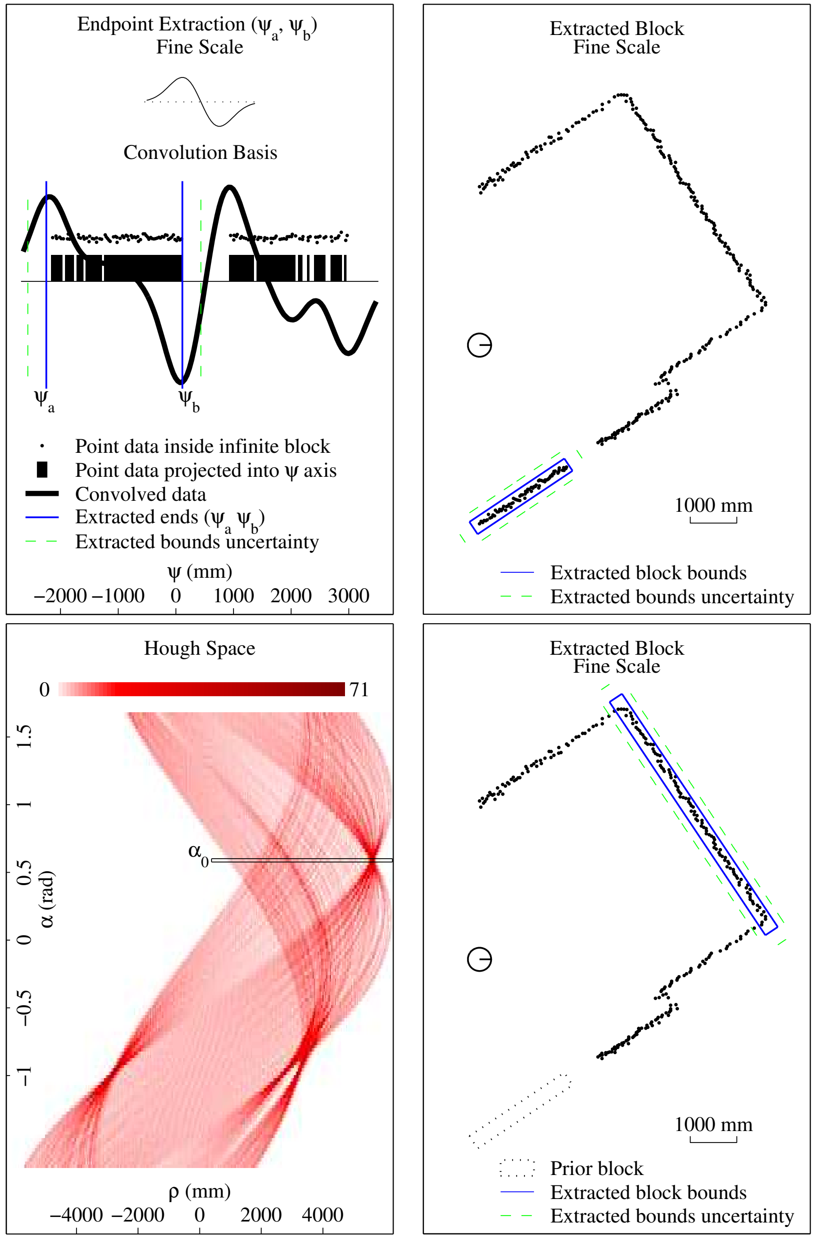


Block Feature Extraction

- Endpoints are extracted using a convolution analysis
- Subsequent features are extracted from remaining points

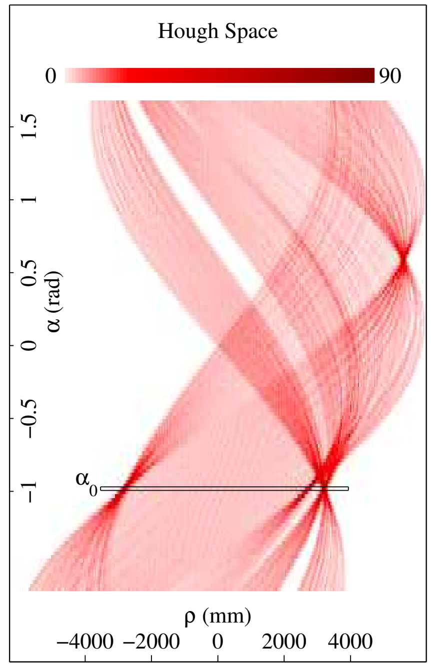
Covariance Terms:

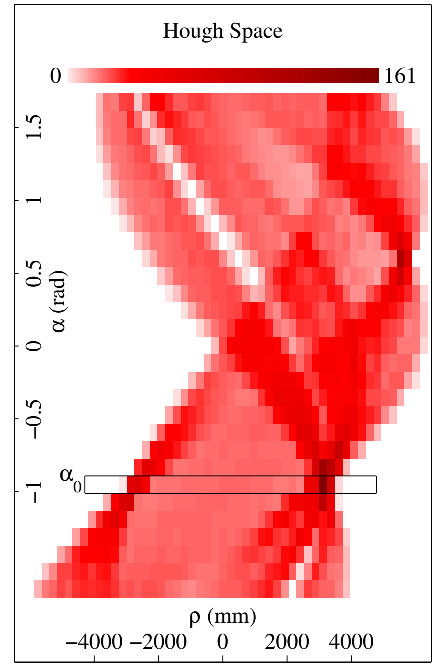
$$
\begin{aligned}
P_{\alpha\alpha} &= (D_\alpha)^2 + P^S_{\alpha\alpha} \\
P_{\rho_a\rho_a} &= (\sigma_\rho)^2 + P^S_{\rho\rho} \\
P_{\rho_b\rho_b} &= (\sigma_\rho)^2 + P^S_{\rho\rho} \\
P_{\psi_a\psi_a} &= (\sigma_\psi)^2 + P^S_{\psi_a\psi_a} \\
P_{\psi_b\psi_b} &= (\sigma_\psi)^2 + P^S_{\psi_b\psi_b} \\
&= scale + noise
\end{aligned}
$$



Efficiency in Multi-scale Extraction

- **Sub-sampling** - At coarse scales the Hough space bin size can be increased
- **Prior estimation** - A prior guess can limit Hough space bounds
- **Reuse** - Hough space calculations can be reused at multiple scales

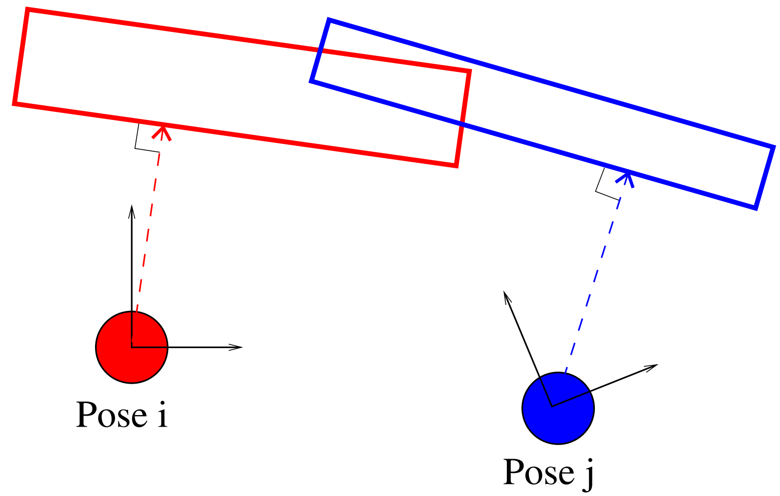




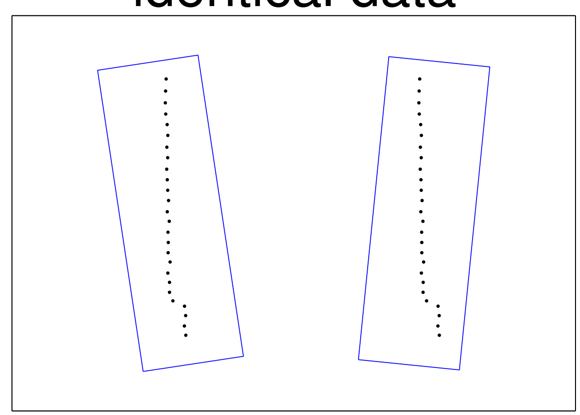
Block Feature Correspondence

Two groups of hypotheses are considered:

- Overlap Hypotheses
  - Takes scale based differences into account
  - Allows for rough matches at coarse scales
- Matching Hypotheses
  - Considers block border correspondence
  - Only takes parameter uncertainty into account



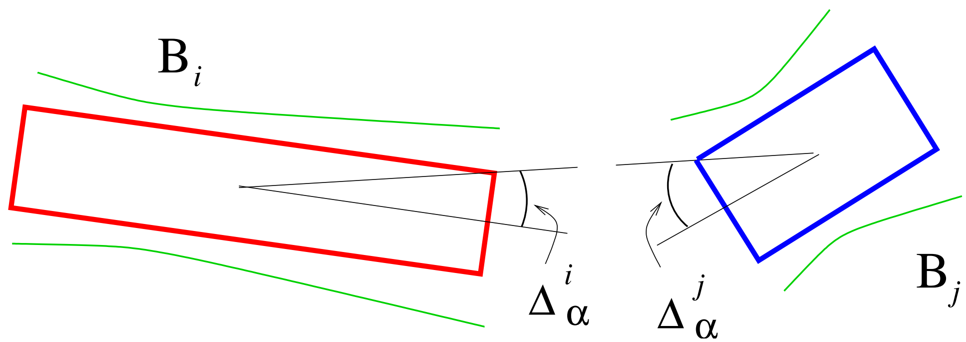
Different representation of identical data



Overlap Hypotheses

Can the blocks be describing the same underlying contour?

Orientation Overlap Test:



$$\Delta^i_\alpha = \tan^{-1}\left(\frac{\rho^i_b - \rho^i_a}{\psi^i_b - \psi^i_a}\right), \quad \Delta^j_\alpha = \tan^{-1}\left(\frac{\rho^j_b - \rho^j_a}{\psi^j_b - \psi^j_a}\right)$$

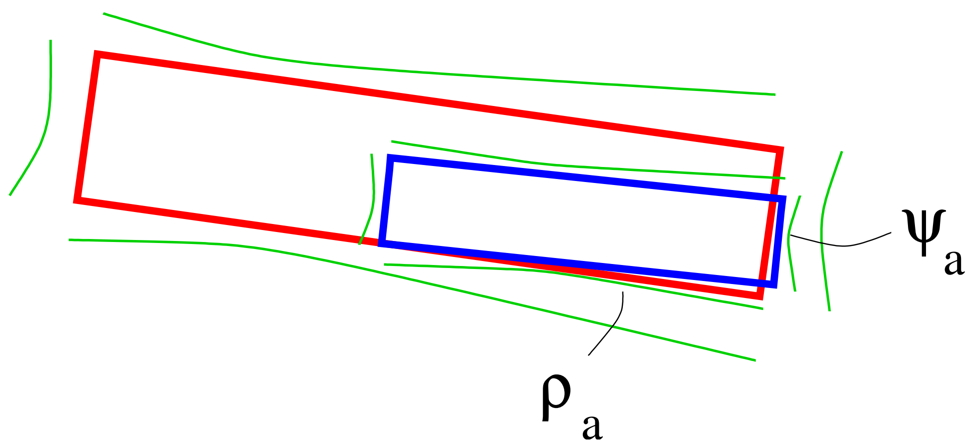
$$if \quad |\alpha^i - \alpha^j| \leq \Delta^i_\alpha + \Delta^j_\alpha \;\; then \;\; D^2 = 0$$

$$if \quad |\alpha^i - \alpha^j| > \Delta^i_\alpha + \Delta^j_\alpha \;\; then \;\; D^2 = \frac{(|\alpha^i - \alpha^j| - \Delta^i_\alpha + \Delta^j_\alpha)^2}{P^i_{\alpha\alpha} + P^j_{\alpha\alpha}}$$

Tests along block width and length dimensions are similarly formulated

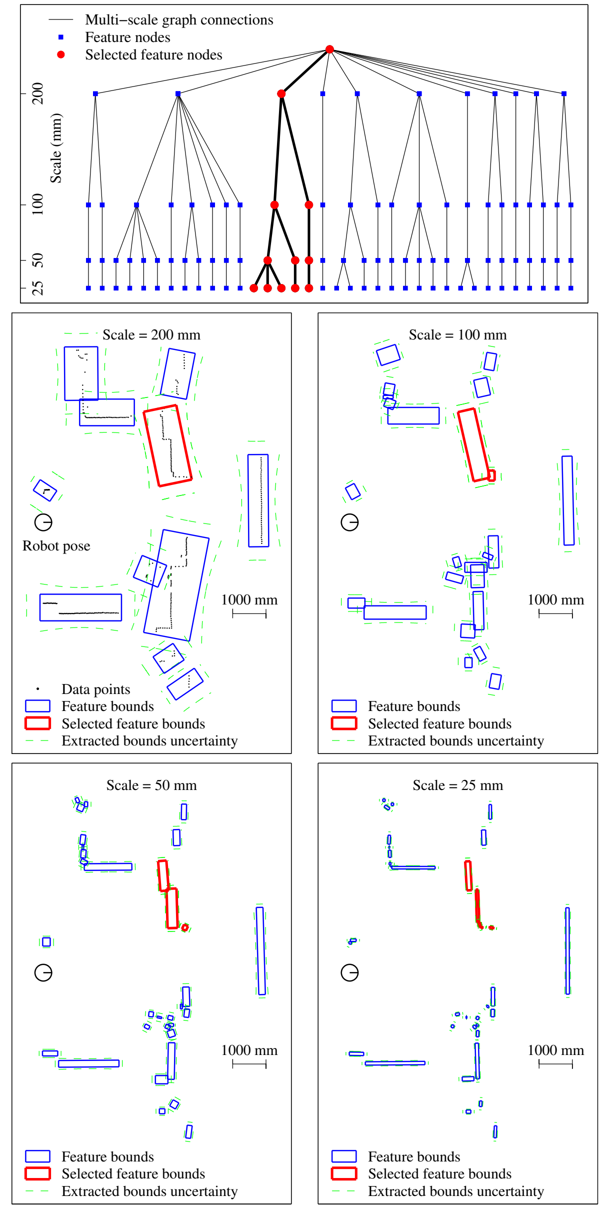
Match Hypotheses

- Chi-squared tests are developed to determine block boundary matches
- Boundary match tests are analogous to line segment matches
- Partial matches can occur
- Matching boundary elements are used to merge and localize



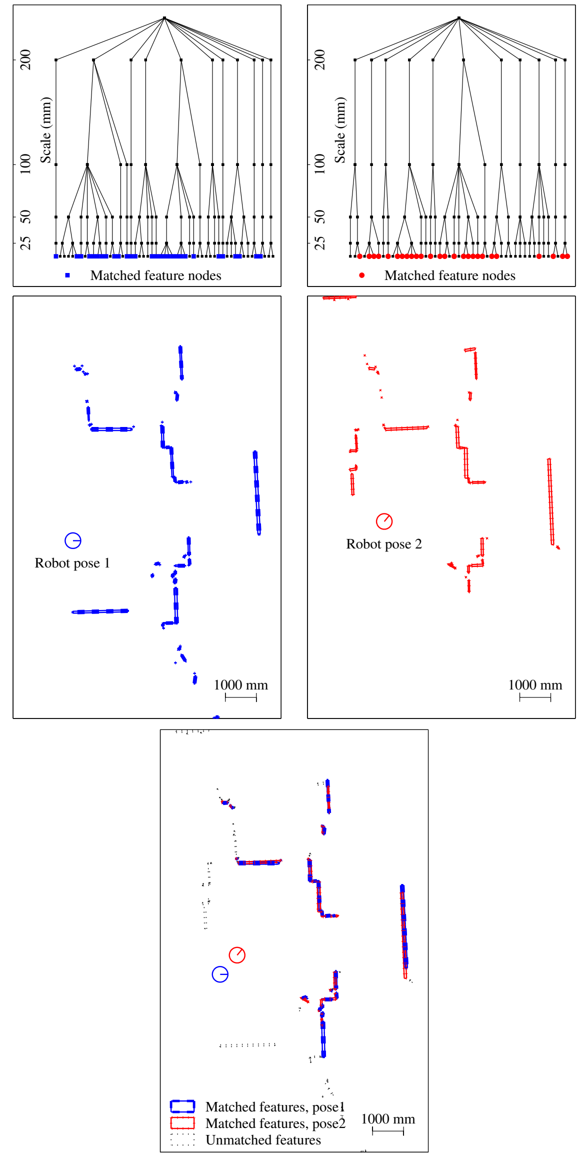
Scale Tree Construction:

- Bottom up approach
  - Benefits in Hough space reuse similar to Gaussian scale tree
- Top down approach
  - Separates data for computation at finer scales
  - Allows for partial construction of tree as needed



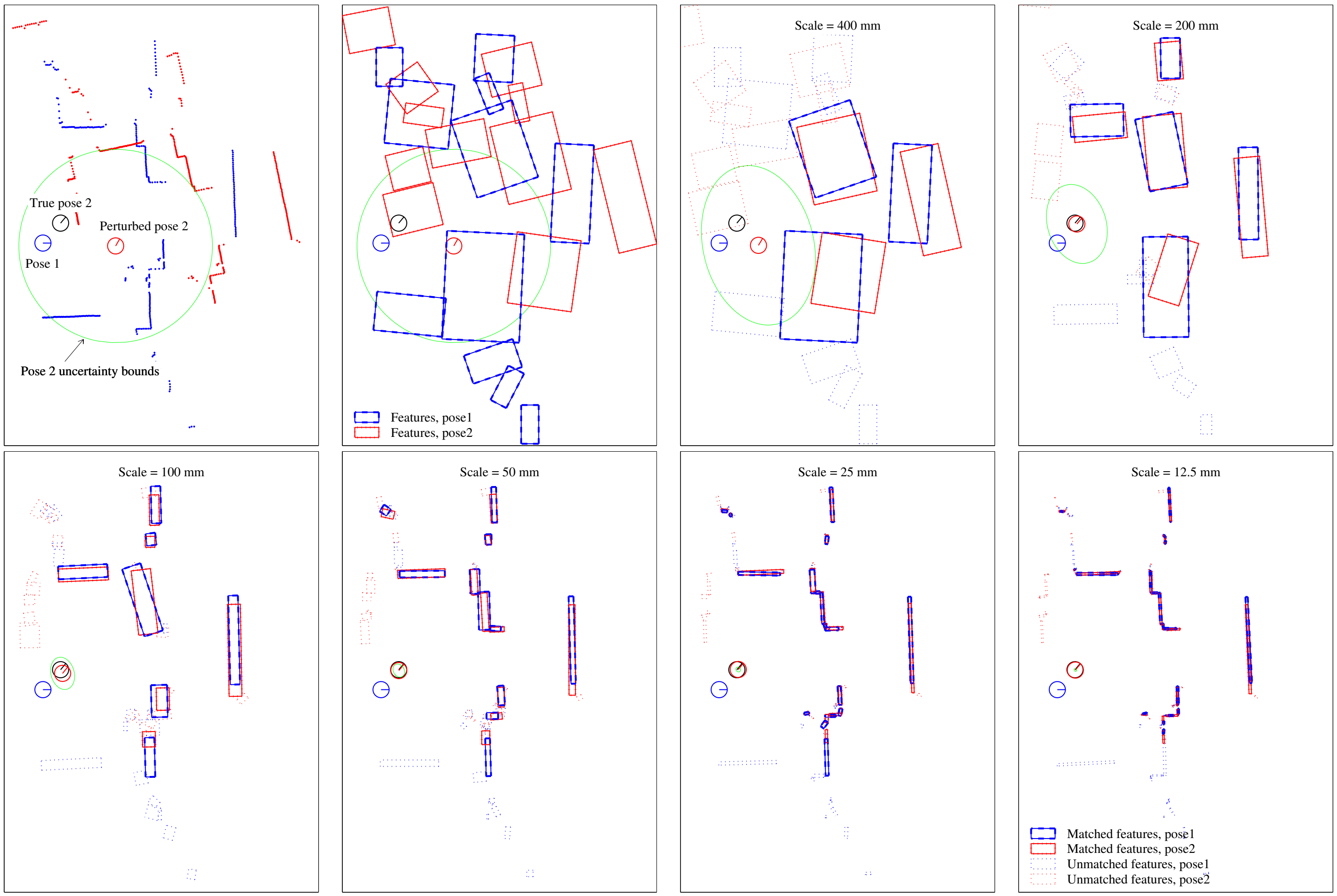
Tree Based Correspondence

- Matches are established across scales descending from coarse to fine
- Finer scale feature match search is guided by coarser matches
- Match search scales linearly with the number of features
- Experimental results vs. single scale:
  - 100 overlapping pairs considered
  - Average of 4-fold decrease in computation time



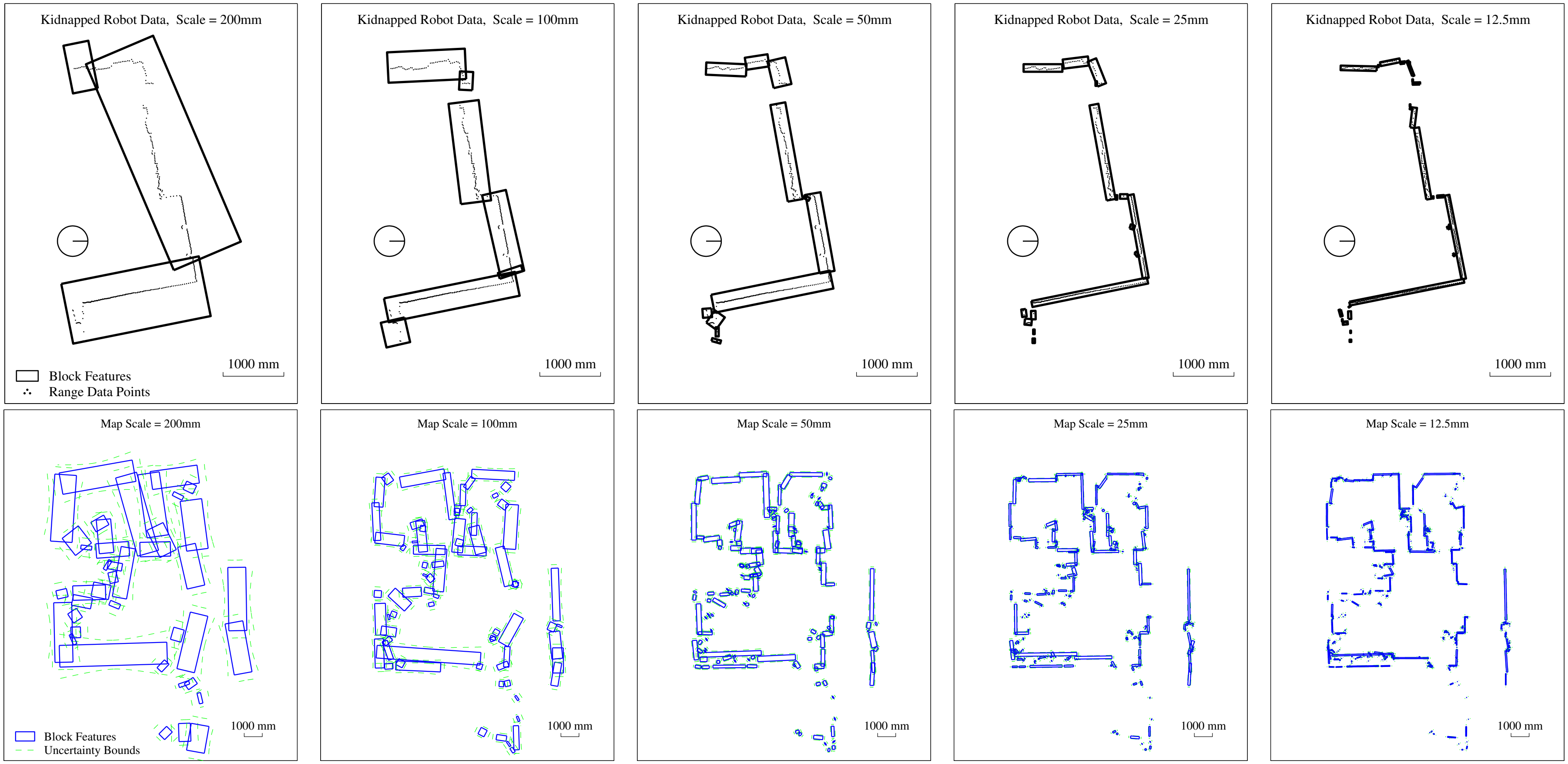
Tree Based Localization

- Matches are established across scales descending from coarse to fine
- An updated displacement estimate is calculated at each scale
- Experimental results show improved robustness to initial error



The Kidnapped Robot Problem

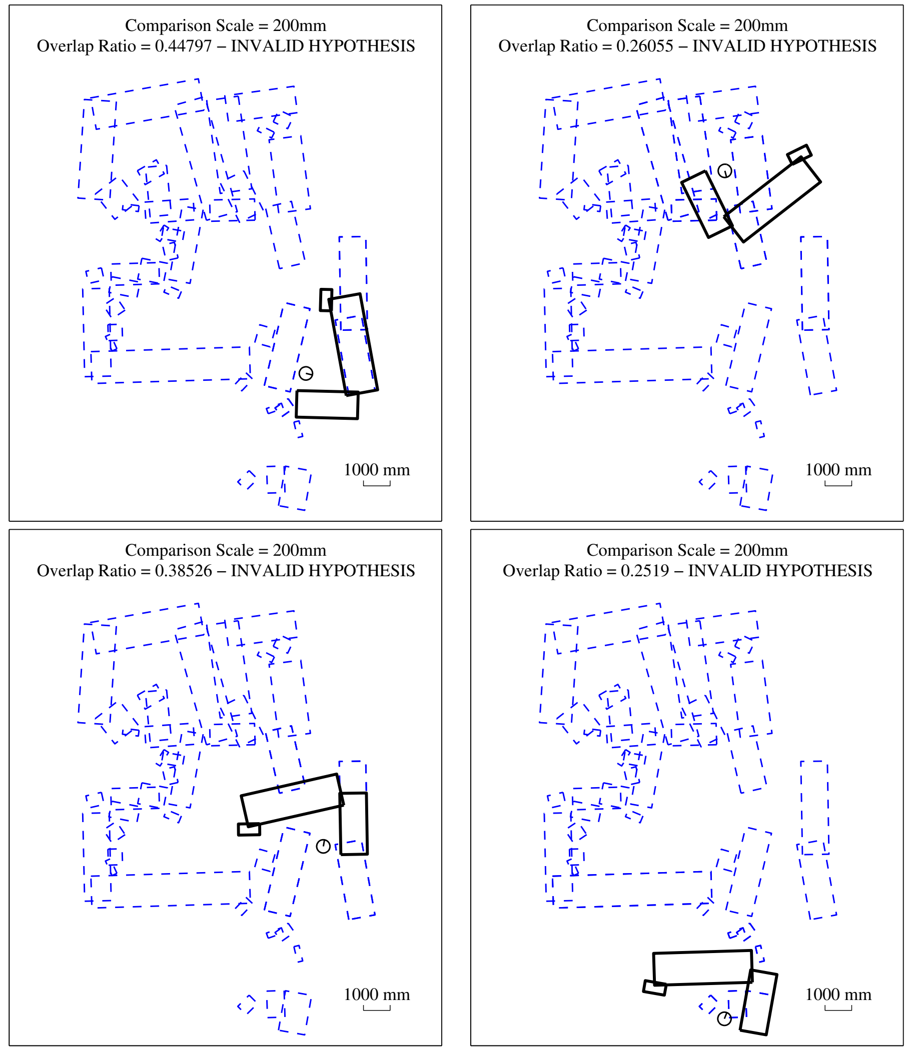
- Consider a large, unmodeled localization error
- The robot has no prior position knowledge
- Goal: Relocalize the robot, or determine it is in a new region



The Kidnapped Robot Problem

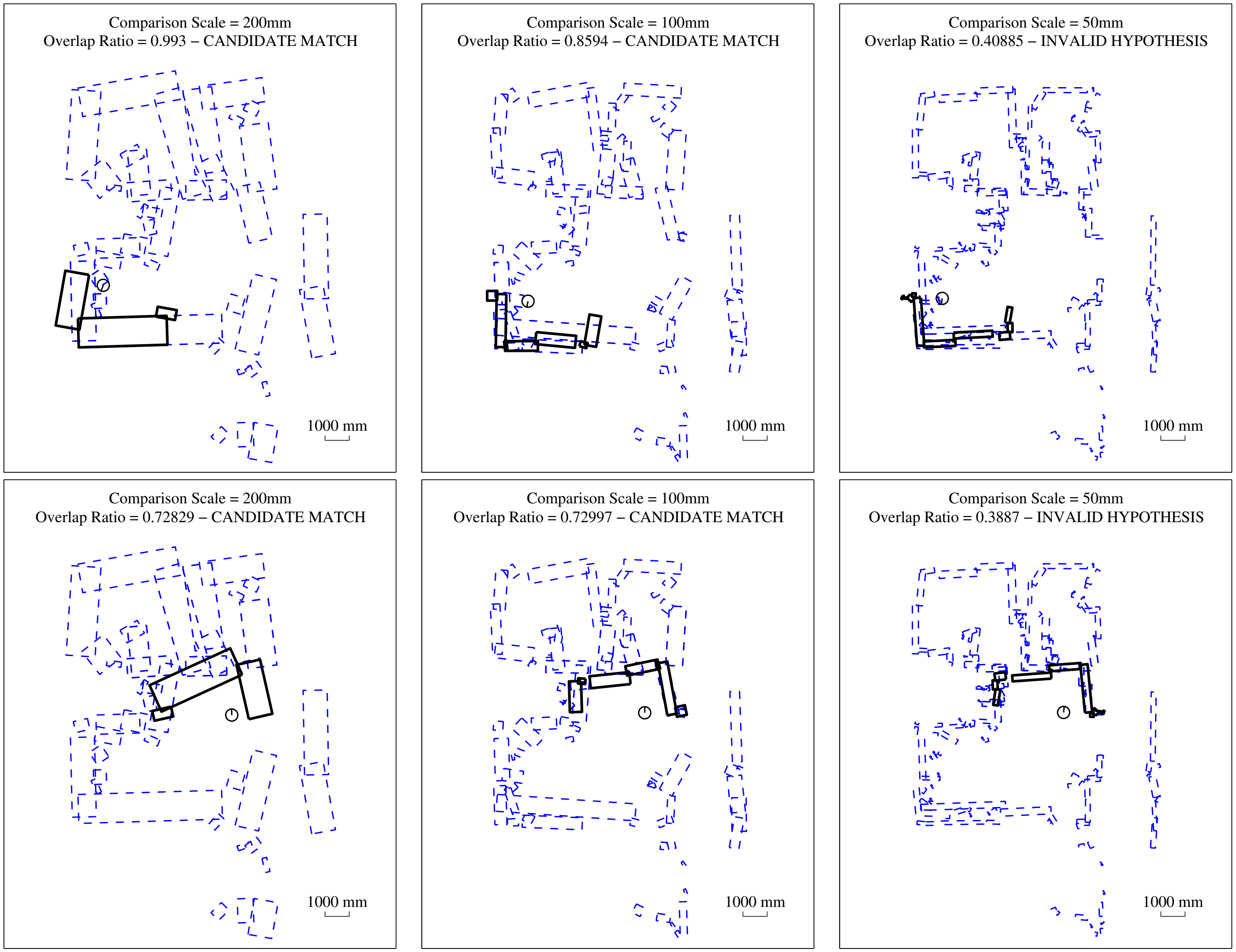
- Generate hypotheses by aligning two features
- Test hypothesis by computing percentage of feature overlap
- If more than 50% overlap, check at a finer scale
- If less than 50% overlap, invalidate hypothesis

Examples of invalidated hypotheses

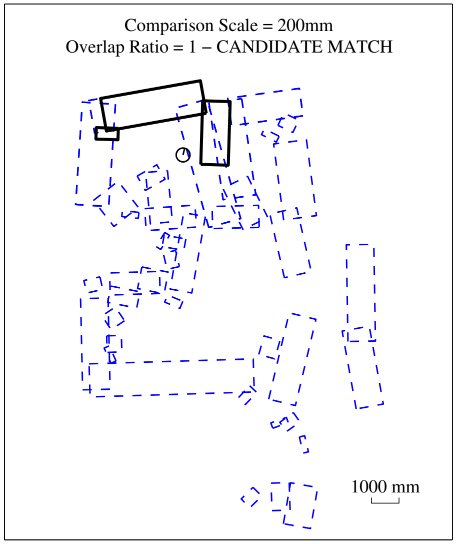


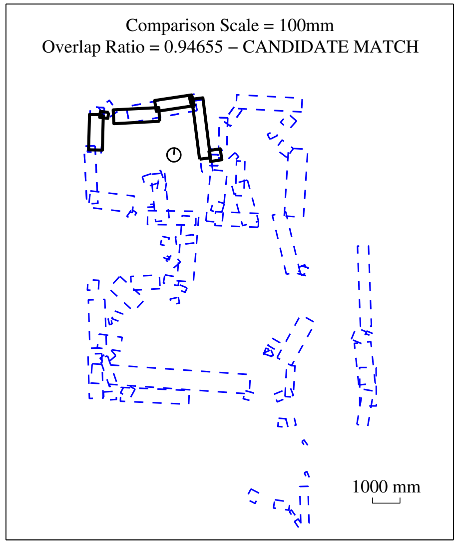
The Kidnapped Robot Problem

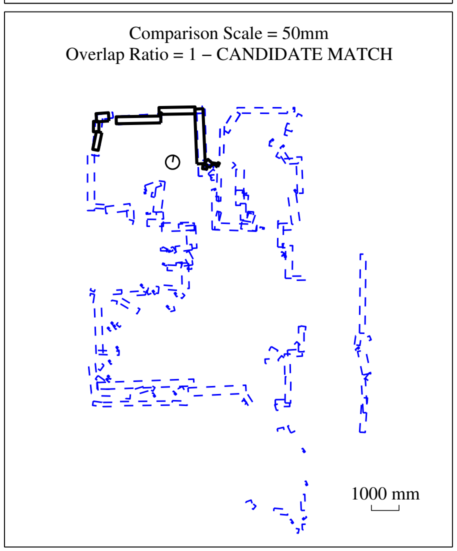
Examples of hypotheses invalidated at a finer scale

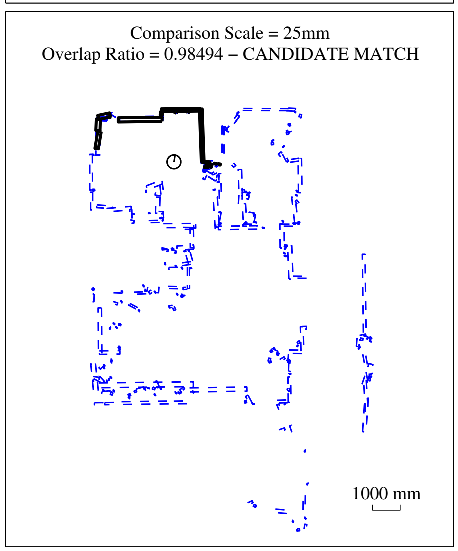


The Kidnapped Robot Problem

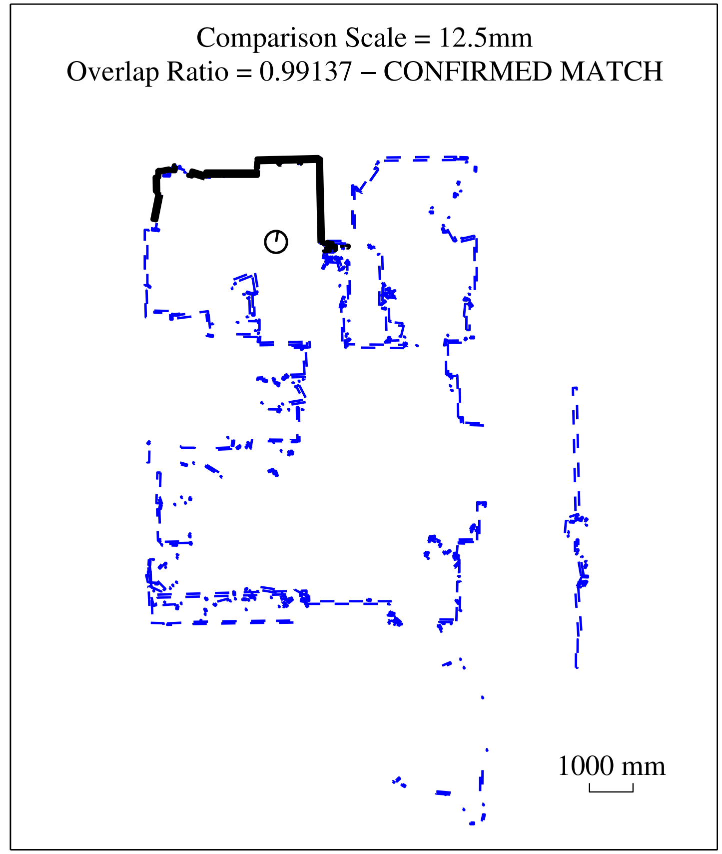








Confirmed unique solution



The Kidnapped Robot Problem

Averages over 50 runs at different positions in the map

- Multi-scale results
  - 2.74 seconds to first solution
  - 9.65 seconds for exhaustive search
- Single-scale results
  - 25.3 seconds to first solution
  - 40 minutes for exhaustive search

Averages over 30 runs from positions not in the map

- Multi-scale results
  - 8.3 seconds for full search and no found hypotheses
- Single-scale results
  - Over 30 minutes per run

Multi-scale Approach Conclusions

Contributions:

- A method of multi-scale feature extraction with individual point noise modeling
- An effective approach to multi-scale feature correspondence
- A more flexible feature :
  - Allows for representation of arbitrary data distribution
  - Allows for comparison of line-like and point-like features
- A multi-scale tree structure for efficient data comparison

Results:

- Maintains the high accuracy of line-segment methods
- more robust to error unstructured environment
- More efficient computation of feature correspondence
- More efficient solution of kidnapped robot problem

Future Work

- Further explore of scale-tree efficiency
  - Focused feature extraction through partial tree construction
- Apply and test features in unstructured outdoor environments
- Develop rigorous multi-scale Kalman filter based SLAM
- Extend algorithms for 3-D mapping

Acknowledgments

- Joel Burdick
- Stergios Roumeliotis
- Kristo Kriechbaum
- Robotics group
- Darpa Team Caltech
- JPL Mars rover crew
- Dad, Mom, Ben, Eliza and Frank
- My wife, Heidi

Acknowledgments

- Joel Burdick
- Stergios Roumeliotis
- Kristo Kriechbaum
- Robotics group
- Darpa Team Caltech
- JPL Mars rover crew
- Dad, Mom, Ben, Eliza and Frank
- My wife, Heidi

Conclusion : Contributions

Weighted Scan Matching Approach

- A method of point correspondence error compensation through modeling
- A general approach to incorporate uncertainty into scan matching

Line Segment Feature Based Approach

- An improved method of line feature extraction
- An effective approach to feature correspondence
- Improved compensation for non-linear effects
- A lossless line feature based approach

Multi-scale Feature Based Approach

- A method of multi-scale feature extraction
- An effective approach to multi-scale feature correspondence
- A multi-scale tree structure for efficient data comparison