Department of Electrical
and
Computer Systems Engineering

Technical Report
MECSE-29-2005

Scan Matching in Polar Coordinates with Application to SLAM

A. Diosi and L. Kleeman

MONASH
UNIVERSITY

Scan Matching in Polar Coordinates with Application to SLAM

Albert Diosi and Lindsay Kleeman
ARC Centre for Perceptive and Intelligent Machines
in Complex Environments
Department of Electrical and Computer Systems Engineering
Monash University, Clayton, VIC 3168, Australia
{Albert.Diosi;Lindsay.Kleeman}@eng.monash.edu.au

July 22, 2005

Abstract

This report presents a novel method for 2D laser scan matching called Polar Scan Matching (PSM). The method belongs to the family of point to point matching approaches. Our method avoids searching for point associations by simply matching points with the same bearing. This association rule enables the construction of an algorithm faster than the iterative closest point (ICP).

Firstly the PSM approach is tested with simulated laser scans. Then the accuracy of our matching algorithm is evaluated from real laser scans from known relative positions to establish a ground truth. Furthermore, to demonstrate the practical usability of the new PSM approach, experimental results from a Kalman filter implementation of simultaneous localization and mapping (SLAM) are provided.

Contents

1 Introduction **3**

2 Scan Preprocessing **5**
2.1 Median Filtering 7
2.2 Long Range Measurements 9
2.3 Segmentation 9
2.4 Motion Tracking 10
2.5 Current Scan Pose in Reference Scan Coordinate Frame 11

3 Scan Matching **12**
3.1 Scan Projection 13
3.2 Translation Estimation 17
3.2.1 Pose Estimation in Cartesian Frame 21
3.3 Orientation Estimation 22

4 Error Estimation **24**
4.1 Covariance Estimate of Weighted Least Squares 24
4.2 Heuristic Covariance Estimation 26

5 SLAM using Polar Scan Matching **26**

6 Experimental Results **28**
6.1 Simple Implementation of ICP 29
6.2 Simulated Room 30
6.3 Ground Thruth Experiment 31
6.4 Convergence Map 42
6.5 SLAM 44

7 Extensions to 3D **48**

8 Discussion **48**

1 Introduction

Localization and map making is an important function of mobile robots. One possible way to assist with this functionality is to use laser scan matching. In laser scan matching, the position and orientation or pose of the current scan is sought with respect to a reference laser scan by adjusting the pose of the current scan until the best overlap with the reference scan is achieved. In the literature there are methods for 2D and 3D scan matching. This report restricts discussion to 2D laser scan matching.

Scan matching approaches can be local [Lu and Milios, 1997] or global [Tomono, 2004]. When performing local scan matching, two scans are matched while starting from an initial pose estimate. When performing global scan matching the current scan is aligned with respect to a map or a database of scans without the need to supply an initial pose estimate. Scan matching approaches also can be categorized based on their association method such as feature to feature, point to feature and point to point. In feature to feature matching approaches, features such as line segments [Gutmann, 2000], corners or range extrema [Lingemann *et al.*, 2004] are extracted from laser scans, and then matched. Such approaches interpret laser scans and require the presence of chosen features in the environment. In point to feature approaches, such as one of the earliest by Cox [1991], the points of a scan are matched to features such as lines. The line features can be part of a predefined map. Features can be more abstract as in [Biber and Straßer, 2003], where features are Gaussian distributions with their mean and variance calculated from scan points falling into cells of a grid. Point to point matching approaches such as the approach presented in this report, do not require the environment to be structured or contain predefined features.

Examples of point to point matching approaches are the following: iterative closest point (ICP), iterative matching range point (IMRP) and the popular iterative dual correspondence (IDC). Besl and Mac Kay [1992] proposed ICP, where for each point of the current scan, the point with the smallest Euclidean distance in the reference scan is selected. IMPR was proposed by Lu and Milios [1997], where corresponding points are selected by choosing a point which has the matching range from the center of the reference scan's coordinate system. IDC, also proposed by Lu and Milios [1997] combines ICP and IMRP by using the ICP to calculate translation and IMPR to calculate rotation. The mentioned point to point methods can find the correct pose of the current scan in one step provided the correct associations are chosen. Since the correct associations are unknown, several iterations are performed. Matching may not always converge to the correct pose, since they can get stuck in a local minima. Due to the applied association rules, matching points have to be searched across 2 scans, resulting in $O(n^2)$ com-

plexity[1], where n is the number of scan points. All three approaches operate in a Cartesian coordinate frame and therefore do not take advantage of the native polar coordinate system of a laser scan. However, as shown later in this report, a scan matching algorithm working in the polar coordinate system of a laser scanner can eliminate the search for corresponding points thereby achieving $O(n)$ computational complexity for translation estimation. $O(n)$ computational complexity is also achievable for orientation estimation if a limited orientation estimation accuracy is acceptable.

These point to point matching algorithms apply a so called projection filter [Gutmann, 2000] prior to matching. The objective of this filter is to remove those points from the reference and current scan not likely to have a corresponding point. The computational complexity of this filter is also $O(n^2)$.

There are other scan matching approaches such as the method of Weiss and Puttkamer [1995]. Here for both reference and current scans, an angle-histogram of the orientation of line segments connecting consecutive points is generated. The orientation of the current scan with respect to the reference scan is obtained by finding the phase with the maximum cross correlation of the 2 angle histograms. The translation is found similarly by calculating x and y histograms, and calculating cross correlations. In scan matching, not all approaches use only that information in a scan, which describes where objects are located. Thrun et al. [2000] in their scan matching method utilize the idea, that free space in a scan is unlikely to be occupied in future scans.

In scan matching another important task, apart from finding the current scans pose, is the estimation of the quality of the match. Lu and Milios [1997] calculate the uncertainty of the match results by assuming white Gaussian noise in the x,y coordinates of scan points. This implicitly assumes that correct associations are made that results in optimistic error estimates, especially in corridors. Bengtsson and Baerveldt in [2001] developed more realistic approaches. In their first approach the pose covariance matrix is estimated from the Hessian of the scan matching error function. In their second approach, the covariance matrix is estimated off-line by simulating current scans and matching them to the reference scan.

Mapping with scan matching has been done for example by minimizing an energy function [Lu, 1995], using a combination of maximum likelihood with posterior estimation [Thrun *et al.*, 2000], using local registration and global cor-

[1]Lu and Milios [1997] claim that the IDC is of $O(n)$ complexity if the search for corresponding points is restricted to a window consisting of a fixed number of points. However if the angular resolution of laser scans is increased then the use of such a search window results in a decrease in the performance of the IDC due to the shrinking size of the search window expressed in angles. Therefore as it is correctly pointed out in [Gutmann, 2000] the computational complexity of IDC is $O(n^2)$.

relation [Gutmann, 2000] and using FastSLAM [Hähnel *et al.*, 2003]. A Kalman filter implementation can be found in [Bosse *et al.*, 2004].

In this report the Polar Scan Matching (PSM) approach is described which works in the laser scanner's polar coordinate system, therefore taking advantage of the structure of the laser measurements by eliminating the search for corresponding points. It assumed that in the 2D laser measurements range readings are ordered by their bearings. Laser range measurements of current and reference scans are associated with each other using the matching bearing rule, which makes translation estimation of the PSM approach $O(n)$ complexity unlike IDC's $O(n^2)$. The orientation estimation's computational complexity is also $O(n)$ if limited accuracy is acceptable, otherwise $O(kn)$, where k is proportional to the number of range readings per unit angle i.e. to the angular resolution of the scan. Note that k is introduced to differentiate between increasing the number of scan points by increasing the field of view or the angular resolution of the laser range finder. An $O(mn)$ complexity scan projection algorithm working in polar coordinates is also described in this report. The variable m equals to one added to the maximum number of objects occluding each other in the current scan viewed from the reference scan's pose. However this projection filter is of $O(n)$ complexity if no occlusions occur in the scan, therefore being more efficient than that of [Gutmann, 2000].

The rest of this report is organized as follows; first scan preprocessing steps, followed by the PSM algorithm is described. A heuristic scan match error model is presented next followed by a Kalman filter SLAM implementation utilizing our scan matching approach. Details of experimental results follow that include simulation, ground truth measurements and an implementation of SLAM. SLAM results with PSM are compared with results from SLAM using laser range finder and advanced sonar arrays [Diosi *et al.*, 2005; Diosi and Kleeman, 2004]. Finally conclusions and future work are presented.

Part of the work presented in this report is published as [Diosi and Kleeman, 2005].

2 Scan Preprocessing

Prior to matching, the current and the reference scans are preprocessed. Preprocessing helps to remove erroneous measurements, clutter or to group measurements of the same object to increase the accuracy and robustness of scan matching. In fig. 1 a laser scan is depicted in a Cartesian coordinate system. Corresponding raw range measurements are shown in fig. 2. Laser scans can have points which are not suitable matching. Such points are:

- Points representing moving objects such as the legs of a person in fig 1. Table and chair legs are also such points, since they are less likely to be

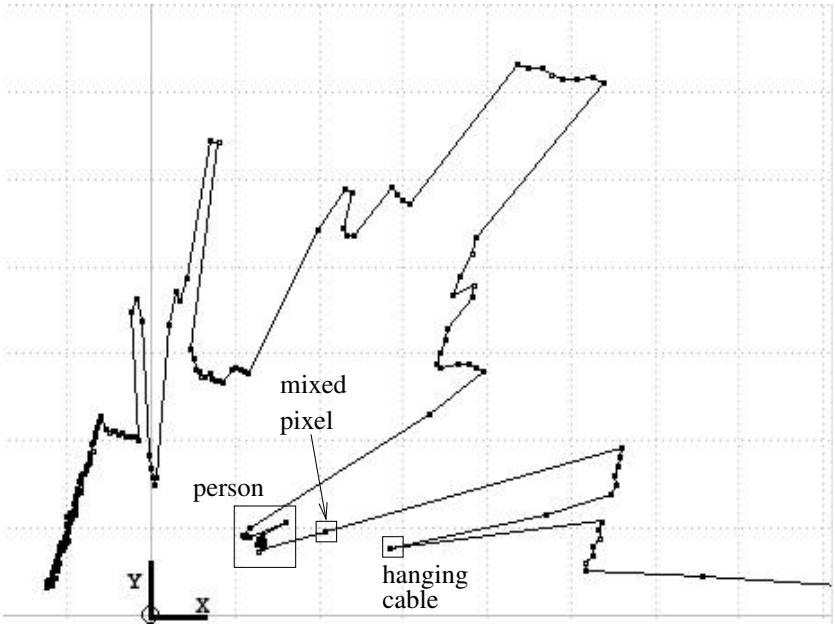


Figure 1: Laser scan in a Cartesian coordinate frame. Grid is 1m.

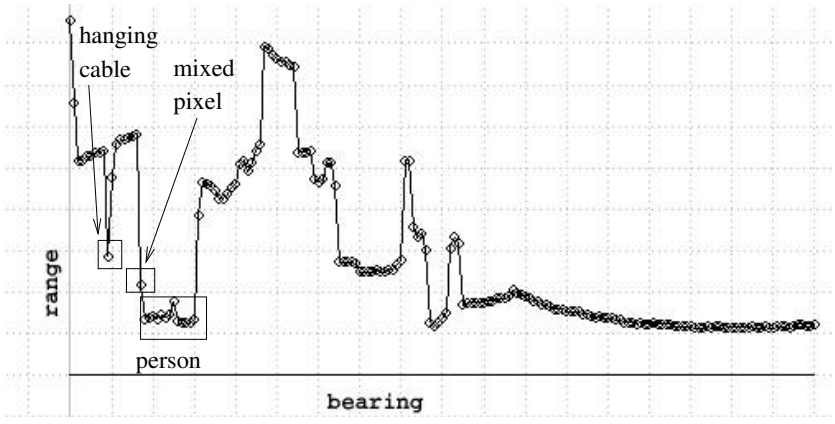


Figure 2: Scan of fig. 1 in the laser's polar coordinate frame. Horizontal grid size is 10°, vertical grid size is 1m.

static in the long term.

- Mixed pixels. At range discontinuities laser scanners often generate measurements which are located in the free space between two objects [Ye and Borenstein, 2002].

- Measurements with maximum range. Such readings are returned, when there is no object withing the range of the scanner. Also some surfaces (for example clean clear glass) do not illuminate well and show a laser spot, therefore they can appear as measurements with maximum range.

Instead of only removing range readings which are out of the range of the sensor, it was found useful to artificially restrict the sensor to a distance

$$PM_MAX_RANGE = 10m \tag{1}$$

and disregard (tag) any more distant readings. When having a sensor with 1° angular resolution, the minimum distance between two readings at 10 meters is 17cm. A large distance between neighboring points complicates the segmentation of scans, since if the distance between measured points is large it is hard to decide if the points belong to the same object. Interpolating between two neighboring points belonging to 2 different objects can be a source of error. In addition by artificially restricting the range of the sensor, difficulties may be introduced in large rooms with a lot of open space.

In the following subsections we will focus on how to exclude these unwanted points from the scan matching process.

2.1 Median Filtering

Median filters are used to replace outliers with suitable measurements [Gutmann, 2000]. After the application of a median filter to the range readings, objects such as chair and table legs are likely to be removed. Similarly to [Gutmann, 2000], a window size of

$$PM_MEDIAN_WINDOW = 5 \tag{2}$$

for the median filter was found satisfactory since it can replace at most 2 neighborings outliers. Mixed pixels are unlikely to be replaced by a median filter. The result after applying median filter to the scan of fig. 1 can be seen in figures 3 and 4. From fig. 4 it is clear, that at least 4 spikes were replaced, but all the mixed pixels still remain.

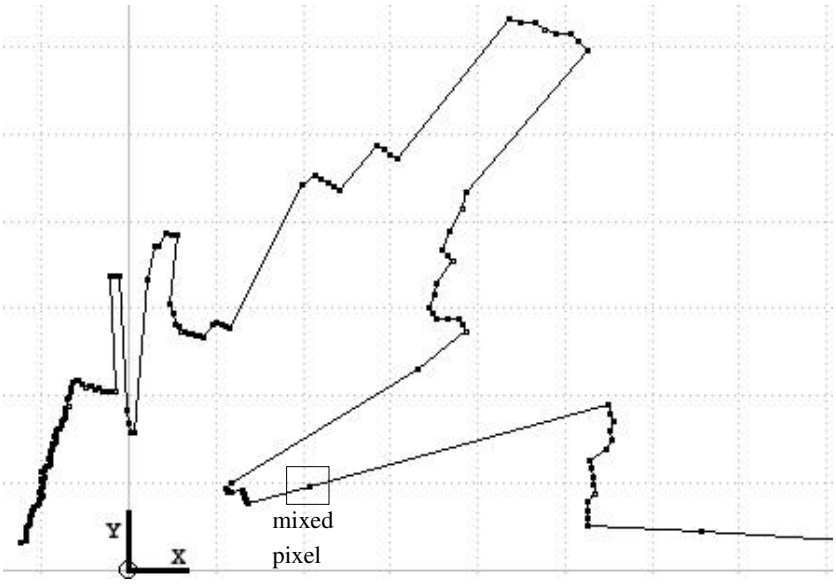


Figure 3: Laser scan of fig. 1 in a Cartesian coordinate frame after median filtering. Grid is 1m.

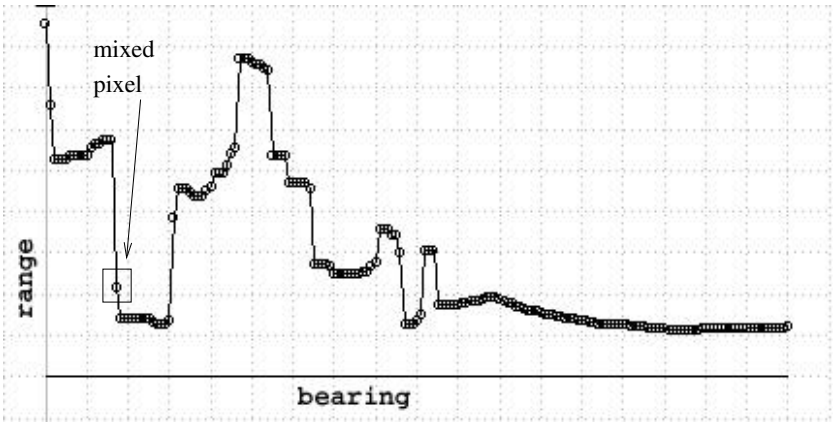


Figure 4: Scan of fig. 1 in the laser's polar coordinate frame after median filtering. Horizontal grid size is 10°, vertical grid size is 1m.

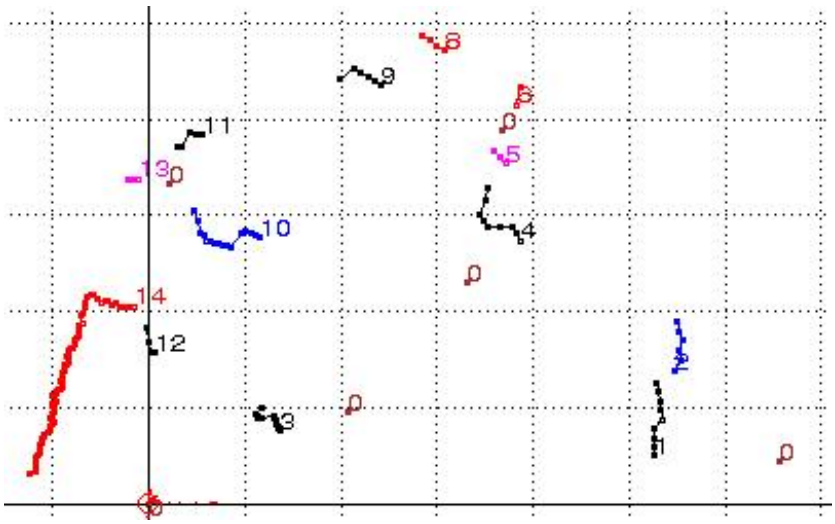


Figure 5: Laser scan of fig. 3 in a Cartesian coordinate frame after segmentation. Grid is 1m. Segments are assigned numbers. 0 is assigned to segments having only one point.

2.2 Long Range Measurements

After the application of a median filter all points further than a threshold *PM_MAX_RANGE* are tagged. These tagged points are used only in segmentation described next and not in scan matching. Range measurements larger than a *PM_MAX_RANGE* are not used in the scan matching because distance between such measurements is large, which makes it hard to decide if they belong to the same object or not.

2.3 Segmentation

Segmenting range measurements can have two advantages. The first advantage is that interpolation between 2 separate objects can be avoided if one knows that the objects are separate. Such interpolation is useful when one wants to know how a scan would look from a different location (scan projection). The second advantage is that if laser scans are segmented and the segments are tracked in consecutive scans then certain types of moving objects can be identified. Tracking moving objects can make scan matching more robust.

Two criteria are used in the segmentation process. According to the first criterion, a range reading, not differing more than

$$PM_MAX_DIFF = 20cm \tag{3}$$

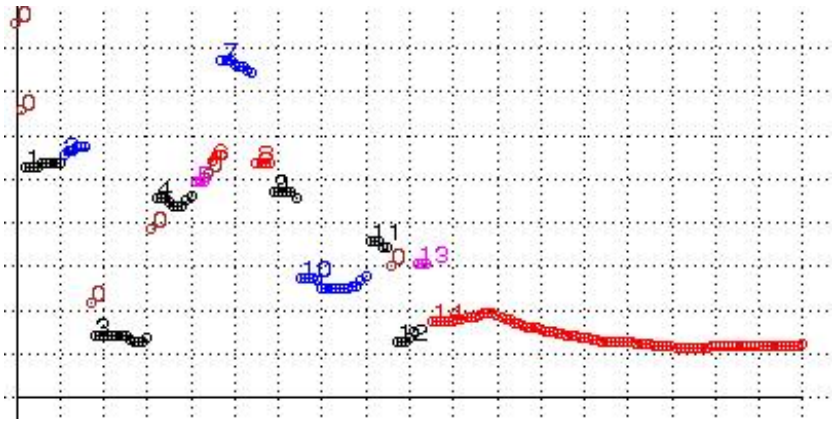


Figure 6: Scan of fig. 3 in the laser's polar coordinate frame after segmentation. Horizontal grid size is 10°, vertical grid size is 1m. Segments are assigned numbers. 0 is assigned to segments having only one point.

from the previous range reading, belongs to the same segment. This criterion fails to correctly segment out points which are for example on a wall oriented towards the laser. Therefore a second criterion is also applied according to which if 3 consecutive range readings lie approximately on the same polar line, then they belong to the same segment. Note that a mixed pixel can only then connect two objects if the distance between first object and mixed pixel and second object and mixed pixel is less than PM_MAX_DIFF. Tagged range readings also break segments.

Segmentation results can be seen in fig. 5 and 6. Different segments are assigned different numbers, except 0, which is assigned to segments consisting of only one point. Segments assigned 0 are also tagged, therefore they are not used in the scan matching process. Note that most of the mixed pixels get assigned 0.

Note that this simple segmentation is of $O(n)$ complexity.

2.4 Motion Tracking

One of the problems of point to point scan matching is that moving objects can cause wrong associations thus reducing the accuracy of the scan matching results, or can even cause divergence. It is therefore advisable to track and tag moving objects in laser scans. Motion tracking constitutes future work and is beyond the scope of this report.

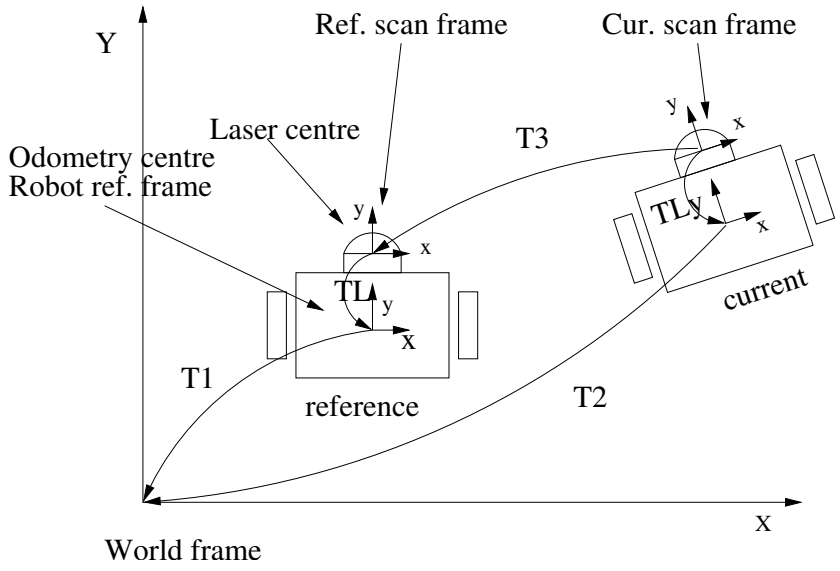


Figure 7: Reference and current coordinate frames.

2.5 Current Scan Pose in Reference Scan Coordinate Frame

To make scan matching simpler, let us set the goal of scan matching to find the position and orientation (or pose) of the current laser scan's coordinate frame with respect to the reference scans coordinate frame (see fig. 7). Otherwise the introduction of a world frame in which to relate the current and reference frame would make the equations describing the relation of current and reference scan points more complicated. The reference scans coordinate frame is the coordinate frame of the laser scanner at the reference location.

To find out the position and orientation of the current frame with respect to reference frame, from fig. 7 we can write:

$$\mathbf{T_1 T_L T_3} = \mathbf{T_2 T_L} \tag{4}$$

where T_1 is the homogeneous transformation matrix from robot frame at reference location to world frame, T_2 is the transformation from robot frame at current location to world frame, T_3 is the transformation from current laser scan frame to reference laser scan frame and T_L is the transformation from laser frame into robot frame. With the multiplication of the matrices it is assumed that coordinates of points are stored in column vectors therefore the transformations are applied from right to left. From (4), the transformation from current into reference frame is:

$$\mathbf{T_3} = \mathbf{T_L^{-1} T_1^{-1} T_2 T_L}. \tag{5}$$

If $(x_{rr}, y_{rr}, \theta_{rr})$ describes the robot's pose at the reference location expressed in world frame, $(x_{cr}, y_{cr}, \theta_{cr})$ describes the robot pose the current location expressed in world frame, (x_c, y_c, θ_c) describes the laser scanner's pose at the current location expressed in the reference frame, and if the laser scanners pose is described with (x_l, y_l, θ_l) in the robots frame, then:

$$\begin{array}{ll} T_3 = \begin{bmatrix} \cos\theta_c & -\sin\theta_c & x_c \\ \sin\theta_c & \cos\theta_c & y_c \\ 0 & 0 & 1 \end{bmatrix} & T_1 = \begin{bmatrix} \cos\theta_{rr} & -\sin\theta_{rr} & x_{rr} \\ \sin\theta_{rr} & \cos\theta_{rr} & y_{rr} \\ 0 & 0 & 1 \end{bmatrix} \\ T_L = \begin{bmatrix} \cos\theta_l & -\sin\theta_l & x_l \\ \sin\theta_l & \cos\theta_l & y_l \\ 0 & 0 & 1 \end{bmatrix} & T_2 = \begin{bmatrix} \cos\theta_{cr} & -\sin\theta_{cr} & x_{cr} \\ \sin\theta_{cr} & \cos\theta_{cr} & y_{cr} \\ 0 & 0 & 1 \end{bmatrix} \end{array} \quad (6)$$

By substituting (6) into (5) and comparing the left and right sides, the current pose (x_c, y_c, θ_c) expressed in the reference frame can be determined as:

$$\theta_c = \theta_{cr} - \theta_{rr} \quad (7)$$

$$x_c = x_l(\cos\beta - \cos\theta_l) + y_l(\sin\beta - \sin\theta_l) + (x_{cr} - x_{rr})\cos\gamma + (y_{cr} - y_{rr})\sin\gamma \quad (8)$$

$$y_c = -x_l(\sin\beta - \sin\theta_l) + y_l(\cos\beta - \cos\theta_l) - (x_{cr} - x_{rr})\sin\gamma + (y_{cr} - y_{rr})\cos\gamma \quad (9)$$

where $\beta = \theta_l + \theta_{rr} - \theta_{cr}$ and $\gamma = \theta_l + \theta_{rr}$.

3 Scan Matching

The laser scan matching method described next aligns the current scan with respect to the reference scan so that the sum of square range residuals is minimized. It is assumed that an initial pose of the current scan is given, expressed in the coordinate frame of the reference scan. The coordinate frame of a laser scan is centered at the point of rotation of the mirror of a laser scanner. The X axis or zero angle of the laser's Cartesian coordinate system coincides with the direction of the first reported range measurement. The current scan is described as $C = (x_c, y_c, \theta_c, \{r_{ci}, \phi_{ci}\}_{i=1}^n)$, where x_c, y_c, θ_c describe position and orientation, $\{r_{ci}, \phi_{ci}\}_{i=1}^n$ describe n range measurements r_{ci} at bearings ϕ_{ci}, expressed in the current scan's coordinate system. $\{r_{ci}, \phi_{ci}\}_{i=1}^n$ are ordered by the bearings in ascending order as they are received from a SICK laser scanner. The reference scan is described as $R = \{r_{ri}, \phi_{ri}\}_{i=1}^n$. Note that if bearings where range measurements are taken are unchanged in current and reference scans then $\phi_{ri} = \phi_{ci}$. The scan matching works as follows: after preprocessing the scans, scan projection followed by a translation estimation or orientation estimation are iterated. In

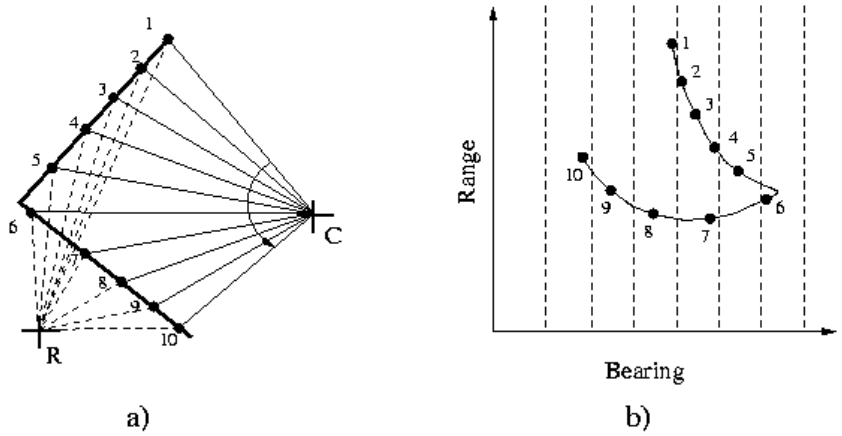


Figure 8: a) projection of measured points taken at C to location R. b) points projected to R shown in polar coordinates. Dashed lines represent bearings which the scanner would have sampled.

the polar scan matching (PSM) of this report, one orientation step is followed by one translation step. More details on these steps are given in the following sub sections.

3.1 Scan Projection

An important step in scan matching is finding out how the current scan would look if it were taken from the reference position. For example in fig. 8, the current scan is taken at location C and the reference scan is taken at position R. The range and bearings of the points from point R (see fig. 8b) are calculated:

$$r'_{ci} = \sqrt{(r_{ci}\cos(\theta_c+\phi_{ci})+x_c)^2+(r_{ci}\sin(\theta_c+\phi_{ci})+y_c)^2} \quad (10)$$

$$\phi'_{ci} = atan2(r_{ci}\sin(\theta_c+\phi_{ci})+y_c,\ r_{ci}\cos(\theta_c+\phi_{ci})+x_c) \quad (11)$$

where $atan2$ is the four quadrant version of arctan.

In fig. 8b the dashed vertical lines represent sampling bearings (ϕ_{ri}) of the laser at position R in fig. 8a. Since the association rule is to match bearings of points, next ranges r''_{ci} at the reference scan bearings ϕ_{ri} are calculated using interpolation. The aim is to estimate what the laser scanner would measure from pose R. This re-sampling step consists of checking (r'_{ci}, ϕ'_{ci}) (i.e. 1,2,..10 in fig. 8b) of each segment if there are one or more sample bearings between 2 consecutive points (i.e.

between 1 and 2 there is one, between 6 and 7 there are 2). By linear interpolation a range value is calculated for each sample bearing. If a range value is smaller than an already stored range value at the same bearing, then the stored range is overwritten with the new one to handle occlusion. As in [Lu and Milios, 1997] a new range value is tagged as invisible if the bearings of the 2 segment points are in decreasing order.

A pseudo code implementation of the described scan projection is shown in fig. 9. Note that unlike the equations in this report, the indexes of vector elements in fig. 9 start from 0. Also note that this particular implementation assumes that laser scans have 1° bearing resolution. However this assumption is used only in the transformation of bearings from radians to indexes into range arrays and can be easy changed to work with scans of arbitrary resolution. The pseudo code on lines 00-09 transforms the current scan readings (ϕ_i, r_{ci}) into the reference scan's coordinate frame, while using the current frame pose (x_c, y_c, θ_c) expressed in the reference frame. Since the projected current scan (ϕ'_{ci}, r'_{ci}) is re-sampled next at the sample bearings ϕ_i of the reference scan, the data structures associated with the re-sampled current scan are also initialized. Status registers $tagged''_{ci}$ contain flags describing if re-sampled range readings r''_{ci} have been tagged or if they contain a range reading. All flags of the status registers are cleared except the flag PM_EMPTY which indicates that no range reading has been re-sampled into the particular position of the range array r''_c. Re-sampled current scan range readings r''_{ci} are set to a value which is larger than the maximum range of the laser scanner.

The re-sampling of the projected current scan readings (ϕ'_{ci}, r'_{ci}) takes place on lines 09-43 in a loop which goes through neighboring pairs of (ϕ'_{ci}, r'_{ci}). Pairs of measurements are only re-sampled if they belong to the same segment and none of them are tagged. Next, on lines 14-30 the measurement pair is checked if it is viewed from behind by testing if $\phi'_{ci} > \phi'_{ci-1}$. Then depending on their order, ϕ'_{ci} and ϕ'_{ci-1} are converted into indexes ϕ_0, ϕ_1 into the re-sampled ranges array, so that $\phi_0 <= \phi_1$. This is done to simplify the following interpolation step where the re-sampled ranges r''_c are calculated in a while loop (lines 31-43) at index ϕ_0 which is incremented until it reaches ϕ_1. In the while loop first range r corresponding to ϕ_0 is calculated using linear interpolation. Then if ϕ_0 is within the bounds of the array r''_c and if r is smaller than the value already stored at $r''_{c\phi_0}$ then the empty flag of $tagged''_{c\phi_0}$ is cleared and $r''_{c\phi_0}$ is overwritten by r. This last step filters out those projected current scan readings which are occluded by other parts of the current scan. Finally the occluded flag of $tagged''_{c\phi_0}$ is cleared or set, depending on if ϕ'_{ci} was greater than ϕ'_{ci-1}, and ϕ_0 is incremented.

The body of the while loop (lines 32-40) of the pseudo code is executed at most $2n$ times for scans with no occlusion, where n is the number of points. However

/**********Scan Projection**********/
00 //*Transform current measurements into reference frame*
01 for i = 0 $\rightarrow$ number_of_points-1 {
02 $x = r_{ci}\cos(\theta_c + \phi_i) + x_c$
03 $y = r_{ci}\sin(\theta_c + \phi_i) + y_c$
04 $r'_{ci} = \sqrt{x^2 + y^2}$
05 $\phi'_{ci} = atan2(y,x)$
06 $tagged''_{ci} = PM_EMPTY$
07 $r''_{ci} = LARGE_VALUE$
08 }
09 //*Given the projected measurements* (r'_{ci}, ϕ'_{ci})*, calculate what would have been*
10 *measured with the laser scanner at the reference pose.*
11 for i = 1 $\rightarrow$ number_of_points-1 {
12 if $segment_{ci} \neq 0$ & $segment_{ci} = segment_{ci-1}$
13 $!tagged_{ci}$ & $!tagged_{ci-1}$ & $\phi'_{ci} > 0$ & $\phi'_{ci-1} > 0$ {
14 if $\phi'_{ci} > \phi'_{ci-1}$ { //*Is it visible?*
15 $occluded = false$
16 $a_0 = \phi'_{ci-1}$
17 $a_1 = \phi'_{ci}$
18 $\phi_0 = ceil(\phi'_{ci-1}\frac{180}{\pi})$
19 $\phi_1 = floor(\phi'_{ci}\frac{180}{\pi})$
20 $r_0 = r'_{ci-1}$
21 $r_1 = r_{ci}$
22 }else{
23 $occluded = true$
24 $a_0 = \phi'_{ci}$
25 $a_1 = \phi'_{ci-1}$
26 $\phi_0 = ceil(\phi'_{ci}\frac{180}{\pi})$
27 $\phi_1 = floor(\phi'_{ci-1}\frac{180}{\pi})$
28 $r_0 = r'_{ci}$
29 $r_1 = r_{ci-1}$
30 }
31 while $\phi_0 \leq \phi_1$ {
32 $r = \frac{r_1 - r_0}{a1 - a0}(\phi_0\frac{\pi}{180} - a_0) + r_0$
33 if$\phi_0 \neq 0$ & $\phi_0 < number_of_points$ & $r''_{c\phi_0} > r$ {
34 $r''_{c\phi_0} = r$
35 $tagged''_{c\phi_0}$& $=\sim PM_EMPTY$
36 if occluded
37 $tagged''_{c\phi_0}| = PM_OCCLUDED$
38 else
39 $tagged''_{c\phi_0}$& $=\sim PM_OCCLUDED$
40 $\phi_0 = \phi_0 + 1$
41 }//while
42 }//if
43 }//for

Figure 9: Scan projection pseudo code for 1° resolution.

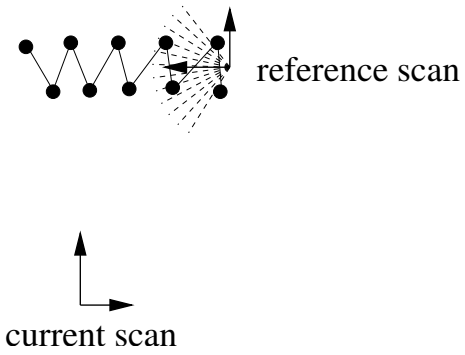


Figure 10: Example for the worst case scenario for scan projection.

it is easy to contrive a scenario where the inside of the while loop would execute at most n^2 times. For example fig. 10 depicts a situation where the noise in the current scan readings (drawn with connected circles) is large and the scan readings are aligned with the reference scan's frame so that most of the reference scan's laser beams go through in between the points of the current scan. In a such case for each pair of current scan points the while loop would execute almost n times resulting in a total number of executions between $2n$ and n^2. The computational complexity of this projection filter is $O(mn)$ where m is the maximum number of objects occluding each other in the current scan viewed from the reference scan's pose incremented by one. For example if there is no occlusion then m is 1. If there is at least one object which occludes another object, while the occluded object does not occlude any other object, then m is 2. If there are objects A,B and C where A occludes B and B occludes C then m is 3.

The scan projection filter described in [Gutmann, 2000] is of $O(n^2)$ complexity, because a double loop is employed to check for occlusion. That occlusion check consists of checking whether any current scan point in XY coordinates is obscured by any other pair of consecutive current or reference scan points. Since the scan projection implementation in fig. 9 is of $O(n)$ complexity when there are no occlusions in the current scan, it is reasonable to believe that under normal circumstances it is more efficient than that described in [Gutmann, 2000]. Due to its efficiency the projection filter of fig. 9 is applied in each iteration of the PSM scan matching algorithm.

Note that the Cartesian projection filter in [Gutmann, 2000] removes all current scan points which are further than one meter from all reference scan points and vice versa. In PSM associated current and reference scan measurements with a residual larger than a preset threshold are ignored in the position estimation process and not in the projection filter. This eliminates the need for performing

the computationally expensive removal of points without correspondence in the projection filter.

3.2 Translation Estimation

After scan projection, for each bearing ϕ_{ri} there is at most one r''_{ci} from the projected current scan and a corresponding r_{ri} from the reference scan. The aim is to find (x_c, y_c) which minimizes $\sum w_i (r_{ri} - r''_{ci})^2$, where w_i is a weight used to reduce weighting[2] of bad matches. To minimize the weighted sum of square residuals linear regression was applied to the linearized eq. 10:

$$\Delta r_i \approx \frac{\partial r''_{ci}}{\partial x_c}\Delta x_c + \frac{\partial r''_{ci}}{\partial y_c}\Delta y_c = \cos(\phi_{ri})\Delta x_c + \sin(\phi_{ri})\Delta y_c \tag{12}$$

$\frac{\partial r''_{ci}}{\partial x_c} = \cos(\phi_{ri})$ has been derived from (10) the following way:

$$\begin{aligned} \frac{\partial r''_{ci}}{\partial x_c} &= \frac{1}{2}\frac{2(r_{cj}\cos(\theta_c+\phi_{cj})+x_c)}{\sqrt{(r_{cj}\cos(\theta_c+\phi_{cj})+x_c)^2+(r_{cj}\sin(\theta_c+\phi_{cj})+y_c)^2}} \\ &= \frac{(r_{cj}\cos(\theta_c+\phi_{cj})+x_c)}{r''_{ci}} = \frac{r''_{ci}\cos\phi_{ri}}{r''_{ci}} = \cos\phi_{ri} \end{aligned} \tag{13}$$

Where ϕ_{cj}, r_{cj} is a virtual, unprojected reading which would correspond to an uninterpolated ϕ_{ri}, r''_{ci}. The derivation of $\frac{\partial r''_{ci}}{\partial y_c}$ is analogous to the derivation of $\frac{\partial r''_{ci}}{\partial x_c}$.

If range differences between projected current range and reference range readings are modeled as

$$(\mathbf{r''_c} - \mathbf{r_r}) = \mathbf{H}\begin{bmatrix} \Delta x_c \\ \Delta y_c \end{bmatrix} + \mathbf{v} \tag{14}$$

where $\mathbf{v}$ is the noise vector and

$$\mathbf{H} = \begin{bmatrix} \frac{\partial r''_{c1}}{\partial x_c} & \frac{\partial r''_{c1}}{\partial y_c} \\ \frac{\partial r''_{c2}}{\partial x_c} & \frac{\partial r''_{c2}}{\partial y_c} \\ \dots & \dots \end{bmatrix}, \tag{15}$$

then position correction $\Delta x_c, \Delta y_c$ of the current scan is then calculated by minimizing the sum of weighted range residuals $\sum w_i (r_{ri} - r''_{ci})^2$ using the well known equation for weighted least squares [Kay, 1993]:

$$\begin{bmatrix} \Delta x_c \\ \Delta y_c \end{bmatrix} = (\mathbf{H^T W H})^{-1}\mathbf{H^T W}(\mathbf{r''_c} - \mathbf{r_r}) \tag{16}$$

[2]Note that there is also an implicit weighting of closer objects, since they cover a larger angle.

where $\mathbf{r}''_{\mathbf{c}}, \mathbf{r}_{\mathbf{r}}$ are vectors containing r''_{ci} and r_{ri} and $\mathbf{W}$ is a diagonal matrix of weights. The elements of $\mathbf{W}$ are calculated according to the recommendations of Dudek and Jenkin in [Dudek and Jenkin, 2000]:

$$w_i = 1 - \frac{d_i^m}{d_i^m + c^m} \tag{17}$$

where $d_i = r''_{ci} - r_{ri}$ is the error between projected current scan range measurements and reference scan range measurements and c is a constant. Equation (17) describes a sigmoid function with weight 1 at $d_i = 0$ and a small weight for large d_i. Parameter c determines where the sigmoid changes from 1 to 0, and m determines how quickly the sigmoid function changes from 1 to 0. In [Dudek and Jenkin, 2000] (17) was used to weight the distance of a laser scan point to a line in a point-to-feature scan matching method.

To reduce the effects of association errors in the implementation of equation (16), only those visible measurements are taken into consideration which are not tagged (see section 2). Also the errors between reference and current scan range measurements have to be smaller than a preset threshold *PM_MAX_ERROR* to be included.

An example implementation for one step of the translation estimation can be seen in fig. 11. In the implementation first $\mathbf{H^T W H}$ and $\mathbf{H^T W \Delta r}$ are calculated for untagged associated reference and current scan measurements, which are closer to each other than a threshold. Note that elements h1, h2 of the Jacobian matrix $\mathbf{H}$ on lines 05-06 have to be calculated only once, since ϕ_{ri} depends only on the type of laser scanner. Matrix $\mathbf{H^T W H}$ is inverted on lines 16-20 followed by the calculation of pose corrections. As one can see from fig. 11, translation estimation is of $O(n)$ complexity. The translation estimation step of IDC and ICP is of $O(n^2)$ complexity.

Note that the equation used in other point-to-point scan matching methods which operate in XY coordinate systems such as ICP or IDC find the correct translation and rotation of the current scan in one step if the correct associations are given. The PSM approach, due to the use of linearization, requires multiple iterations. Since the correct associations are in general not known multiple iterations are necessary for the other methods as well. Also note that the PSM approach for translation estimation is most accurate if the correct orientation of the current scan is known. Estimating the orientation of the current scan is described in section 3.3.

A negative property of this translation estimation approach is apparent when matching scans which were taken of long featureless corridors - the position error along the corridor can drift. In fig. 12 the reference and current scan contain only a wall. The associations are depicted with an arrow pointing from the current scan point to the reference scan point. The direction of the arrows also coincide with the

/**********Polar Translation Estimation**********/
00 //*Matrix multiplications for linearized least squares*
01 for i = 0 $\rightarrow$ number_of_points-1 {
02 $\Delta r = r_{ri} - r''_{ci}$
03 if !$tagged''_{ci}$ & !$tagged_{ri}$ &| Δr| $< PM_MAX_ERROR$ {
04 $w = \frac{C}{\Delta r^2 + C}$ //weight calculation
05 $h1 = \cos\phi_{ri}$
06 $h2 = \sin\phi_{ri}$
07 //calculating $H^T W \Delta r$
08 $hwr1 = hwr1 + w * h1\Delta r$
09 $hwr2 = hwr2 + w * h2\Delta r$
10 $hwh11 = hwh11 + w * h1^2$//calculating $H^T WH$
11 $hwh12 = hwh12 + w * h1 * h2$
12 $hwh21 = hwh21 + w * h1 * h2$
13 $hwh22 = hwh22 + w * h2^2$
14 }//if
15 }//for
16 $D = hwh11 * hwh22 - hwh12 * hwh21$
17 $inv11 = \frac{hwh22}{D}$
18 $inv12 = -\frac{hwh12}{D}$
19 $inv21 = -\frac{hwh21}{D}$
20 $inv22 = \frac{hwh11}{D}$
21 $\Delta x = inv11 * hw1 + inv12 * hw2$
22 $\Delta y = inv21 * hw1 + inv22 * hw2$
23 $x_c = x_c + \Delta x$
24 $y_c = y_c + \Delta y$

Figure 11: Pseudo code for translation estimation in polar coordinates.

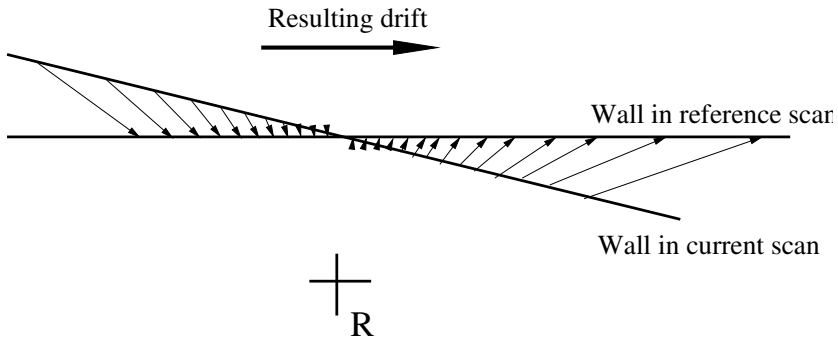


Figure 12: Cause of drift in for translation estimation in corridor like environments.

corresponding Jacobians which project into the x and y corrections. From fig. 12 it can be observed, that all the arrows have a positive x component, therefore the translation correction will drift to the right.

There are two reasons why polar scan matching estimates translation separately from orientation. First reason: if the partial derivatives $\frac{\partial r''_{ci}}{\partial \theta_c} = y_c \cos\phi_{ri} - x_c \sin\phi_{ri}$ are appended to matrix $\mathbf{H}$ (15), matrix $\mathbf{H^T W H}$ can become ill-conditioned and the estimation process can diverge. The cause of ill-conditioning lies in the structure of $\mathbf{H}$:

$$\mathbf{H} = \begin{bmatrix} \vdots & \vdots & \vdots \\ \cos\phi_{ri} & \sin\phi_{ri} & y_c \cos\phi_{ri} - x_c \sin\phi_{ri} \\ \vdots & \vdots & \vdots \end{bmatrix}, \tag{18}$$

where two columns contain small numbers in the range of $\langle -1, 1 \rangle$ and the third column contains potentially large numbers depending on the value of x_c and y_c. As an example let us assume that $x_c = 100,\ y_c = 100,\ \phi_{ri} = 0^\circ, 1^\circ, 2^\circ, ..., 180^\circ$ and $\mathbf{W}$ is a diagonal matrix with 1's on the diagonal. Then the largest eigenvalue of $\mathbf{H^T W H}$ is about $2x10^6$ and the smallest eigenvalue is about $3x10^{-33}$ which means the matrix $\mathbf{H^T W H}$ is ill-conditioned and will likely cause numerical instability. On the other hand if x_c and y_c are 0, then the right column of $\mathbf{H}$ will consist of 0's and $\mathbf{H^T W H}$ will have 0 determinant and will not have an inverse which is necessary for the computation of (16). The second reason why polar scan matching estimates translation separately from orientation is that estimating orientation as described later is much more efficient.

Note, that if uniform weights were used, and all measurements were used in each scan matching, then matrix $(\mathbf{H^T W H})$ is a constant matrix and as such it has to be calculated only once.

It is interesting to investigate how the matching bearing association rule performs with the pose estimation equations described in Lu and Milios [1997]. The details are given next.

3.2.1 Pose Estimation in Cartesian Frame

Lu and Milios in [1997] minimize the sum of square distance between current and actual scan points. To increase robustness it is recommended in [Gutmann, 2000], that only the best 80% of matches take part in the estimation process. Here instead of sorting the matches, each match is weighted based on its "goodness", as in the previous subsection. The original objective function in [Lu and Milios, 1997] expressed using the notation used in this report is:

$$E = \sum_{i=1}^{n} (x''_{ci}\cos\Delta\theta_c - y''_{ci}\sin\Delta\theta_c + \Delta x_c - x_{ri})^2 + (x''_{ci}\sin\Delta\theta_c + y''_{ci}\cos\Delta\theta_c + \Delta y_c - y_{ri})^2 \quad (19)$$

Where (x''_{ci}, y''_{ci}) correspond to the projected and interpolated current scans (ϕ_{ri}, r''_{ci}) in Cartesian coordinate frame. (x_{ri}, y_{ri}) corresponds to (ϕ_{ri}, r''_{ri}) of the reference scan. The weighted version used in this report:

$$E = \sum_{i=1}^{n} w_i \left[(x''_{ci}\cos\Delta\theta_c - y''_{ci}\sin\Delta\theta_c + \Delta x_c - x_{ri})^2 + (x''_{ci}\sin\Delta\theta_c + y''_{ci}\cos\Delta\theta_c + \Delta y_c - y_{ri})^2 \right] \quad (20)$$

Since (x_{ci}, y_{ci}) belong to the same bearing as (x_{ri}, y_{ri}), (20) is equivalent to the sum of weighted square range residuals $\sum w_i (r_{ri} - r''_{ci})^2$ used in the previous subsection. A solution to (20) can be obtained by solving $\frac{\partial E}{\partial x_c} = 0$, $\frac{\partial E}{\partial y_c} = 0$ and $\frac{\partial E}{\partial \theta_c} = 0$:

$$\Delta\theta_c = atan2\left(\bar{x}_r\bar{y}''_c - \bar{x}''_c\bar{y}_r + W(S_{y_r x''_c} - S_{x_r y''_c}), -\bar{y}_r\bar{y}''_c - \bar{x}''_c\bar{x}_r + W(S_{x_r x''_c} - S_{y_r y''_c})\right) \quad (21)$$

$$\Delta x_c = \frac{\bar{x}_r - \bar{x}''_c\cos\theta_c + \bar{y}''_c\sin\theta_c}{W} \quad (22)$$

$$\Delta y_c = \frac{\bar{y}_r - \bar{x}''_c\sin\theta_c - \bar{y}''_c\cos\theta_c}{W} \quad (23)$$

where

$$\begin{array}{cc} \bar{x}_r = \sum w_i x_{ri}, & \bar{y}_r = \sum w_i y_{ri} \\ \bar{x}''_c = \sum w_i x''_{ci}, & \bar{y}_c = \sum w_i y''_{ci} \\ S_{x_r y''_c} = \sum w_i x_{ri} y''_{ci}, & S_{x''_c y_r} = \sum w_i x''_{ci} y_{ri} \\ S_{x_r x''_c} = \sum w_i x_{ri} x''_{ci}, & S_{y''_c y_r} = \sum w_i y''_{ci} y_{ri} \\ W = \sum w_i & \end{array} \quad (24)$$

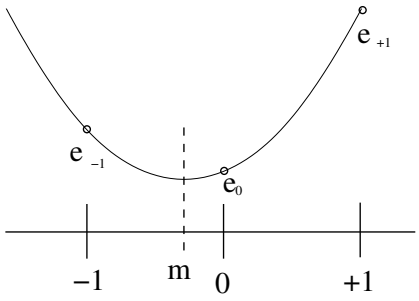


Figure 13: Orientation estimate improvement by interpolation.

Even though the objective function here is equivalent to the objective function in the previous subsection, the solutions are not equivalent. In the previous subsection, one iteration returns an approximate solution for x_c, y_c. Linearization was necessary due to the square root in (10). Here on the other hand a solution is calculated without linearization and without the need for multiple iterations (assuming known associations), which also contains θ_c and not just x_c, y_c. In experiments it was found that if only (21-23) are used to estimate pose, then the convergence speed is unsatisfactory, and the estimation process is more likely to get stuck in a local minima. Therefore just as in the previous subsection, it is best to interleave the described way of estimating x_c, y_c, θ_c with the orientation estimation described in the following subsection.

Note that the advantage of using (21)-(23) for calculating a solution of $\sum w_i (r_{ri} - r''_{ci})^2$ in one step opposed to the multiple iteration needed when using (16) is not important since the unknown associations of the reference and current scan points require an iterative pose estimation process. Also note that from now on using (21)-(23) together with the orientation estimation approach described next will be called PSM-C.

3.3 Orientation Estimation

Change of orientation of the current scan is represented in a polar coordinate system by a left or right shift of the range measurements. Therefore assuming that the correct location of the current scan is known and the reference and current scans contain measurements of the same static objects, the correct orientation of the current scan can be found by shifting the projected current scan (r''_{ci}, ϕ_{ri}) until it covers the reference scan. A $\pm 20^\circ$ shift was implemented at 1° intervals of the projected current scan, and for each shift angle the average absolute range residual is calculated. Orientation correction is estimated by fitting a parabola to the 3 closest points to the smallest average absolute error, and calculating the abscissa

of the minimum.

The calculation of the abscissa of the minimum is performed as follows. Assume that the 3 points of the error function are $(-1, e_{-1})$, $(0, e_0)$ and $(+1, e_{+1})$ (see fig. 12). Then the abscissa m of the minimum e_m of the parabola described as $e = at^2 + bt + c$ is sought. Given the equation of the parabola, the abscissa of the minimum can be found at:

$$\frac{\partial e}{\partial t} = 0 = 2am + b = 0 \Rightarrow m = -\frac{b}{2a} \tag{25}$$

To find a, b let us substitute the 3 known points into the equation of the parabola:

$$a - b + c = e_{-1} \tag{26}$$

$$c = e_0 \tag{27}$$

$$a + b + c = e_{+1} \tag{28}$$

By substituting (27) into (26) and (27), and adding (26) and (27), one gets:

$$2a + 2e_0 = e_{-1} + e_{+1} \Rightarrow a = \frac{e_{-1} + e_{+1} - 2e_0}{2} \tag{29}$$

Similarly b can be calculated by subtracting (26) from (27):

$$2b = e_{+1} + e_{-1} \Rightarrow b = \frac{e_{+1} - e_{-1}}{2} \tag{30}$$

Then the abscissa of the minimum is:

$$m = -\frac{b}{2a} = -\frac{\frac{e_{+1} - e_{-1}}{2}}{2\frac{e_{-1} + e_{+1} - 2e_0}{2}} = \frac{e_{+1} - e_{-1}}{2(2e_0 - e_{-1} - e_{+1})} \tag{31}$$

Assuming the orientation correction corresponding to 0 in fig. 13 is $\Delta\theta_1$, the distance between 0 and 1 in fig. 13 is $\Delta\phi$, then the estimated orientation correction will be

$$\Delta\theta_c = \Delta\theta_1 + m\Delta\phi \tag{32}$$

A simple pseudo code implementation of the orientation estimation is shown in fig 14. In fig 14 on lines 00-19 average absolute range residuals are calculated while shifting the reference range readings left and right by Δi. The value of Δi changes in the range of $\pm WINDOW$. The value of $WINDOW$ is chosen so, that the range of shift is around $\pm 20^\circ$. On lines 02-07 those indexes into the current range readings array are calculated which overlap with the shifted reference range array. In a for loop average absolute range residuals are calculated only for untagged range readings. The average range residuals for the corresponding shift values are then stored in $error_k$ and in β_k. Then the minimum error and the

corresponding shift value is found on lines 21-25, which is improved by fitting a parabola on line 26-27. $\Delta\phi$ on line 27 is the angle corresponding to changes of Δi.

The computational complexity of this orientation estimation approach depends on how the increments of Δi are chosen. If the reference scan is shifted by constant increments for example by 1° then the computational complexity is $O(n)$. The justifications for using constant increments, opposed to the smallest possible increment which is the angular resolution of the scan are the following:

- The orientation estimates are improved by quadratic interpolation.
- When performing scan matching in real environments the error in orientation due to fixed Δi increments will likely to be much smaller than errors caused by incorrect associations.

If the increments of Δi are chosen to be equal to the bearing resolution of the scans, then assuming constant size of search window in angles, the orientation estimation will be of $O(kn)$ complexity, where k is proportional to the number of range measurements per unit angle, i.e. to the angular resolution of the scan.

The last possibility discussed here in the choice of the increments of Δi is when one starts from a coarse increment of Δi and iteratively reduce Δi together with the size of the search window. In this case the computational complexity of $O(n \log n)$ may be achieved.

4 Error Estimation

4.1 Covariance Estimate of Weighted Least Squares

If correct associations are assumed, then the covariance estimate for the translation estimate of the polar scan matching algorithm is the same as the covariance estimate for weighted least squares [Kay, 1993]:

$$\mathbf{C} = \sigma_r^2 (H^T W H)^{-1}, \tag{33}$$

where σ_r^2 is estimated range error variance. σ_r^2 can be estimated based on the range residuals similarly to [Cox, 1991] as

$$\sigma_r^2 = \frac{[r_c'' - r_r]^T [r_c'' - r_r]}{n-4} \tag{34}$$

Unfortunately even if the current and reference scan were taken of the same scene, there can always be incorrect associations for example due to moving objects, or due to objects which appear differently from different location (e.g. vertically non uniform objects observed from a slightly tilted laser scanner). For this reason a heuristic error estimation usually yields more accurate results.

```
/**************Orientation estimation*****************/
00 for Δi = −WINDOW → +WINDOW {
01    n = 0, e = 0
02    if Δi <= 0
03       min_i = −Δi
04       max_i = number_of_points
05    else
06       min_i = 0
07       max_i = number_of_points − Δi
08    for i = min_i → max_i − 1 {
09       if !tagged''_ci & !tagged_ri+Δi
10          e = e + |r''_ci − r_ri+Δi|
11          n = n + 1
12    }
13    if n > 0
14       error_k = e/n
15    else
16       error_k = 1000
17    β_k = Δi
18    k = k + 1
19 }
20 e_min = LARGE_VALUE
21 for i = 0 → k-1 {
22    if error_i < e_min
23       e_min = error_i
24       i_min = i
25 }
26 m = (e_{i_min+1} − e_{i_min−1}) / (2(2e_{i_min} − e_{i_min−1} − e_{i_min+1}))
27 θ_c = θ_c + (β_{i_min} + m)Δφ
```

Figure 14: Pseudo code for orientation estimation in polar coordinates.

4.2 Heuristic Covariance Estimation

A simple heuristic error estimation approach was chosen to circumvent overoptimistic error estimates arising from incorrect associations. A preset covariance C_0 matrix is scaled with the square of average absolute range residual from which an offset S_0 is subtracted:

$$C = max\left((\frac{1}{n}\sum|\Delta r|)^2 - S_0, 1\right)C_0 \tag{35}$$

If the mean error is smaller than S_0 then $C = C_0$ to ensure, that the covariance estimate does not get too small. It is assumed that smaller range residuals are the result of better association and better scan matching results. However on a featureless corridors, one can have a small mean absolute range residual error, and a large along corridor error. Therefore different C_0 are chosen for corridor like areas and for non-corridor like areas. For scans of non corridor like areas, a diagonal covariance matrix is chosen. A non-diagonal covariance matrix is chosen for corridors which expresses the larger along corridor error. Classification of scans into corridors is done by calculating the variance of orientations (measured as angles of the normals) of line segments obtained by connecting neighboring points. If this variance is smaller than a threshold, then the scan is classified as a corridor. The orientation of the corridor necessary for the covariance matrix generation is estimated by calculating an angle histogram [Weiss and Puttkamer, 1995] from the line segment orientations. The angle perpendicular to the location of the maximum of the histogram will correspond to the corridor orientation.

5 SLAM using Polar Scan Matching

A simple implementation of Kalman filter SLAM was programmed in C++ to evaluate the practical usability of the described scan matching method. As in [Bosse *et al.*, 2004] laser scanner poses are used as landmarks. With each landmark the associated laser scan is also stored. Each time the robot gets to a position which is further than one meter from the closest landmark, a new landmark is created. Each time the robot gets closer than 50cm and 15° to a landmark not updated in the previous step, an update of the landmark is attempted. Note that consecutive scans are not matched. This is because it is assumed that short term odometry of the robot when traveling on flat floor is much more accurate than scan matching.

When updating a landmark, the observation is obtained by scan matching. The laser measurement is passed to scan matching as the reference scan, and the scan stored with the landmark is passed as the current scan. The result of scan matching is the position of the landmark expressed in the laser's coordinate system at the robot's current position.

A few details follow next. Following the notation and equations of Davison [Davison, 1998], a new feature is appended to the state as

$$\mathbf{x}_{new} = \begin{bmatrix} \mathbf{x}_v \\ \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_i \end{bmatrix} \tag{36}$$

$$\mathbf{P}_{new} = \begin{bmatrix} \mathbf{P}_{xx} & \mathbf{P}_{xy_1} & \cdots & \mathbf{P}_{xx}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \mathbf{P}_{y_1x} & \mathbf{P}_{y_1y_1} & \cdots & \mathbf{P}_{y_1x}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{P}_{y_{i-1}x} & \mathbf{P}_{y_{i-1}y_1} & \cdots & \mathbf{P}_{y_{i-1}x}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T \\ \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xx} & \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xy_1} & \cdots & \frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}\mathbf{P}_{xx}\frac{\partial \mathbf{y}_i}{\partial \mathbf{x}_v}^T + \frac{\partial \mathbf{y}_i}{\partial \mathbf{h}}\mathbf{R}\frac{\partial \mathbf{y}_i}{\partial \mathbf{h}}^T \end{bmatrix} \tag{37}$$

Where $\mathbf{x}_v$ is the vehicle pose, $\mathbf{y}_n = [x_n, y_n, \theta_n]$ is the n-th landmark, y_i is new landmark in world frame, $\mathbf{h}$ is the measurement of the new landmark expressed in robot frame and $\mathbf{R}$ is the estimated covariance of $\mathbf{h}$. Since the measurement $\mathbf{h}$ corresponding to the new landmark $\mathbf{y}_i$ is simply the accurately known pose of the laser on the mobile robot, the measurement noise covariance $\mathbf{R}$ is a null matrix. The transformation of the measurement into world frame is as following:

$$x_i = x_v - y_L \sin(\theta_v) \tag{38}$$

$$y_i = y_v + y_L \cos(\theta_v) \tag{39}$$

$$\theta_i = \theta_v \tag{40}$$

where it is assumed that the laser scanner's X axis is parallel to the robot's X axis and that the laser scanner's center is placed to $(0, y_L)$ on the robot. The Jacobian $\frac{\partial y_i}{\partial x_v}$ is then the following:

$$\frac{\partial y_i}{\partial x_v} = \begin{bmatrix} 1 & 0 & -\cos(\theta_v)x_L \\ 0 & 1 & -\sin(\theta_v)y_L \\ 0 & 0 & 1 \end{bmatrix} \tag{41}$$

Next the specifics for update are described. The prediction $h_i = [x_{hi}, y_{hi}, \theta_i]^T$ of the i-th landmark, i.e. the i-th landmark expressed in the lasers frame is calculated as:

$$x_{hi} = (x_i - x_v)\cos(\theta_v) + (y_i - y_v)\sin(\theta_v) \tag{42}$$

$$y_{hi} = -(x_i - x_v)\sin(\theta_v) + (y_i - y_v)\cos(\theta_v) - y_L \tag{43}$$

$$\theta_{hi} = \theta_i - \theta_v \tag{44}$$

PM_MAX_ERROR	100 cm
PM_MAX_RANGE	1000 cm
PM_MAX_ITER	30
PM_MIN_VALID_POINTS	40
PM_MAX_DIFF	20cm
C	70 cm reduced to 10 cm after 10 iterations

Table 1: Parameters used in scan matching during the experiment.

Then the Jacobian on h_i necessary for the update is:

$$\frac{\partial h_i}{\partial x} = \left[\begin{array}{cccccccc} \frac{\partial h_i}{\partial x_v} & 0^T & \cdots & 0^T & \frac{\partial h_i}{\partial y_i} & 0^T & \cdots \end{array} \right] =$$

$$\left[\begin{array}{cccccccc} -\cos\theta_v & -\sin\theta_v & -\sin(\theta_v)(x_v-x_i)+\cos(\theta_v)(y_v-y_i) & \cdots & \cos\theta_v & \sin\theta_v & 0 & \cdots \\ \sin\theta_v & -\cos\theta_v & -\cos(\theta_v)(x_v-x_i)-\sin(\theta_v)(y_v-y_i) & \cdots & -\sin\theta_v & \cos\theta_v & 0 & \cdots \\ 0 & 0 & -1 & \cdots & 0 & 0 & 1 & \cdots \end{array} \right] \quad (45)$$

6 Experimental Results

The results of 4 experiments are presented where the performance of PSM, PSM-C (polar scan matching using Cartesian coordinates) and a simple implementation of ICP are compared. In the first experiment simulated laser scans are matched and evaluated. The remaining experiments use a SICK LMS 200 laser range finder at a 1° bearing resolution in indoor environments. In the second experiment, laser scan measurements are matched at 10 different scenes by positioning the laser manually in known relative poses and the results are compared with the known relative poses. In the third experiment, the area of convergence for a particular pair of scans are investigated. The scan matching algorithms are evaluated in a SLAM experiment in the fourth experiment. The parameters used in all scan matching experiments are shown in table 1.

Every scan matching variant was stopped and divergence declared if the number of matches sank bellow PM_MIN_VALID_POINTS. The terminating condition for PSM and PSM-C was that either in 4 consecutive iterations

$$\varepsilon = |\Delta x_c[cm]| + |\Delta y_c[cm]| + |\Delta\theta_c[^\circ]| \quad (46)$$

was smaller than 1 or the maximum number of iterations PM_MAX_ITER has been reached. The need for a hard limit on the number of iterations is necessary, since PSM position estimate often drifts along corridors. Another reason for a hard limit is, that both PSM and PSM-C can enter a limit cycle. This limit cycle

is not a problem however if the terminating condition is chosen as 1. In the case of ICP, terminating condition had to be chosen as $\varepsilon < 0.1$, because of the low convergence speed of ICP. In the case of $\varepsilon < 1$, ICP often terminated with a too large error. Due to the slow convergence speed, the maximum number of iterations was chosen 60 for ICP, which is twice as much as that for PSM.

In PSM one position estimated step was followed by one orientation estimation step. These 2 steps are considered as 2 iterations. In PSM-C 3 pose estimation steps are followed by 1 orientation step. These 4 steps were considered as 4 iterations. Note that this counting of iterations is different to [Lu and Milios, 1997] where one position estimation step followed by an orientation estimation step was considered as one iteration for IDC.

In the following results, all the run times were measured on a 900Mhz Celeron laptop.

6.1 Simple Implementation of ICP

In the ICP implementation, initially a median filter is applied to the range readings of the reference and current scan. Then in each iteration the projection of the current scan follows similarly to [Lu and Milios, 1997]. First each current scan point is transformed into the reference scan's Cartesian coordinate system. Current scan points are then checked if they are visible from the reference position by checking the order of points. This is followed by checking if two neighboring (in a bearing sense) reference or current scan points occlude the current scan point being checked. Occluded current scan points are then removed, if they are at least one meter further back than their interpolated reference counterparts. Current scan points not in the field of view of the laser at the reference location are also removed. Note that none of the reference scan points are removed like in the projection filter in [Gutmann, 2000]. Reference scan points are not searched in this projection filter implementation, therefore this implementation is simpler but faster than of [Gutmann, 2000].

After scan projection, the implementation of the closest point association rule follows. For each remaining current scan point the closest reference scan point is sought in a $\pm 20^\circ$ window. Unlike in [Lu and Milios, 1997], there is no interpolation between neighboring reference scan points, which increases speed, but reduces accuracy. Associated points with larger than PM_MAX_ERROR distance are ignored. Then the worst 20% percent of associations are found and excluded. From the remaining associated point pairs pose corrections are calculated using equations from [Lu and Milios, 1997] and the current pose is updated.

The ICP algorithm is simpler than that described in [Lu and Milios, 1997] since at the corresponding point search there is no interpolation between two neighboring reference scan points and the search window size is not reduced ex-

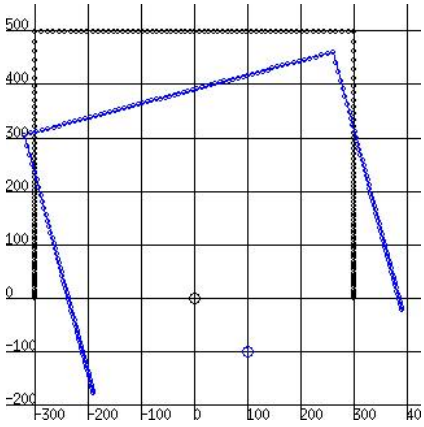


Figure 15: Current and reference scan prior to matching. Grid size in 1x1m.

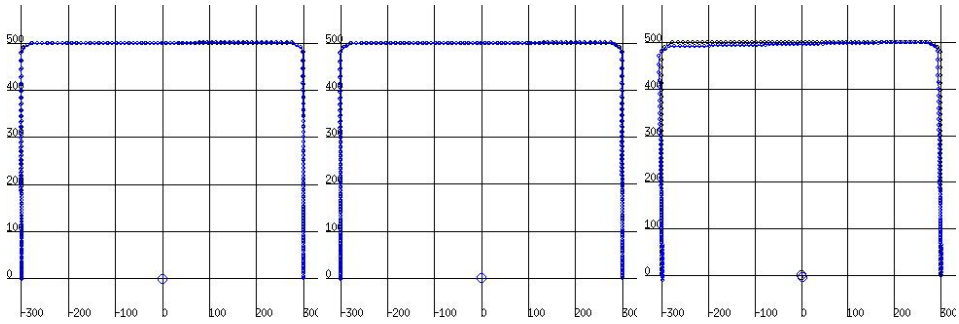
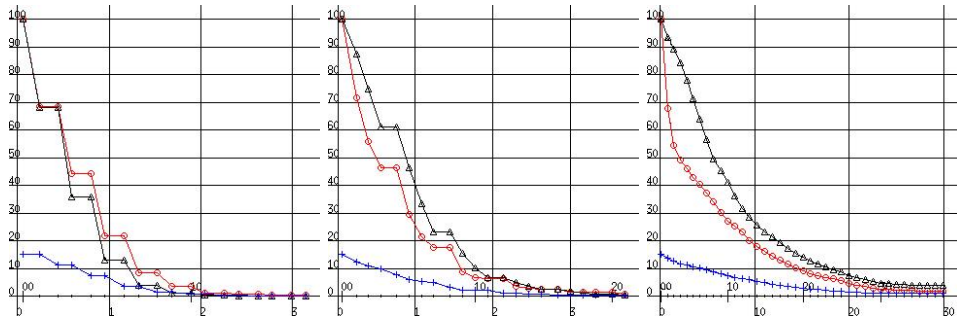


Figure 16: PSM, PSM-C and ICP results in the simulated experiment.Grid size in 1x1m.

ponentially with the number of iterations. However, projection of the current scan with occlusion testing has been implemented without expensive searches and therefore it has been included at the beginning of each iteration. Performing an occlusion check in each iteration opposed to once at the beginning can increase the accuracy of the results in the case of large initial errors where many visible points may be removed incorrectly or many invisible points are left in the scan incorrectly.

6.2 Simulated Room

Figure 15 shows two simulated scans of a room. The scans were taken of the same location, but the x and y position of the current scan was altered by 100 cm. Orientation was altered by 15°. Figure 16 shows the results after scan matching



a) b) c)

Figure 17: Evolution of x (circles),y (triangles) and orientation (crosses) error expressed in $[cm]$ and $[^\circ]$, respectively of PSM, PSM-C and ICP in the simulated experiment. Grid size is 1ms x 10cm and 1ms x 10°, respectively. The horizontal resolution of the grid for ICP is 10 ms. Iterations are marked with small vertical lines on the horizontal axis. Each 10-th iteration is marked with a longer vertical line.

.	iterations	time [ms]	$\|\Delta x\|$[cm]	$\|\Delta y\|$[cm]	$\|\Delta\theta\|$ $[^\circ]$
PSM	17	3.1	0.4	0.005	0.16
PSM-C	21	3.5	0.8	0.5	0.29
ICP	38	13.7	1.9	3.9	1

Table 2: Scan matching results of the simulated room.

using PSM, PSM-C and ICP. Figure 17 shows the evolution of errors. The final errors can be see on tab. 2.

From fig. 17 and tab. 2 it is clear that the PSM algorithm reached the most accurate result in shortest time, and ICP was the slowest and least accurate.

6.3 Ground Thruth Experiment

To determine how the polar scan matching algorithm variants cope with different types of environments, an experiment with ground truth information was conducted. On 4 corners of a 60x90cm plastic sheet, 4 Sick LMS 200 laser scanner outlines were drawn with different orientations. This sheet was then placed into different scenes ranging from rooms with different degrees of clutter to corridors. At each scene, laser scans were recorded from all 4 corners of the sheet, and matched against each other with initial positions and orientations deliberately set to 0 in the iterative procedure. The combinations of scans taken at corners which

take part in the scan matching are shown in tab. 3. Ground truth values have been determined by first measuring the left bottom corners of each outline with respect to an accurate grid printed on the plastic sheet. Then the measurement of outline orientations followed using ruler and the inverse tangent relationship. The determined orientations were also checked using a protractor. Finally a Matlab script was written which calculated the ground truth values by utilizing the relationship of the laser range finder's left bottom corner and the center of rotation of the mirror determined from the technical drawings of the Sick LMS's manual [SICK AG, 2002]. The carefully determined ground truth values for current scan poses in reference scan frames which also correspond to the initial errors are also displayed in tab. 3. From tab. 3 one can see, that the initial errors were up to 80cm displacement and up to 27° orientation. During the experiments the environment remained static.

A photo of each scene numbered 0-9 is shown on fig. 18. A matched current and reference scan from each scene is displayed in fig. 19 for PSM, fig. 21 for PSM-C and fig. 23 for ICP. The evolution of pose error is shown in fig. 20 for PSM, fig. 22 for PSM-C and fig. 24 for ICP. The displayed scans have all undergone median filtering. Only results for match 3 for each scene are displayed because match 3 contains a large initial error in displacement (77 cm) and a large initial error in orientation (-27°) as can be seen from tab. 3. Absolute residual between ground truth and match results together with number of iterations and runtime are shown in tables 5-7. There are 6 error vectors corresponding to each match for each scene. In Table 5 "ERROR" denotes a situation, when scan matching stopped due to the lack of corresponding points and divergence was declared.

Scene 0 is a room with a small degree of clutter. Current and reference scans were quite similar, and the matching results are good. Scene 1 is in a more cluttered room where laser scans from different locations look different as one can see in Fig. 19. The reason why the current scan differs from the reference scan so much is not clear. Perhaps the objects in the room were not uniform in the vertical direction and the laser beam is not a small spot or the laser was slightly tilted. The results for scene 1 (see Table 5-7, row 1) are not good for all 3 implementations, but they are still usable for example in a Kalman filter with an appropriate error estimate. In scene 2 the sheet was placed in front of a door to a corridor. The results are excellent. Scene 3 is a corridor without features. While the orientation error and the error in the cross corridor direction were quite small, the along corridor errors are large. PSM has the largest along corridor error of all, since the solution can drift in the direction of the corridor. With a proper error model (small orientation and cross corridor error, large along corridor error) the results are still useful when used with a Kalman filter. Scenes 4,5 and 6 are similar to 3. In scene 4 PSM diverged once. When observing the results for 4,5 and 6 in Fig. 19, there are phantom readings appearing at the corridor ends, even though the real corridor

match number	ref. scan recorded at corner	current scan recorded at corner	x [cm]	y [cm]	θ [°]
0	0	1	39.41	2.12	13
1	0	2	2.02	66.55	-14
2	0	3	38.84	66.99	12
3	1	2	-21.94	68.33	-27
4	1	3	14.04	68.33	-1
5	2	3	35.62	9.33	26

Table 3: Combinations of scans taken at different corners (numbered 0-3) of the plastic sheet for the ground truth experiment. These combinations marked as match number 0-5 were used for each scene. The pose of current scan with respect to reference scan is also shown. These poses correspond to the initial errors as well.

	iterations	time [ms]	orientation err. [°]	displacement err. [cm]
PSM	18.57	3.35	0.86	3.8
PSM-C	15.78	2.66	1.04	4.36
ICP	30.8	12.66	4.1	15.3

Table 4: Summary of average scan matching results in the ground truth experiment.

ends were 30 meters away. The likely reason for the phantom readings is a slight tilt of the laser beams causing and readings from the floor to be obtained. Scene 7 is situated on the border of a room and a corridor. The results are good for all 3 scan matching methods. Scenes 8 and 9 were situated in a room. The results are quite good except those of ICP.

To compare the 3 scan matching approaches average of errors, number of iterations and run times were calculated and shown on tab 4. Average orientation error, iteration and run time were calculated for all scenes except for the scenes 4,5,6 with the large phantom objects. In the average displacement error calculation, all corridor like environments (3,4,5,6) were left out, due to the large along corridor errors.

In the ground truth experiment, the implemented PSM and PSM-C clearly outperformed the implemented ICP. According to tab. 4 the performance of PSM and PSM-C are almost the same, with PSM being slightly more accurate but slower.

Figure 18: A photo of each scene with the plastic sheet in the foreground.

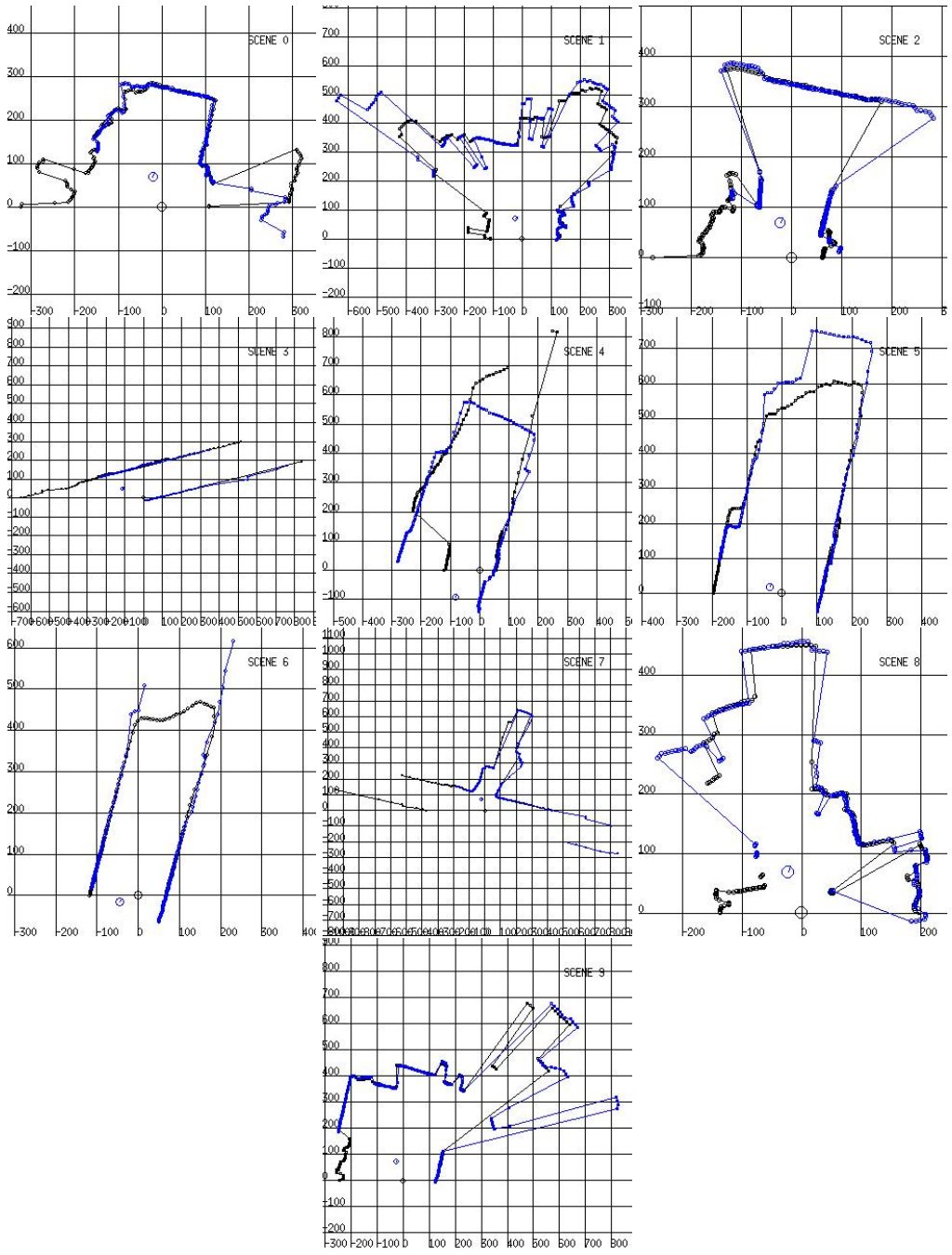


Figure 19: Scan match result for each scene for match number 3 in the experiment with ground truth using PSM.

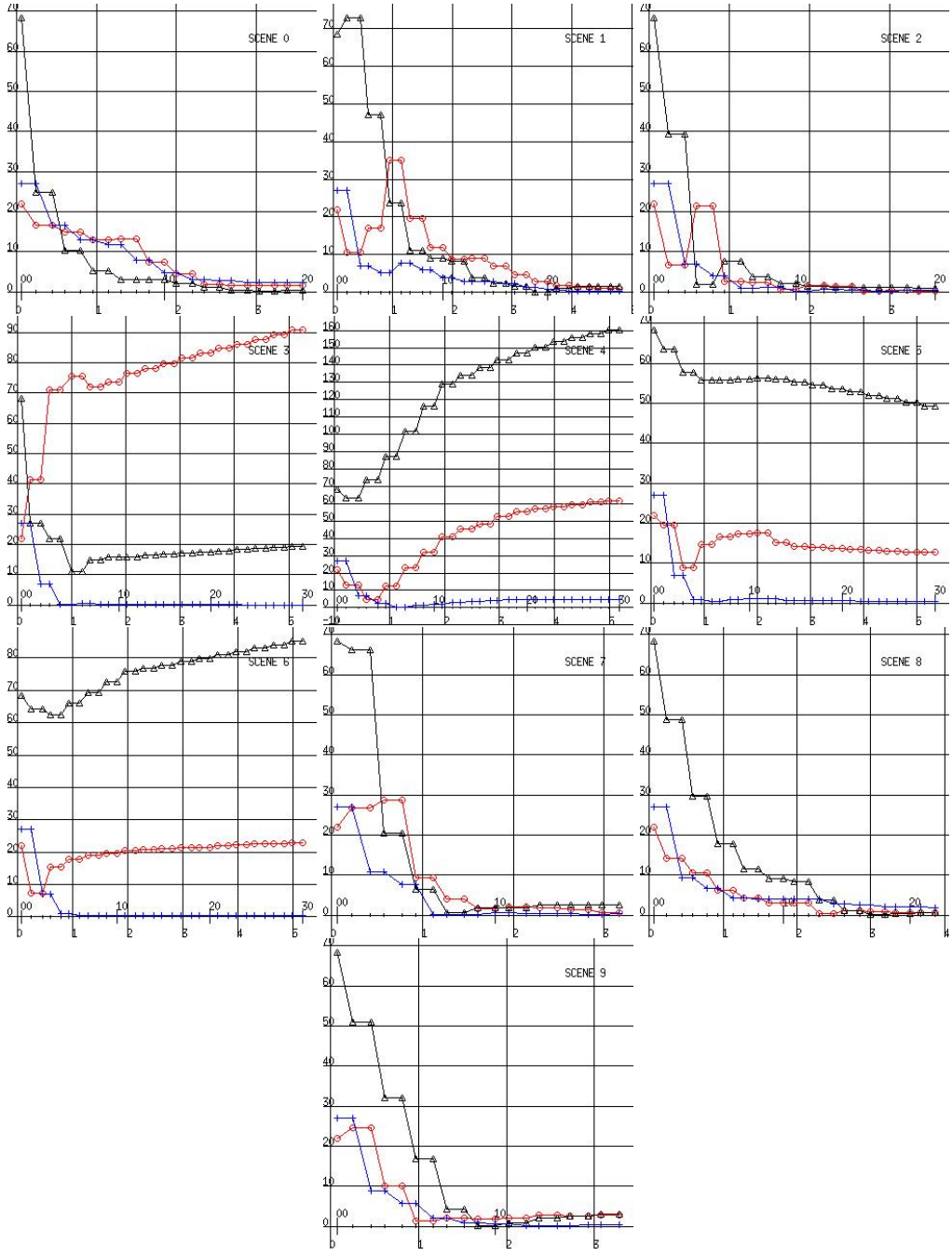


Figure 20: Match 3 scan match error evolution for each scene in the experiment with ground truth using PSM. Error in x (circles),y (triangles) and orientation (crosses) are expressed in $[cm]$ and $[^\circ]$, respectively. Grid size is 1ms x 10cm and 1ms x 10°, respectively. Iterations are marked with small vertical lines on the horizontal axis. Each 10-th iteration is marked with a longer vertical line.

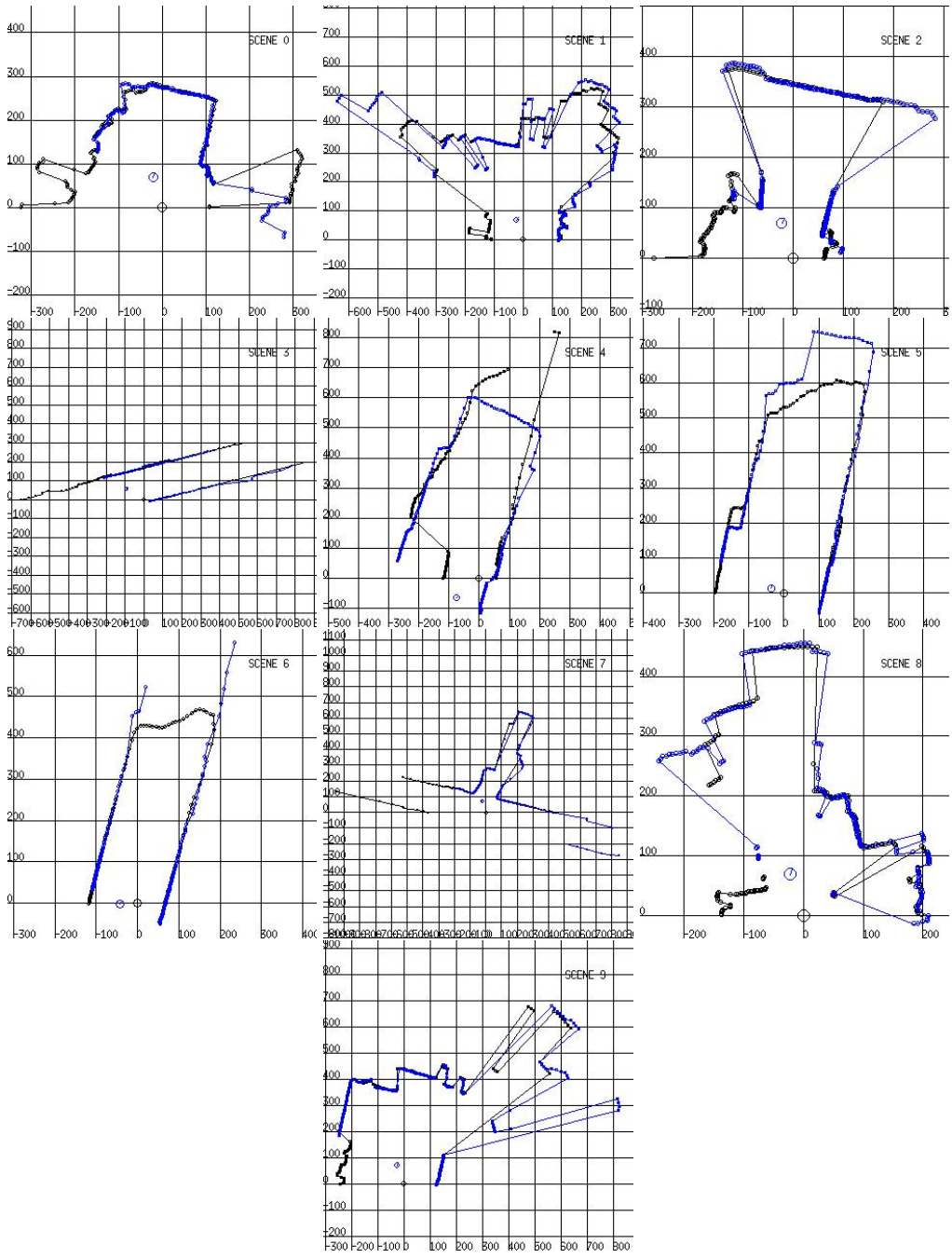


Figure 21: Scan match result for each scene for match number 3 in the experiment with ground truth using PSM-C.

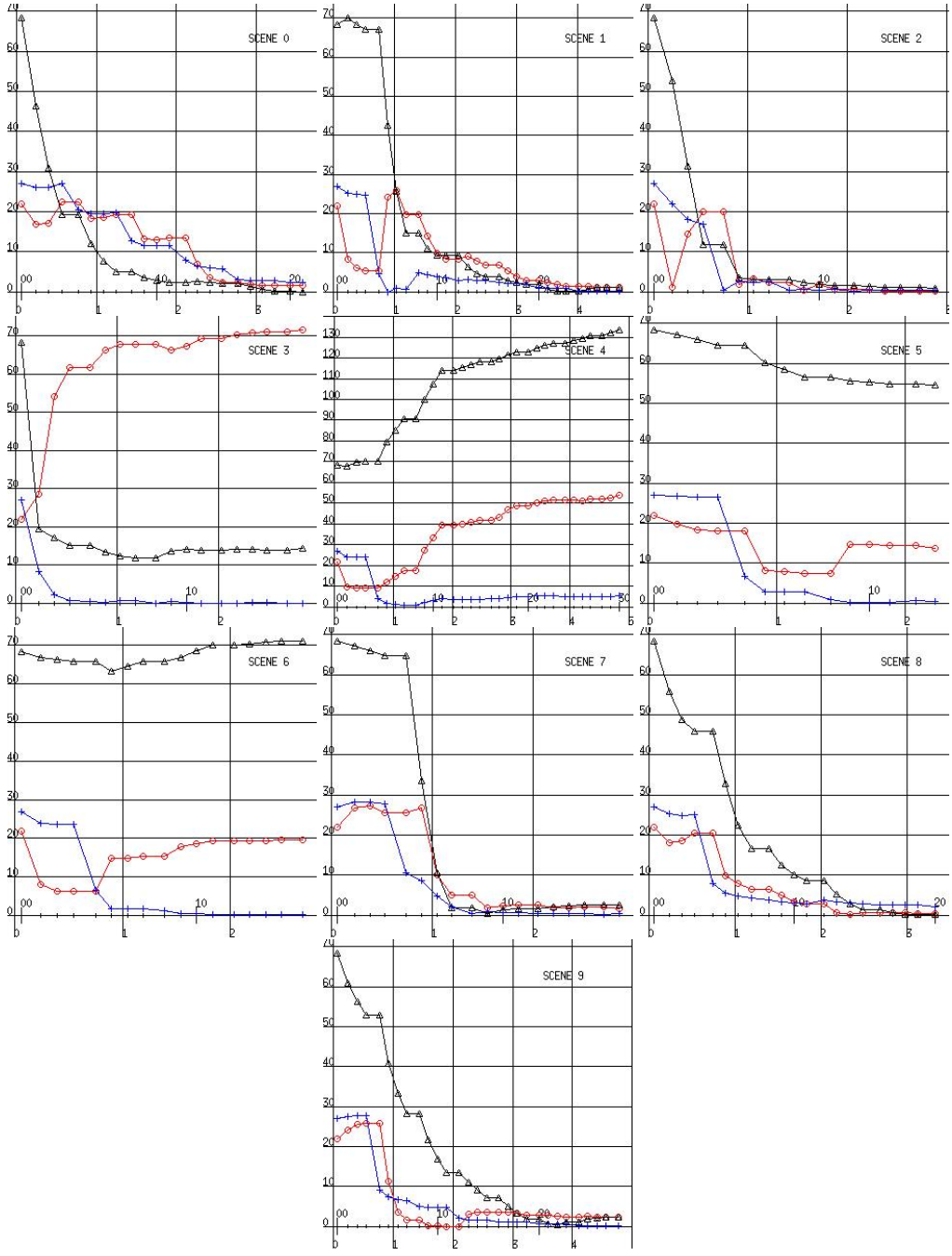


Figure 22: Match 3 scan match error evolution for each scene in the experiment with ground truth using PSM-C. Error in x (circles),y (triangles) and orientation (crosses) are expressed in $[cm]$ and $[^\circ]$, respectively. Grid size is 1ms x 10cm and 1ms x 10°, respectively. Iterations are marked with small vertical lines on the horizontal axis. Each 10-th iteration is marked with a longer vertical line.

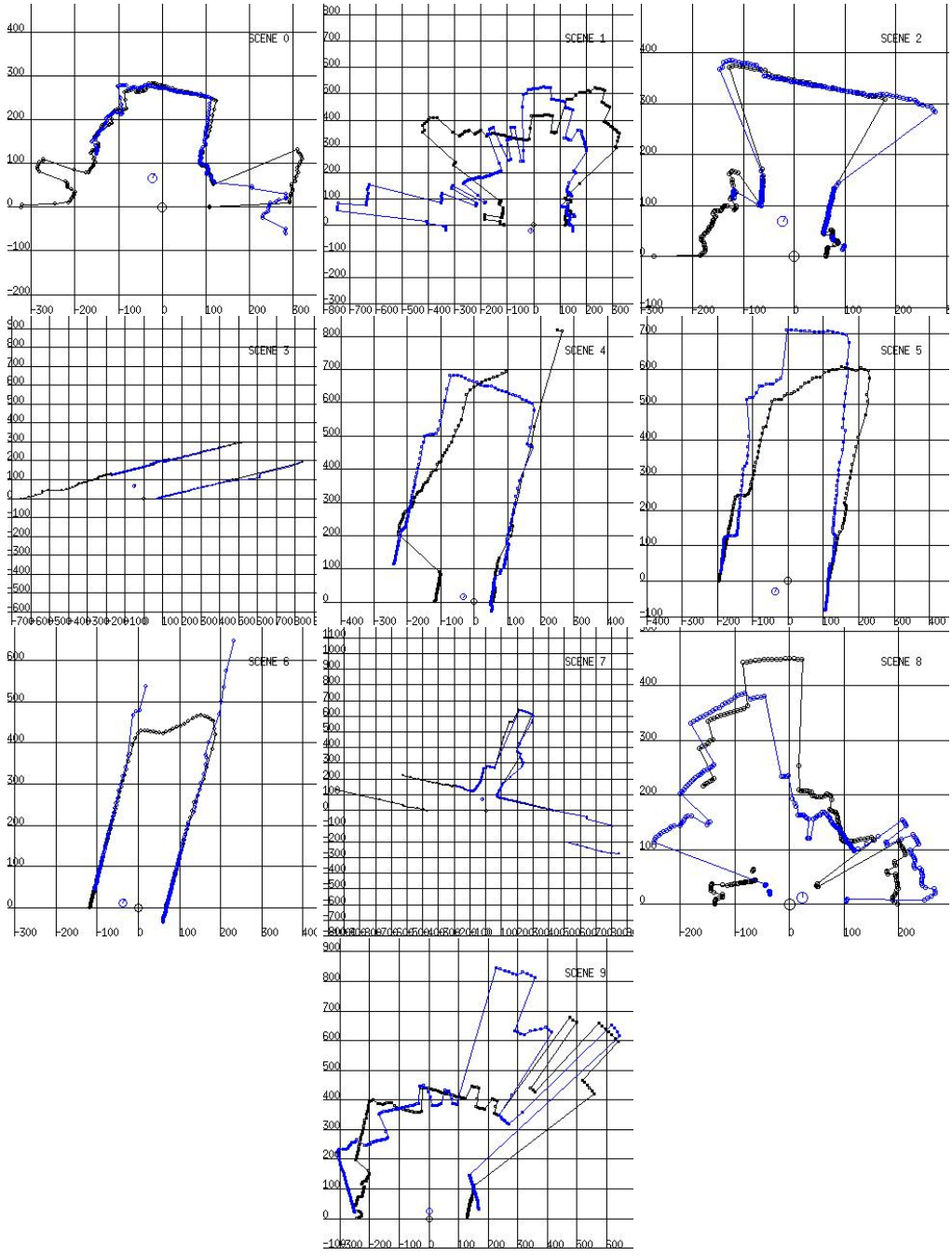


Figure 23: Scan match result for each scene for match number 3 in the experiment with ground truth using ICP.

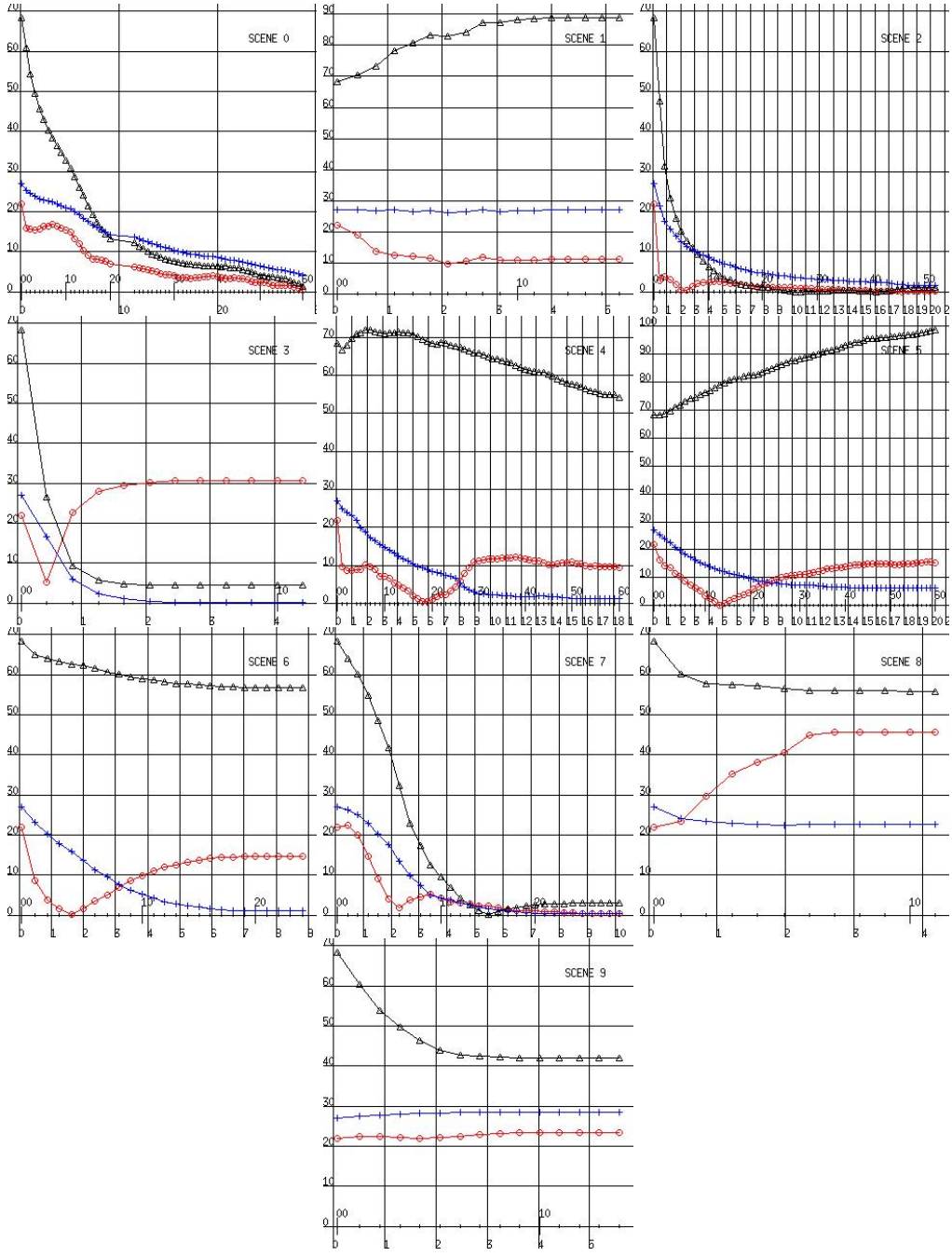


Figure 24: Match 3 scan match error evolution for each scene in the experiment with ground truth using ICP. Error in x (circles),y (triangles) and orientation (crosses) are expressed in $[cm]$ and $[^\circ]$, respectively. Grid size is 1ms x 10cm and 1ms x 10°, respectively. The horizontal resolution of the grid for scene 0 is 10ms. Iterations are marked with small vertical lines on the horizontal axis. Each 10-th iteration is marked with a longer vertical line.

0	(0.9, 1.5, 0.3) 14, 2.7	(0.7, 0.4, 1.3) 16, 2.9	(1.1, 0.2, 0.1) 18, 3.3	(1.5, 0.4, 2.4) 20, 3.6	(0.6, 7.4, 0.2) 12, 2.2	(3.6, 0.1, 1.3) 12, 2.2
1	(5.1, 17.3, 5.5) 20, 3.7	(7.7, 13.1, 5.8) 28, 5.1	(0.4, 24.8, 8.3) 30, 5.5	(1.0, 1.5, 0.4) 27, 4.8	(0.3, 5.0, 0.6) 24, 4.3	(2.4, 0.5, 0.6) 12, 2.3
2	(0.4, 0.3, 0.3) 8, 1.5	(0.2, 0.6, 0.1) 24, 4.2	(0.5, 1.0, 0.3) 20, 3.6	(0.2, 0.9, 0.3) 20, 3.5	(0.2, 4.8, 0.3) 28, 4.9	(1.0, 2.7, 0.3) 12, 2.2
3	(9.7, 5.0, 0.2) 18, 3.2	(51.8, 25.3, 0.2) 12, 2.2	(22.1, 11.1, 0.3) 20, 3.6	(90.8, 19.4, 0.0) 30, 5.2	(24.3, 11.2, 0.1) 16, 2.9	(55.3, 46.3, 0.1) 30, 5.2
4	ERROR	(4.2, 47.9, 1.3) 20, 3.7	(0.9, 4.3, 0.0) 30, 5.1	(61.6, 160.3, 4.7) 30, 5.2	(73.8, 210.5, 1.5) 30, 4.9	(1.0, 6.1, 0.1) 30, 5.1
5	(0.6, 20.3, 0.4) 24, 4.4	(0.2, 24.0, 0.4) 30, 5.5	(1.3, 10.7, 0.4) 30, 5.4	(12.6, 49.3, 0.5) 30, 5.4	(3.6, 6.6, 0.9) 21, 3.8	(1.6, 4.6, 1.5) 19, 3.4
6	(1.4, 30.7, 0.3) 18, 3.3	(2.0, 63.0, 0.1) 16, 2.9	(2.7, 79.1, 0.2) 16, 3.0	(23.0, 85.2, 0.4) 30, 5.2	(21.8, 86.6, 0.3) 16, 3.0	(0.8, 4.7, 0.0) 9, 1.7
7	(0.2, 0.1, 0.0) 26, 4.7	(1.5, 0.2, 0.2) 16, 2.9	(0.1, 0.3, 0.1) 20, 3.6	(0.8, 2.6, 0.3) 18, 3.2	(0.9, 4.9, 0.1) 16, 2.9	(0.0, 0.6, 0.3) 15, 2.7
8	(0.7, 0.0, 0.0) 11, 2.0	(1.3, 2.1, 0.1) 22, 3.9	(0.1, 0.4, 0.3) 19, 3.3	(0.6, 0.6, 1.9) 22, 3.9	(0.0, 5.6, 0.9) 23, 4.0	(0.6, 0.4, 0.1) 12, 2.2
9	(3.7, 1.7, 0.8) 10, 1.9	(2.0, 0.4, 0.4) 18, 3.3	(1.4, 0.9, 0.2) 18, 3.3	(2.8, 3.0, 0.5) 18, 3.3	(1.6, 9.5, 0.7) 12, 2.2	(1.1, 1.6, 0.2) 13, 2.4

Table 5: Absolute errors in x[cm], y[cm], $\theta[^\circ]$, number of iterations and runtime [ms] of the *PSM* algorithm in the experiments with ground truth.

0	(0.7, 1.4, 0.2) 13, 2.3	(0.8, 0.4, 1.3) 20, 3.5	(1.0, 0.3, 0.2) 17, 2.9	(1.6, 0.0, 2.4) 21, 3.6	(0.8, 6.5, 0.4) 14, 2.4	(3.0, 0.3, 1.1) 13, 2.3
1	(7.7, 19.3, 5.4) 12, 2.1	(7.4, 15.0, 6.2) 30, 5.0	(1.4, 26.7, 8.5) 30, 5.0	(1.2, 1.3, 0.3) 28, 4.6	(4.9, 16.3, 2.8) 13, 2.2	(1.0, 0.1, 0.4) 13, 2.2
2	(0.4, 0.1, 0.3) 9, 1.6	(0.8, 0.5, 0.1) 13, 2.2	(0.3, 0.7, 0.4) 13, 2.2	(0.3, 1.0, 0.3) 17, 2.8	(0.2, 4.6, 0.4) 13, 2.2	(1.0, 2.5, 0.3) 13, 2.2
3	(13.2, 6.9, 0.3) 9, 1.5	(49.4, 24.0, 0.2) 5, 0.9	(34.1, 17.2, 0.2) 6, 1.0	(71.5, 14.5, 0.0) 17, 2.8	(23.2, 10.9, 0.0) 6, 1.0	(29.1, 24.7, 0.1) 9, 1.5
4	(2.6, 44.4, 2.5) 17, 2.8	(6.0, 57.3, 1.6) 18, 3.1	(0.9, 1.5, 0.0) 30, 4.8	(53.8, 133.5, 5.3) 30, 4.9	(65.4, 169.9, 5.8) 30, 4.7	(6.1, 23.8, 0.2) 16, 2.6
5	(1.2, 17.8, 0.6) 17, 2.9	(0.1, 39.7, 0.5) 17, 2.9	(0.9, 25.8, 0.4) 21, 3.7	(13.8, 54.6, 0.4) 13, 2.2	(3.2, 2.8, 1.3) 26, 4.4	(0.9, 0.5, 0.7) 17, 2.9
6	(1.3, 24.1, 0.4) 17, 3.0	(1.5, 58.1, 0.4) 17, 2.8	(2.7, 74.1, 0.0) 13, 2.3	(19.6, 71.1, 0.3) 16, 2.6	(20.3, 79.9, 0.1) 17, 2.9	(1.3, 4.3, 0.3) 9, 1.6
7	(0.6, 0.1, 0.1) 17, 2.8	(0.9, 0.3, 0.1) 13, 2.2	(0.6, 0.0, 0.1) 14, 2.3	(1.5, 2.5, 0.3) 17, 2.9	(1.2, 4.9, 0.2) 13, 2.2	(0.2, 0.8, 0.1) 16, 2.7
8	(0.6, 0.7, 0.1) 13, 2.2	(1.1, 2.0, 0.3) 21, 3.4	(0.1, 0.3, 0.2) 20, 3.3	(0.5, 0.1, 2.2) 20, 3.3	(8.5, 13.3, 6.6) 13, 2.2	(0.3, 0.1, 0.0) 17, 2.9
9	(1.5, 1.4, 0.6) 13, 2.3	(1.5, 1.1, 0.2) 24, 4.1	(1.3, 2.5, 0.0) 13, 2.3	(2.3, 2.4, 0.1) 28, 4.7	(0.2, 5.2, 0.5) 20, 3.4	(0.8, 1.4, 0.2) 17, 2.9

Table 6: Absolute errors in x[cm], y[cm], $\theta[^\circ]$, number of iterations and runtime [ms] of the $PSM-C$ algorithm in the experiments with ground truth.

0	(0.4, 0.1, 0.5) 56, 24.3	(0.2, 0.5, 1.3) 60, 28.1	(0.9, 0.2, 0.4) 22, 10.5	(0.4, 1.5, 4.1) 60, 27.3	(0.7, 5.3, 0.7) 43, 20.2	(1.4, 0.8, 1.0) 60, 25.9
1	(7.2, 16.5, 6.4) 18, 7.5	(2.1, 1.6, 1.0) 38, 14.3	(13.0, 29.9, 9.6) 13, 5.0	(11.0, 88.9, 27.0) 16, 5.3	(13.5, 53.3, 0.6) 16, 6.3	(24.7, 5.5, 22.6) 60, 21.4
2	(0.2, 1.1, 1.3) 25, 10.9	(0.1, 0.6, 0.4) 55, 22.2	(1.2, 1.7, 1.8) 42, 17.4	(0.1, 1.1, 1.6) 52, 20.7	(1.3, 5.4, 1.1) 38, 15.9	(2.1, 4.4, 4.6) 35, 14.1
3	(3.0, 1.3, 0.2) 8, 3.3	(40.5, 19.9, 0.1) 21, 9.1	(37.8, 19.6, 0.3) 9, 3.9	(30.7, 4.5, 0.2) 11, 4.6	(30.5, 12.8, 0.0) 8, 3.6	(6.9, 5.7, 0.1) 23, 9.2
4	(0.5, 2.7, 1.3) 21, 8.3	(5.5, 61.3, 1.5) 30, 10.1	(6.2, 55.5, 1.5) 20, 7.0	(9.4, 54.3, 1.4) 60, 18.6	(10.7, 70.1, 1.3) 60, 20.8	(5.8, 22.4, 0.1) 36, 13.1
5	(0.1, 1.4, 0.1) 35, 13.7	(7.8, 89.4, 4.3) 60, 22.2	(0.6, 47.4, 0.4) 52, 21.9	(15.4, 98.7, 6.3) 60, 20.7	(14.6, 67.7, 0.0) 9, 4.1	(0.3, 2.2, 0.1) 47, 17.5
6	(0.1, 1.4, 0.1) 23, 9.9	(1.6, 65.4, 0.5) 35, 13.3	(2.4, 63.7, 0.1) 19, 7.6	(14.7, 56.7, 1.1) 24, 9.0	(17.0, 68.6, 0.2) 12, 5.2	(0.6, 0.5, 0.4) 27, 10.1
7	(0.2, 0.1, 0.0) 19, 7.4	(0.4, 0.0, 0.1) 23, 8.7	(0.1, 0.2, 0.3) 20, 7.9	(0.5, 3.1, 0.3) 29, 10.3	(1.6, 6.5, 3.0) 16, 6.8	(0.0, 0.2, 0.6) 32, 11.7
8	(1.9, 0.3, 2.0) 28, 10.9	(0.2, 0.1, 0.5) 23, 8.8	(28.0, 89.1, 11.6) 18, 6.5	(45.5, 55.9, 22.6) 11, 4.3	(2.1, 7.2, 1.9) 37, 15.2	(0.3, 0.4, 0.7) 56, 23.1
9	(1.1, 0.8, 0.8) 43, 17.9	(19.5, 55.6, 11.8) 21, 9.4	(1.0, 0.4, 0.6) 36, 14.6	(23.4, 42.0, 28.6) 14, 5.7	(0.5, 4.8, 0.4) 32, 13.7	(0.4, 1.8, 1.4) 48, 17.7

Table 7: Absolute errors in x[cm], y[cm], $\theta[^\circ]$, number of iterations and runtime [ms] of the *ICP* algorithm in the experiments with ground truth.

6.4 Convergence Map

The purpose of this experiment is to find those initial poses from which scan matching converges to an acceptable solution. Ideally one varies the initial position and orientation of the current scan in a systematic way and observes if the found solution is close enough to the true solution. Areas of convergence can be visualized by drawing the trajectory of the current scan into an image. To make visualization simpler just like in [Dudek and Jenkin, 2000] only the initial pose was changed.

Scan pairs from scenes 0-9, match 2 were selected for the convergence map experiment. Match 2 was selected in this experiment because the point of view for the current and reference scan differed the most from all matches. The initial orientation error was always 12°. The initial position varied from -250cm to 250cm for x and y in 10cm increments. The resulting convergence plots for scene 9 can be seen in fig. 25. Dark circles represent initial positions where the scan matching algorithms failed for the lack of associated points. Light colored circles represent final positions. Black lines correspond to the trajectories of the scans. Gray crosses mark the correct current scan positions.

When examining fig. 25, one has to keep in mind that in all scan matching implementations, associations having a larger error than one meter were discarded. To get an objective value for the performance of the implementations, the total number of matches and the number of successful matches were counted. Successful matches were matches with less than 10cm error in position and 2° in orienta-

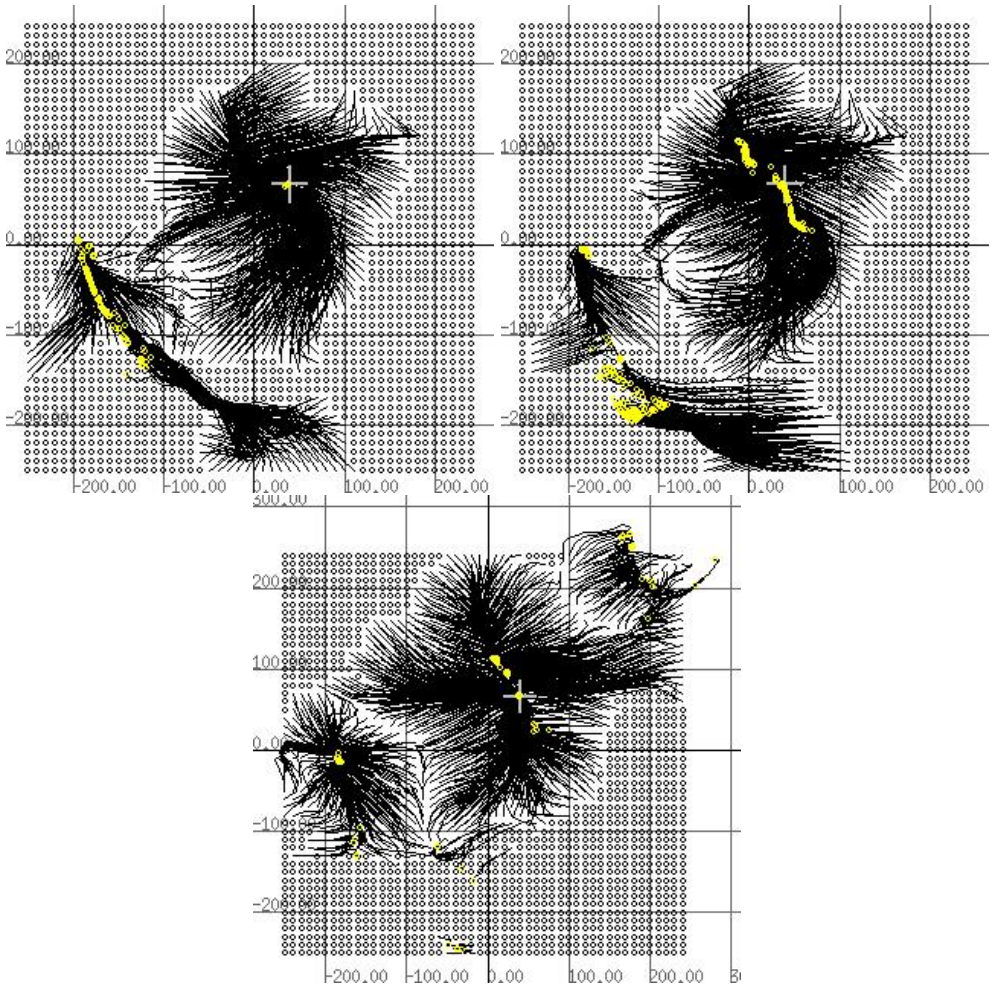


Figure 25: PSM, PSM-C and ICP convergence maps.

scene	PSM $[m^2]$	PSM-C $[m^2]$	ICP $[m^2]$
0	6.68	6.53	**7.18**
1	0.00	0.23	**2.05**
2	**4.38**	**4.38**	4.34
3	**0.52**	0.42	0.42
4	**0.81**	0.56	0.66
5	**3.10**	2.05	1.57
6	0.10	0.26	**0.36**
7	**3.71**	3.06	2.92
8	**3.60**	3.59	3.45
9	**7.01**	5.81	5.09
average	**2.99**	2.69	2.80

Table 8: Convergence areas for match 2 of all scenes for PSM, PSM-C and ICP.

tion. The total number of scan matching trials was 2500 which corresponds to an area of $25m^2$. PSM had 701 successful matches which means that it converged to the correct solution from around a $7m^2$ area. PSM-C had 581, giving $5.8m^2$. ICP had 509 correct ones which corresponds to $5.09m^2$.

A non-graphical representation of all the results can be seen in tab. 8. From this table one can observe that PSM has the largest average area of convergence followed by ICP and PSM-C. However the differences of the averages are small. One can also observe, that the area of convergence of PSM was in 7 cases larger than that of ICP. Note that the small areas of convergence from scenes 3-6 were due to the corridor like character of the scenes, where the lack of features limit the accuracy of scan matching results in the along corridor direction.

From this experiment one can conclude, that on average the implemented PSM converged from slightly larger area than ICP when using the dataset for scene 0-9, match 2. PSM-C performed slightly worse than ICP.

6.5 SLAM

The raw data set used in the Kalman filter SLAM is shown in fig. 26. The structures in the middle of the two rooms on the left are office cubicles. The third room is a seminar room filled with tables and chairs. The table and chair legs are represented by a randomly distributed point cloud. The robot was equipped with one SICK LMS 200 and odometry and started from the corridor intersection between the 2 rooms on the left. It visited the left room, and after one loop, it proceeded through the corridor to the middle room where it performed a large and a

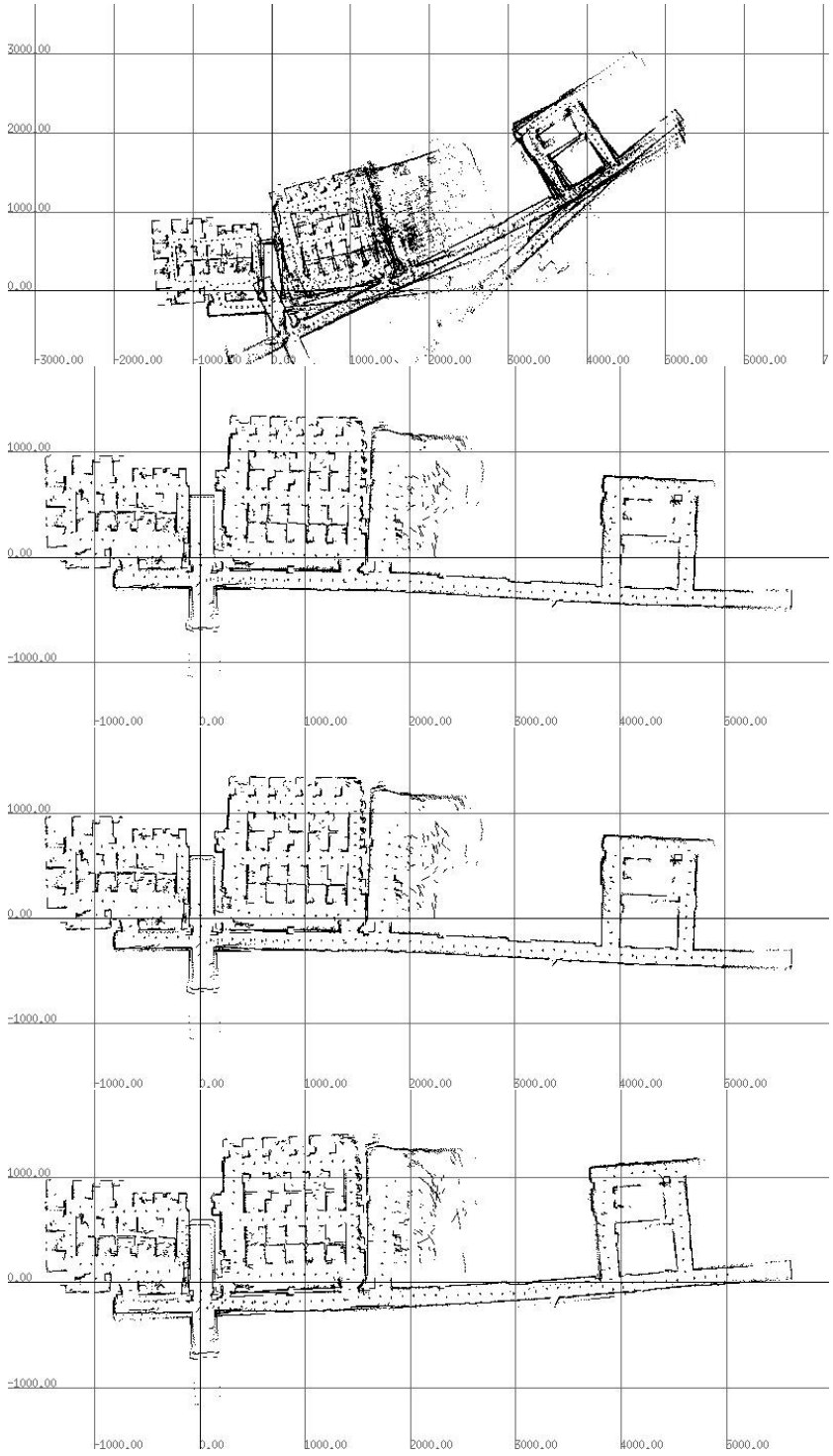


Figure 26: Maps resulting from odometry only, SLAM with PSM, PSM-C and ICP. Grid size 10x10m.

small loop and continued to the third (seminar) room. In the third room the robot was twice driven over a 1.5cm high cable protector on the floor at 40cm/s and at 20cm/s speed. After the visit to the third room the robot returned to its initial location from which it traveled to the far end of the corridor, went around a loop and came back. During the traversal of the environment, no less than 10 people walked in the view of the laser scanner and some doors were opened and closed. Considering the presence of walking people, repetitive cubicles, long corridors and 2 collisions with an obstacle on the floor, this dataset is not the most ideal for mapping.

The SLAM results are shown in fig. 26. The SLAM results are significantly better than those from odometry only (fig 26, top). Consecutive laser scans were not matched against each other, since it was assumed that over short distances odometry is more accurate than scan matching. PSM performed best on this data set followed by PSM-C, and worst with the ICP implementation. Note that the odometry of the robot was reasonably calibrated. This was necessary to be able to perform loop closing in the repetitive cubicle environment of the second room without the implementation of special loop closing algorithms.

In the C++ implementation of scan matching and SLAM, the 20 minutes worth of data consisting of 35×10^3 scans and 13×10^4 odometry readings took about 2 minutes to process on a 900MHz Celeron laptop for all variants. There were 100 successful (no divergence) scan matches for PSM SLAM variant with an average of $3.1ms$ scan matching time. There were 100 scan matches with PSM-C, with $2.1ms$ average time. ICP was successfully used 65 times with an average of $12.6ms$.

The dataset used in this SLAM experiment was also processed using a program performing SLAM using fused laser and advanced sonar measurements. The advanced sonar arrays are capable of measuring range and bearing with a standard error of 0.1° and 0.2mm respectively, and classifying sensed targets into *planes*, *right angle corners* and *edges* [Kleeman, 2002]. In the fusion scheme, the advanced sonar aids laser line segmentation by detecting small edge and corner features not seen by the laser. Planes and corners measured by laser and sonar are fused, achieving greater robustness and accuracy. Finally laser measurements help to remove spurious sonar measurements and to select good point features. More on SLAM using advanced sonar and laser can be read in [Diosi and Kleeman, 2004]. To accurately overlay the laser scans on a the feature based SLAM map the robot's path together with laser scans are stored with respect to neighboring SLAM features. Then when mapping is completed, the previous robot poses are recalculated using the stored neighboring features and the associated SLAM map features in a process similar to scan matching. More on how to use feature based SLAM for occupancy grid generation can be read in [Diosi *et al.*, 2005].

When comparing the map created using SLAM with PSM (fig. 26) with the

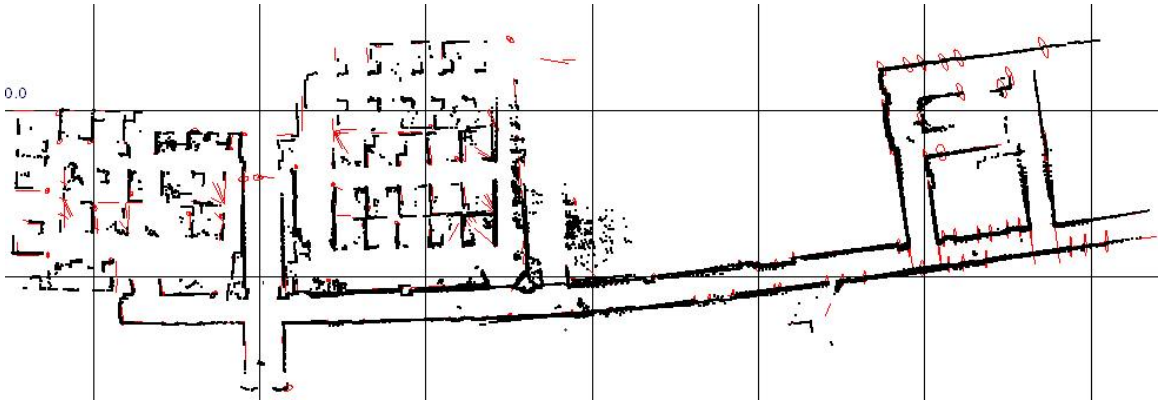


Figure 27: Results from SLAM using laser and advanced sonar fusion.

results in fig. 27 from SLAM with fusion of laser and advanced sonar, one can notice that laser scans are better aligned in fig. 27, however SLAM with sonar and laser could not cope well with the error introduced at the collision of the robot with the cable protector which resulted in a curved corridor.

The advantages of performing SLAM with PSM over SLAM with fusion of advanced sonar and laser measurements are the following:

- Since laser scans are in general more distinguishable than line segments or point features, association can be made more robust due to easier detection of failure.
- Scan matching may converge from an initial error in the order of meters and tens of degrees. When performing SLAM with advanced sonar and laser, such errors in the relative pose of the robot with respect to neighboring landmarks would likely result in an invalid association or in a robot unable to update it's pose due to many features falling into measurement validation gates. SLAM with scan matching is more robust with respect to error in the robot pose.
- Since the robot is allowed to have a larger pose estimate error with respect to neighboring landmarks, the SLAM filter can be run at lower update rate which makes scan matching SLAM more computationally efficient.
- Since there is no need to extract features, SLAM with scan matching may work in environments lacking lines, points and corners necessary for the functioning of SLAM with advanced sonar and laser.
- Since SLAM with scan matching is implemented using only a laser scanner and an odometry sensor, in environments where a laser scanner is sufficient

for navigation a robot performing SLAM with scan matching is cheaper. Less sensors in this case also mean easier coding and debugging. SLAM with advanced sonar and laser requires two extra advanced sonar arrays which make a robot more expensive and require more complicated coding and debugging.

The disadvantages of performing SLAM with PSM over SLAM with fusion of advanced sonar and laser measurements are the following:

- Scan matching can get stuck in local minima.
- Unlike in SLAM with advanced sonar and laser, moving objects may have adverse effects on the scan matching results.
- In corridors, scan matching may correct the robot's pose in the cross corridor direction. This can result in a robot getting lost in long corridors. In SLAM using laser and advanced sonar, the existence of small features on the corridor such as doorjambs or wall moldings are sufficient for correct operation. Therefore a robot navigating with SLAM using scan matching may get lost while a robot with SLAM using advanced sonar and laser still operates properly.
- SLAM with scan matching works best using 360° scans. If a robot with a laser scanner providing only 180° coverage heads down a corridor, then on the way back it will not be able to update the landmarks created in the way there due to the lack of sufficient overlap. If the robot uses 360° scans, the lack of overlap problem does not occur.

7 Extensions to 3D

3D scan matching has gained popularity in recent years. Therefore it would be interesting to know if PSM could be adapted to 3D. The projection filter and position estimation would be still of $O(mn)$ and $O(n)$ complexities, respectively. Estimation of the 3 orientation angles if done sequentially would still result in $O(kn)$ complexity, where k is proportional to the number of range readings per unit angle. Even though 3D scan matching with a modified PSM is an exciting problem, due to the lack of time we have to consider it as possible future work.

8 Discussion

In this report a laser scan matching method is proposed which works with the laser measurements in their native, polar form. The polar scan matching (PSM)

approach belongs to the class of point to point matching algorithms. PSM takes advantage of the structure of laser scanner measurements by functioning in the laser scanner's polar coordinate system. The direct use of range and bearing measurements coupled with a matching bearing association rule and a weighted range residual minimization, results in an $O(n)$ complexity pose estimation approach and an $O(kn)$ complexity orientation estimation approach, opposed to the $O(n^2)$ complexity of the popular IDC algorithm by Lu and Milios [Lu and Milios, 1997]. In $O(kn)$ of the orientation estimation approach, k is proportional to the angular resolution of the laser scans. Opposed to the $O(n^2)$ projection filter of [Gutmann, 2000], preprocessing of scans is done also with $O(n)$ complexity if there are no occlusions in the current scan when viewed from the reference scans position. A variant of PSM, PSM-C is also introduced where the translation estimation step of PSM is replaced with a weighted variant of the pose estimation equations from [Lu and Milios, 1997]. In PSM-C due to the use of the matching bearing rule, equations from [Lu and Milios, 1997] also minimize the sum of square range residuals. For comparison, a basic ICP has also been implemented.

Simulation of matching scans in a room demonstrates that the current scan pose error decreases more quickly with PSM and PSM-C to a small value, than with ICP. Scan matching experiments were also performed with a SICK LMS 200 in a variety of environments. Comparison of the results with ground truth revealed that in the tests, the performance of PSM and PSM-C surpasses that of ICP in speed and accuracy. However when matching corridors, a position drift in the direction of the corridor has been observed with PSM. This drift was not observed when using PSM-C or ICP. A comparison of areas of convergence for PSM, PSM-C and ICP were also performed. It was found, that PSM converged to the correct solution from slightly larger area than PSM-C and ICP.

A simple heuristic scan matching error model has been proposed where first scans are classified into rooms and corridors. For rooms a diagonal covariance matrix is scaled by the sum of square errors. For corridors, the xy sub-matrix (of the covariance matrix) with a large error in the x direction and small error in the y direction is rotated first, so that the large error aligns with the direction of the corridor. Then the resulting covariance matrix is scaled by the average square error.

The usability of the proposed scan matching approaches has been tested by performing Kalman filter SLAM with scan matching in a static environment. The maps created by PSM and PSM-C are better than that of ICP as shown in fig. 26. In fig. 26, the quality of the maps can be judged by the straightness of the corridor and by the presence of walls with multiple representations.

A map resulting from SLAM with PSM has been compared to a map resulting from SLAM using laser and advanced sonar fusion and path tracking. Laser scans of the map resulting from SLAM with PSM were less aligned than those

from SLAM using advanced sonar and laser. In general SLAM with PSM can experience problems when there is significant motion in the scans and also on corridors with not enough features to correct the along corridor error of the robot. On the other hand SLAM with PSM compared to SLAM with advanced sonar and laser can handle larger robot errors relative to neighboring landmarks, needs less frequent landmark updates, is simpler and it does not extract features from the environment.

As future work, the tracking and tagging of moving objects could be considered. Also the real advantage of the efficient PSM over $O(n^2)$ methods becomes more apparent when the number of points is large. One such case is in 3D scan matching. The modification of PSM for 3D scan matching is also considered as future work.

References

[Bengtsson and Baerveldt, 2001] O. Bengtsson and A.-J. Baerveldt. Localization by matching of range scans - certain or uncertain? In *Eurobot'01 - Fourth European Workshop on Advanced Mobile Robots*, pages 49–56, Lund, Sweden, Sep. 2001.

[Besl and McKay, 1992] P. J. Besl and N. D. McKay. A method for registration of 3D shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, 1992.

[Biber and Straßer, 2003] P. Biber and W. Straßer. The normal distributions transform: A new approach to laser scan matching. In *IROS'03*, volume 3, pages 2743–2748. IEEE, 2003.

[Bosse *et al.*, 2004] Michael Bosse, Paul Newman, John Leonard, and Seth Teller. Simultaneous localization and map building in large-scale cyclic environments using the Atlas framework. *The International Journal of Robotics Research*, 23(12):1113–1139, 2004.

[Cox, 1991] I. J. Cox. Blanche–an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Transactions on Robotics and Automation*, 7(2):193–203, april 1991.

[Davison, 1998] A. Davison. *Mobile Robot Navigation Using Active Vision*. PhD thesis, University of Oxford, 1998.

[Diosi and Kleeman, 2004] A. Diosi and L. Kleeman. Advanced sonar and laser range finder fusion for simultaneous localization and mapping. In *Proc. of IROS'04*, volume 2, pages 1854–1859. IEEE, 2004.

[Diosi and Kleeman, 2005] A. Diosi and L. Kleeman. Laser scan matching in polar coordinates with application to SLAM. Accepted for publication in the Proc. of IROS'05, 2005.

[Diosi *et al.*, 2005] A. Diosi, G. Taylor, and L. Kleeman. Interactive SLAM using laser and advanced sonar. In *Proc. of ICRA'05*. IEEE, 2005.

[Dudek and Jenkin, 2000] G. Dudek and M. Jenkin. *Computational Principles of Mobile Robotics*. Cambridge University Press, Cambridge, 2000.

[Gutmann, 2000] J.-S. Gutmann. *Robuste Navigation autonomer mobiler Systeme*. PhD thesis, Albert-Ludwigs-Universität Freiburg, 2000.

[Hähnel *et al.*, 2003] D. Hähnel, W. Burgard, D. Fox, and S. Thrun. An efficient fastSLAM algorithm for generating maps of large-scale cyclic environments from raw laser range measurements. In *IROS'03*, volume 1, pages 206–211. IEEE, 2003.

[Kay, 1993] Steven M. Kay. *Fundamentals of Statistical Signal Processing*, volume 2. Estimation Theory. Prentice Hall, New Jersey, 1993.

[Kleeman, 2002] L. Kleeman. On-the-fly classifying sonar with accurate range and bearing estimation. In *IEEE/RSJ Int. Conf. on Intelligent Robots & Systems*, pages 178–183. IEEE, 2002.

[Lingemann *et al.*, 2004] K. Lingemann, H. Surmann, A. Nüchter, and J. Hertzberg. Indor and outdoor localization for fast mobile robots. In *IROS'04*, volume 3, pages 2185–2190. IEEE, 2004.

[Lu and Milios, 1997] F. Lu and E. Milios. Robot pose estimation in unknown environments by matching 2D range scans. *J. of Intelligent and Robotic Systems*, 20:249–275, 1997.

[Lu, 1995] Feng Lu. *Shape Registration Using Optimization for Mobile Robot Navigation*. PhD thesis, University of Toronto, 1995.

[SICK AG, 2002] SICK AG, Germany. *LMS 200 / LMS 211 / LMS 220 / LMS 221 / MS 291 Laser Measurement Systems*, 2002.

[Thrun *et al.*, 2000] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3D mapping. In *ICRA'00*, volume 1, pages 321–328. IEEE, 2000.

[Tomono, 2004] M. Tomono. A scan matching method using Euclidean invariant signature for global localization and map building. In *ICRA'04*, volume 1, pages 886–871. IEEE, 2004.

[Weiss and Puttkamer, 1995] G. Weiss and E. Puttkamer. A map based on laser-scans without geometric interpretation. In *Intelligent Autonomous Systems - 4*, pages 403–407, Germany, 1995.

[Ye and Borenstein, 2002] C. Ye and J. Borenstein. Characterization of a 2-D laser scanner for mobile robot obstacle negotiation. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, pages 2512–2518, Washington DC, May 2002. IEEE.