A Multi-Layer Mobile Robot Localisation Solution using a Laser Scanner on Reconstructed 3D Models[+]

João Gomes-Mota[*] , Maria Isabel Ribeiro[*]

Instituto Superior Técnico/Instituto de Sistemas e Robótica
Av.Rovisco Pais
P 1096 LISBOA CODEX
PORTUGAL

Abstract. This paper presents a localisation strategy for a mobile robot using a Laser Range Scanner. A three module recursive algorithm was developed. One module uses feature matching to compute the robot localisation without any prior posture estimate. The other module, based on local data matching, provides enhanced posture estimation given an initial posture and the third module is a likelihood test to verify the correction and accuracy of the algorithms' solutions.

1. Introduction

In many operation environments, one of the first tasks a mobile robot must accomplish is to localise itself relative to some external reference. The sensors used for localisation have evolved along with the advance of mobile robotics. From odometry and ultrasounds to video images or GPS, new sensors, more accurate and sophisticated, have emerged. Among them, the Laser Range devices are increasingly popular given their high accuracy, resolution and speed of acquisition.

Although the acquisition devices differ in their construction and operation, the data output from Laser Range Scanners is similar for nearly all range scanner sensors: a set of discrete points, usually in one common 2D plane, measured radially from the sensor. The difference between the various localisation methods is the type of external reference (the map) and the data processing algorithm.

This paper presents a novel strategy to perform localisation using a Laser Range Scanner without any prior estimate of the mobile robot posture, defined by position in the ground plane ($\mathbf{x}$,$\mathbf{y}$) and orientation, $\boldsymbol{\theta}$. It is based on a three module algorithm, each module processing the data differently. This approach overcomes most problems of individual algorithms, increases robustness while keeping the development and analysis fairly simple, due to its modular nature.

The paper is organised as follows: Section 2 describes the RESOLV project, for which this algorithm was developed. Section 3 summarises some localisation algorithms using laser range scanners developed in recent years. Section 4 presents the first module of the proposed solution, the Frame Localisation algorithm, to be used in case no *a priori* posture estimate is available. Section 5 presents the second module, the Likelihood Test, which is an error estimate rule. Section 6 presents the Approximate Localisation algorithm, to be used if an initial posture estimate is available. Sections 4 to 6 are illustrated with experimental data of a real essay main stages. Section 7 describes the interactions between the three modules. Finally, in Section 8, some conclusions are taken, along with comments on known issues and directions for further research.

2. The RESOLV project

The RESOLV-REconstruction using Scanned Laser and Video project's goal is to create a 3D reconstruction of a real environment using laser range images and video images to be shown and operated through the Internet, [Leevers *et al*, 98], [RESOLV, Web]. The 3D models, similar to Figure 1c are represented in VRML language and are available via a Internet browser.

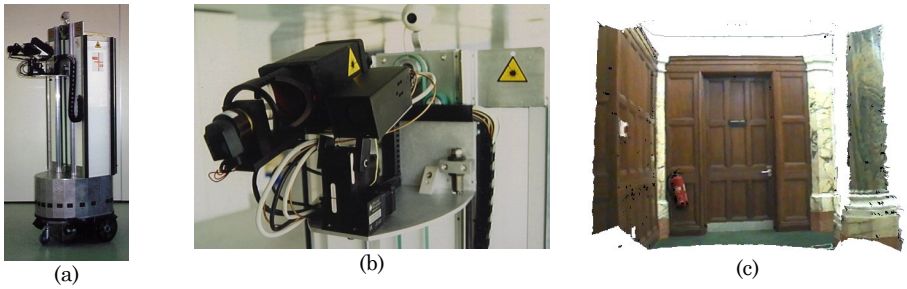
[+] Work supported by the project RESOLV-REconstruction using Scanned Laser and Video of the ACTS Programme, EU.

[*] Email: jsgm@isr.ist.utl.pt, mir@isr.ist.utl.pt

The main sensor for RESOLV is represented in Figure 1b: it consists of a video camera and a laser camera with a rotating mirror on the front, both mounted on top of a Pan & Tilt motorised unit. This compound sensor is connected to a computer which generates the reconstructed 3D models. This computer also hosts the Internet server, providing access to the 3D models and system operation via the Internet. In order to reconstruct a complex scene the system must be positioned at different spots to resolve occlusions and increase the quality of the representation by approaching the scene features. At first, the equipment was mounted on a manually pushed trolley, but it was soon realised that a mobile robot was the perfect candidate for the task, since automatic motion was required.

The mobile robot is used to transport the RESOLV equipment from between successive acquisition points and the compound system, shown in Figure 1a, was termed AEST - Autonomous Environment Sensor for Telepresence. The AEST includes odometers for in-motion localisation and ultrasound sensors for obstacle detection and avoidance.

The sensors for Localisation were constrained to the ones already embedded on the system. Moreover, the reconstructed 3D models are based on 3D spatial information computed from Laser Range data, leading to the natural solution of using the Laser Range Scanner for localisation purposes. Localisation data is used for navigation, correcting the odometry cumulative errors and for 3D reconstruction, when a newly acquired scene referred to the new posture is merged with the already reconstructed world model.



(a) (b) (c)

Figure 1 - (a) RESOLV system, (b) sensing device, (c) sample 3D reconstructed environment [Butterfield *et al*, 97]

The Localisation algorithm for RESOLV was required to produce a (**x**,**y**,**θ**) posture estimate on planar indoor environments, with and without an initial estimate of the AEST posture. The map is only partially known - the goal of a RESOLV mission is to build up the environment model - and it is updated at each iteration. The map elements are the 3D surfaces extracted from the laser range data. The Localisation solution should act as a subsidiary element of the 3D reconstruction process described in [Leevers *et al*, 98] and [Sequeira, Ng, Butterfield *et al*, 98], thus it should use its maps and acquisition tools. Moreover, it must be fully functional in the manually operated trolley where odometry is unavailable. The localisation is performed while the AEST is still, and it should produce an estimate with less than 5 cm position error and 0.02 rad (≈1º) heading error, because of reconstruction requirements. These requirements are met by the proposed algorithm.

3. Related research

Several Laser Range systems have been developed for indoor localisation, under different world models and applications. Some solutions are briefly described below and compared with the RESOLV solution.

Some methods use the laser range data associated with other sensors, usually odometry, to compute incremental updates on the robot posture. Such is the case presented in [Dubrawski and Semiatkowska, 98], where significant features, such as straight lines, are extracted in different moments and the displacement between them is used to compute the robot's movement. This solution, although very robust, is insufficient for RESOLV since it requires an initial estimate and also the presence of odometry which is not available in RESOLV manual trolley.

A class of solutions assume the map is known beforehand; for instance, in [Crowley *et al*, 98] a subset of points in the map are associated to a numerical pattern ("Principal Components of Range

Data") extracted from the range profile at each of those points. A space is created upon these data and the posture estimate from a particular range scan is mapped into the space. A Kalman-Bucy filter is used to extract the most likely solution. This solution is inadequate for RESOLV because it requires a closed and fully described environment and also a set-up phase.

In [Vestli, 95] a method closer to RESOLV's is proposed. The laser data is transformed into a set of straight lines or circle arcs, which are combined to resemble natural landmarks, such as corners. If the robot displacement is approximately known, a feature extracted from the range data should lie in the vicinity of the correspondent feature on the map. Once the two features are matched the robot posture is known. In RESOLV's solution the range data is also transformed in a set of lines, albeit in a different manner, but since the map is previously unknown, the system must find its own "landmarks" and decide how to associate extracted landmarks to landmarks on the map.

A method inspired in the previous one is described in [Arras, Vestli, 98]. Straight lines, extracted from the rage data, are combined to form natural landmarks. Laser intensity is also measured to enhance the statistical representation of data and to complement or replace natural landmarks by artificial landmarks such as reflective adhesive bands. In RESOLV's solution the laser intensity is also used to enhance range data analysis but artificial landmarks are unavailable.

Briefly, RESOLV's solution differences are: it requires no initial posture estimate, thus odometry is a supplement, whereas it is necessary for other methods; it includes two complementary algorithms, enhancing its overall performance and minimising each algorithm's pitfalls, and, last but not the least, requires only a small fraction of the environment to be properly mapped.

In [Weckesser, Dillmann, 97] a solution for 3D mapping is presented. This work is fairly related to RESOLV's goals, although the paradigm is different: it uses a 3 camera video system as an acquisition tool. Surface boundaries are its distinctive features and range data is translated into a similar description. To perform localisation the lines extracted from the range scan are compared to the lines in the world model and a Mahalanobis distance criterion is used to find a match. This solution complies to the constraints of RESOLV system, except for the acquisition sensors. However, it should be interesting to try adapting it to RESOLV.

4. Frame Localisation Algorithm [FrameLoc]

The localisation data acquisition is based on a horizontal laser scan, which generates a 2D sequence of points centred around the AEST (Figure 2a). In turn, the reconstructed 3D model may be reduced to a sequence of 2D curves expressed in the world (inertial) reference, if the 3D surface representation is intersected by an horizontal plane (Figure 2b). Clearly, if the plane intersection is made at the laser scan height, one can assume the range data (Figure 2a) is similar to the 2D curves set extracted from the 3D model, albeit in a different form and with a different coordinate system.

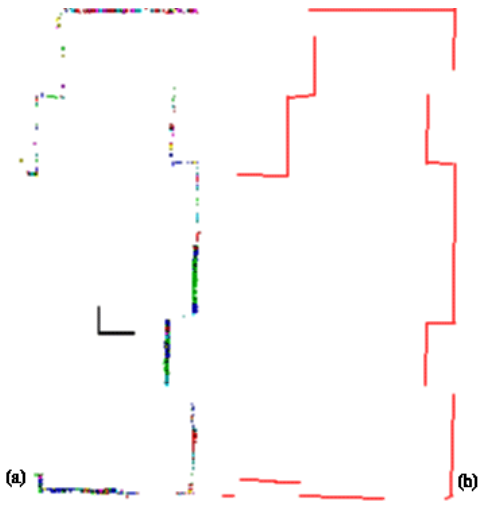


Figure 2 - (a) Sample laser scan with the AEST reference (b) 2D Map from reconstructed 3D model

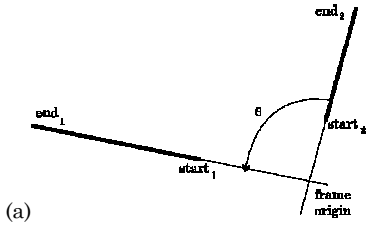
If the two data sets are expressed in a common form it is possible to compare them and find matching elements. Then, to solve the localisation problem, it is necessary to compute the transform relating the range scan coordinates to the 2D map coordinates.

To achieve a common representation, the Frame Localisation Algorithm (FrameLoc) introduces geometrical entities, termed frames, whilst the Approximate Localisation Algorithm (ApproxLoc) transforms the map data into a "simulated laser scan". This later algorithm is described in Section 6.

The FrameLoc first step is to extract parametric curves from the laser scan. The most often used are line segments, although it is also possible to use other parametric curves, biquadratics for instance. However, the current implementation is restricted to lines since most of the elements in the 3D models are planar surfaces. FrameLoc uses the planar surfaces available in the range scan and reconstructed scene while ignoring all non-planar surfaces.

The line extraction procedure is an auxiliary pre-processing phase and is beyond the scope of this paper. Nevertheless, it is worth saying that the line extraction is based on recursive iterative least squares and relies also on the reflectance analysis of the laser data. The line extraction for the 2D map is very simple because the original 3D model surfaces are already expressed by its parameters.

Once the 2D curves are extracted in the range data and in the map data, they are sorted by decreasing size and stored in two lists: the scan list and the map list. The elements in each list are then combined to form new geometric entities termed *frames*. A *frame* is defined by two non-parallel curves (usually lines) called **axis**[1] and the point where the two axis, or their extensions meet, called **origin**(Figure 3a). A *frame* has two types of parameters: five local parameters (Figure 3a), which are independent of the coordinates their axis are expressed in, and three global parameters relating the *frame* origin to the axis' coordinate system (Figure 3b).

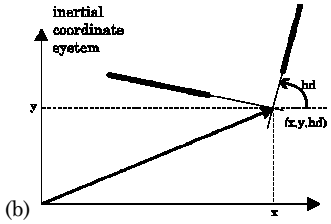


(a)

start1, start2: from the origin to the beginning of axis 1 and 2,

end1, end2: from the origin to the end of axis 1 and 2,

θ: frame inner angle



(b)

x, y, hd: frame origin x,y,heading-coordinates expressed in its reference: the world for the map frames and the AEST for the scan frames

Figure 3 - (a) Frame Local Parameters, (b) Frame Global Parameters

The five local parameters are: the distance from beginning of axis and end of axis to the origin, $\mathbf{start}_i$ and $\mathbf{end}_i$ for both axis (i=1,2) and the axis' inner angle **θ**. Extending FrameLoc to curves would require the curve coefficients to be added to the frame local parameters. The three global parameters enclose the information relating the frame to its coordinate system: given the position and heading of the frame origin (**x**, **y**, **hd**) and the set of local parameters it is possible to restore the axis in their original posture.

The first elements in the scan list are combined to produce the *scan frame list* and a similar procedure on the first elements of the map list generates the *map frame list*.

To test whether a scan frame corresponds to a specific map frame (Figure 4), only their local parameters are compared. Two frames constitute a possible match pair, if (1) is verified[2],

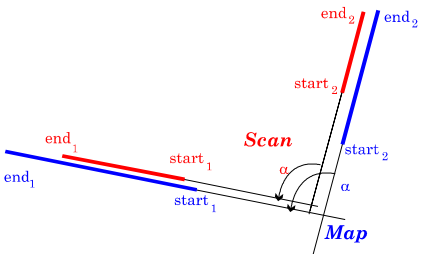


Figure 4 - Frame match

$$\begin{cases} start_1^{Map} \le start_1^{Scan} \le end_1^{Scan} \le end_1^{Map} \\ start_2^{Map} \le start_2^{Scan} \le end_2^{Scan} \le end_2^{Map} \\ \left|\theta^{Scan} - \theta^{Map}\right| < \delta \end{cases} \quad (1)$$

[1] In case biquadratics are used, the axis direction is defined by the chord joining the two endpoints.

[2] Since the axis indexing is arbitrary, a symmetric test is required in case axis 1 on a frame matches axis 2 on the other.

where δ is a heuristic limit to the angle difference. Using the canonical coordinate transform equation (2) applied to the frames global parameters, a candidate solution for the AEST posture (x_R, y_R, θ_R) [3] is found.

$$\theta_R = hd_1^{Scan} - hd_1^{Map}$$
$$\begin{vmatrix} x^{Map} \\ y^{Map} \\ 1 \end{vmatrix} = \begin{bmatrix} \cos\theta_R & -\sin\theta_R & x_R \\ \sin\theta_R & \cos\theta_R & y_R \\ 0 & 0 & 1 \end{bmatrix} . \begin{bmatrix} x^{Scan} \\ y^{Scan} \\ 1 \end{bmatrix} \quad (2)$$

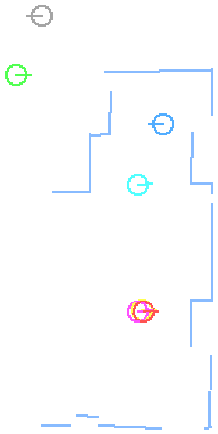


Figure 5 - Some possible postures

The frame matching procedure is applied iteratively to the elements on the frame lists, beginning with the longer frames, which are more significant, until a coherent pattern is apparent. In the process, isolated *false hits* are found due to environment symmetry (Figure 5) while the *true hits* concentrate in one common neighbourhood (around A and B in Figure 5). To compute the most likely solutions a weight clustering procedure is used, fusing all solutions in a single neighbourhood into one posture and discarding isolated postures, which are clearly false hits. The weight criterion is based on the product of scan axis sizes, because frames with long balanced axis are more likely to be properly defined than frames based in lines defined by a reduced number of points.

With the exception of very odd or symmetric environments or insufficient data, the algorithm converges swiftly to a reduced set of candidate clusters. The sorted lists implementation is crucial to restrain the algorithm's exponential nature. This is why the lines are sorted by decreasing size, frames generation begins with the top elements of the line lists and frames lists are also sorted by decreasing size. For instance, the current example was based on 107 scan lines and 38 map lines and 16 frames were sufficient to compute the solution. Moreover, the algorithm is iterative in the sense that, if a given number of frames is insufficient to compute a likely solution, new frames may be added to the frame lists and matched against existing or new ones and the new posture candidates added to the existing ones without losing any of the existing data.

The FrameLoc final result is a short list of postures, sorted by decreasing weight. Most often, the first element is the correct posture and takes more than 50% of the total weight. The clustering procedure is very stringent, therefore clusters A and B in Figure 5, although only 8 cm apart, are not merged. This option favours homogeneous clusters with small variance, at the expense of a longer list of postures and diluted weights. Even though FrameLoc correct solution is less clear, the Likelihood Test, described next, will access very accurately the quality of each possible posture.

5. Likelihood Test

The solutions computed by FrameLoc and ApproxLoc (described in Section 6) are based on subsets of the range scan and 2D map data. To access the quality of each candidate posture a thorough match is required, between the whole laser scan (Figure 2a) and the 2D map (Figure 2b) extracted from the reconstructed 3D model.

The first step is to express both profiles in a common form. For each candidate posture (x_R, y_R, θ_R), a *simulated laser scan* profile is evaluated using the 2D map assuming the AEST is located at (x_R, y_R, θ_R). This simulated scan technique will be used also for ApproxLoc.

[3] x_R, y_R and θ_R represent the AEST posture vector expressed in the world reference.

The simulated laser scan must replicate the main characteristics of the real laser scan, i.e., it is centred on the AEST and taken at the sensor head distance from the ground, it has the same angular width and number of points. However, the simulated laser scan doesn't consider the detailed sensor characteristics, namely sizeable footprint, reflectance, angle of incidence and surface transitions, since these approximations have a minor influence in the range error pattern. The range error of the laser sensors used in RESOLV project follow a normal distribution with zero mean and 1 to 3 cm variance [Sequeira, 96][4]. Thus, an histogram on the distance error (the modulus of the range error) is restricted to non-negative values and should resemble the normal curve with twice the amplitude. This histogram may be regarded as the laser scanner "signature".

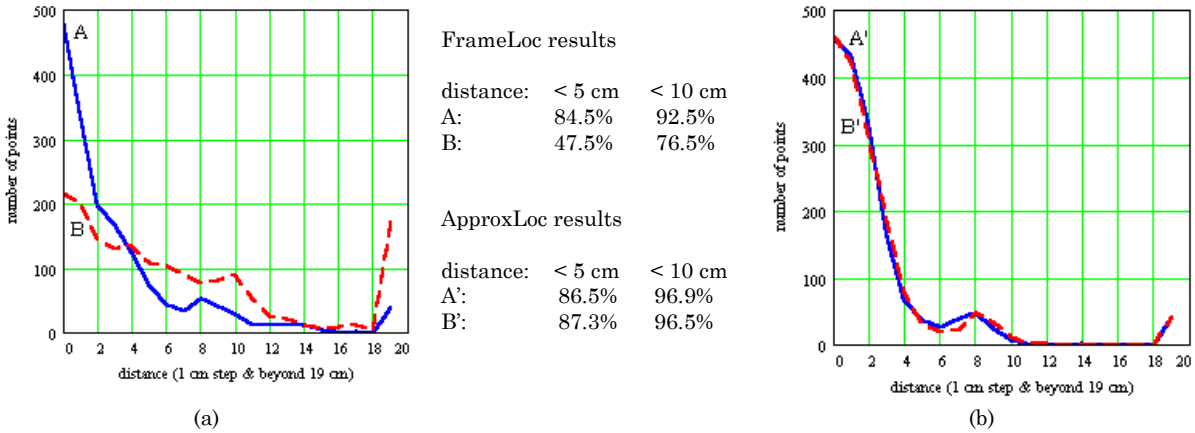
Once the real laser data and the 2D map data are expressed as "scans" it is possible to compute their point-to-point difference and store the results in a histogram. This measure establishes a likelihood criterion relating the two scans. If the posture estimate (x_R, y_R, θ_R) were exact, the difference between the real scan and the simulated scan was due only to the laser errors and to the inaccuracies in the simulated scan model. Drawing an histogram with the point-to-point distances should reveal an error pattern similar to the laser scanner signature.

The point to point distance measurement must account for invalid points that may exist in both "scans". The points in the simulated scan are invalid when it encompasses void areas in the reconstructed map. The real scan profile has some invalid points too, which are mainly due to specular reflections, device errors, time-outs and poor reflectance. All invalid points are removed and replaced by a flag. Only the pairs with both valid points are considered for the histogram.

After these preliminary steps, the distance histogram is computed: for each pair of valid points the Cartesian distance is computed and stored in 1 cm slots, up to a limit. Above that, all points are grouped in one common slot. If the likelihood measure falls below a given threshold the test fails (e.g., less than 50% pairs below 10 cm for Frame Localisation and less than 90% pairs below 5 cm for Approximate Localisation).

An example is shown to illustrate the characteristics of the Likelihood pattern. This experiment was made in an 12x6 meter laboratory room, with a 280º range scan sampled at 1800 points. The two solutions presented in Figure 5 correspond to the clusters A=(2.296m, 2.381m, 0.022rad (1.24º)) and B=(2.217m, 2.366m, 0.019rad (1.1º)) computed by FrameLoc[5].

The graphics in Figure 6 show the distance distribution classified in 1 cm slots from 0 to 19 cm and a final class above 19 cm. In Figure 6a the two lines represent the Likelihood of the solution computed by FrameLoc. Solution A is better, although B is already acceptable: 76.5% pairs are closer than 10 cm. The Approximate Localisation introduced in Section 6 (Figure 6b), updates A and B to A'=(2.285m, 2.371m, 1.25º) and B'=(2.285m, 2.371m, 1.35º) respectively. Comparing the two images it is clear how a slight posture update "pushes" most of the instances in the histogram to the left, increasing the Likelihood measure.



(a) (b)

Figure 6 - (a) Likelihood after Frame Localisation Algorithm, (b) Likelihood after Approximate Localisation Algorithm

[4] The AEST may be equipped with different Laser Scanners, hence the interval. This test used a 2 cm variance scanner.

[5] This case highlights the sensivity of the Likelihood Test, which is able to discriminate two postures only (8 cm,0.002rad(0.14º)) apart and reacts to a small correction such as A to A' (1.5 cm, 1.7×10^{-4}rad (0.01º)).

Up to around 4 cm, i.e. twice the range error variance, a good histogram, such as A, is close to the laser signature. A poorer solution, such as B, presents a reduced Likelihood measure, which can be enhanced by ApproxLoc up to the signature curve (from B to B'). From 4 cm to 19 cm, the histogram on Figure 6a accounts for the instances that may be fitted closer by the ApproxLoc. The last section includes the pairs farther from the sensor, and therefore very sensitive to angular errors. It also includes the residual pairs that do not match at all, because of sensor or map errors. The graph on Figure 6b shows the results after ApproxLoc. It is apparent that A' and B' are very close to the Laser Scanner signature, except for the residual pairs above 19 cm and a minor peak at 8 cm, due to map inaccuracies.

6. Approximate Localisation Algorithm [ApproxLoc]

The Approximate Localisation Algorithm (ApproxLoc) requires an initial posture estimate, (x_I, y_I, θ_I). This posture is used to compute a *simulated laser scan*, according to the method defined for the Likelihood test. The initial values are provided either by FrameLoc or by the AEST navigation system as described in [Castro *et al*, 98].

The operation principle of ApproxLoc is again based on a correspondence between pairs of points in the range scan and the simulated scan. Each point is defined by (x,y) coordinates; the range scan is centred on the AEST, whilst the simulated scan is centred on the initial posture estimate. Most often, the coordinates in the simulated scan do not match exactly their counterparts in the range scan, because of a posture estimation error. The update transform from the simulated range scan reference (SimSc) to the actual AEST posture can be computed using (2), rewritten as (3). This system has two equations and three unknowns, (x_P, y_P, θ_P). To obtain a third equation one could try to extract the orientation of a line segment instead of using single points. However, this process is very sensitive to errors; the solution taken was to consider **n** points simultaneously, as described by (4)

$$\begin{vmatrix} x \\ y \end{vmatrix}^{AEST} = \begin{bmatrix} \cos(\theta_P) & \sin(-\theta_P) & x_P \\ \sin(\theta_P) & \cos(\theta_P) & y_P \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}^{SimSc} \qquad (3)$$

$$\begin{vmatrix} x_1 \\ y_1 \\ \vdots \\ x_n \\ y_n \end{vmatrix}_{(2n\times 1)}^{AEST} = \begin{bmatrix} \cos(\theta_P) & \sin(-\theta_P) & x_P \\ \sin(\theta_P) & \cos(\theta_P) & y_P \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ y_1 \\ \vdots \\ x_n \\ y_n \end{bmatrix}_{(2n\times 1)}^{SimSc} \qquad (4)$$

Assuming the angular correction θ_P is small, it is possible to linearise the system around $\theta_P=0$ (5). Replacing the terms in θ_P as in (5) and rearranging (4) in order to the unknowns, (x_P, y_P, θ_P) yields two new matrices: **M**, the coefficient matrix (6a) and **Δ**, the difference matrix (6b).

$$\begin{matrix} \cos(\theta_P) \approx 1 \\ \sin(\theta_P) \approx \theta \end{matrix} \quad iff \quad \theta_P \approx 0 \qquad (5)$$

$$M = \begin{vmatrix} 1 & 0 & -y_1^{SimSc} \\ 0 & 1 & -x_1^{SimSc} \\ \vdots & \vdots & \vdots \\ 1 & 0 & -x_n^{SimSc} \\ 0 & 1 & -y_n^{SimSc} \end{vmatrix}_{(2n\times 3)} \qquad (6a)$$

$$\Delta = \begin{vmatrix} x_1^{AEST} - x_1^{SimSc} \\ y_1^{AEST} - y_1^{SimSc} \\ \vdots \\ x_n^{AEST} - x_n^{SimSc} \\ y_n^{AEST} - y_n^{SimSc} \end{vmatrix}_{(2n\times 1)} \qquad (6b)$$

The overdetermined system linearised after (4) may be solved with Singular Value Decomposition in the form of (7), where the coefficients of $\mathbf{M^T\Delta}$ and $\mathbf{M^TM}$ are defined in (8a) and (8b), respectively.

$$M^T\Delta = M^TM \cdot \begin{vmatrix} x_P \\ y_P \\ \theta_P \end{vmatrix} \qquad (7)$$

$$M^T\Delta = \left| \begin{array}{c} (x_{AEST} - x_{SimSc}) \\ (y_{AEST} - y_{SimSc}) \\ (x_{SimSc}(y_{AEST} - y_{SimSc}) - y_{SimSc}(x_{AEST} - x_{SimSc})) \end{array} \right|_{(3\times1)} \quad (8a)$$

$$M^T M = \left| \begin{array}{ccc} n & 0 & -y_{SimSc} \\ 0 & n & y_{SimSc} \\ -y_{SimSc} & x_{SimSc} & (x_{SimSc}^2 + y_{SimSc}^2) \end{array} \right|_{(3\times3)} \quad (8b)$$

In (8a) and (8b) all sums are from 1 to the number of points, **n**. It is now clear that (7) represents a (3x3) linear system which can be solved by standard Gauss elimination. The result, (x_P, y_P, θ_P), is the transform parameters relating the actual AEST posture where the range scan was taken, to the initial posture estimate, (x_I, y_I, θ_I), where the simulated laser scan was taken.

The AEST posture is updated according to (9),

$$\begin{aligned} \theta_{AEST} &= \theta_I + \theta_P \\ \left| \begin{array}{c} x \\ y \\ 1 \end{array} \right|_{AEST} &= \left[\begin{array}{ccc} \cos\theta_P & -\sin\theta_P & x_P \\ \sin\theta_P & \cos\theta_P & y_P \\ 0 & 0 & 1 \end{array} \right] . \left[\begin{array}{c} x_I \\ y_I \\ 1 \end{array} \right] \end{aligned} \quad (9)$$

The difficult issue in ApproxLoc is to detect **n** pairs of points (**n**≥2), where the exact counterparts in both "scans" can be established with a high degree of confidence. To detect these natural landmarks, a sliding window encompassing up to 10 points is swept along the real laser scan and the simulated scan, looking for consistent change of direction and/or surface discontinuities.

The search starts in the simulated scan since it has no noise, making the contour patterns easier to follow. When a landmark is detected at index ***k***, a similar feature is searched in the range scan at the vicinity of ***k***, for instance from ***k-5*** to ***k+5***. If the initial estimate is accurate enough and the 2D map extracted from the reconstructed 3D model includes enough landmarks within the scan width, a sufficient number of landmarks are detected (Figure 7a). Otherwise, the search fails, such as in (Figure 7b); in this particular case there are too few landmarks because the reconstructed model perceived from the top of the scene is very incomplete (Figure 2b). In such cases, detecting more landmarks would require a relaxed confidence criterion.

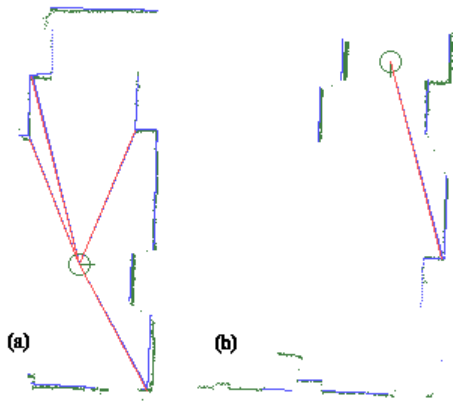


Figure 7 - Natural landmark detection (a) success: four pairs are found (b) failure: one pair is found while two are required

If the correspondence between landmarks in the simulated and range scans was perfect, one iteration of ApproxLoc would suffice to find the best estimate possible. Unfortunately, this is seldom the case and the algorithm is called iteratively, using the resulting posture of one iteration to initialise the next iteration. The posture computed at each iteration is submitted to the Likelihood Test to follow the evolution of the Likelihood measure. Usually, ApproxLoc converges to the final solution, i.e., the Likelihood histogram stabilises, within three to five iterations.

7. Operation Flow

The Localisation operation flow depends on the presence of odometry data. If an initial posture estimate from odometry is available, then ApproxLoc is called. Otherwise, FrameLoc must be called to compute a first estimate for the AEST posture which is fed into ApproxLoc.

The simple operation sequence 1) *FrameLoc* → 2) *LklTest* → 3) *ApproxLoc* → 4) *LklTest* , solves the vast majority of cases. However, some difficulties subsist especially while the map is very sparse or if the scan resolution or accuracy is diminished. Moreover, calling FrameLoc is seldom necessary when odometry is available. Thus, a loop structure (Figure 8) was created for efficiency, accuracy and robustness sake.

The start point of the Localisation cycle depends on the presence of odometry. If there is no odometry data the process starts at Frame Localisation, otherwise the posture estimate read from the odometers is used to compute the Simulated Scan, which is entered in the ApproxLoc, together with the real Range Scan. The resulting posture estimate is sent to the Likelihood Test to check if the two scans match. In case of success the algorithm ends, otherwise, FrameLoc is called.

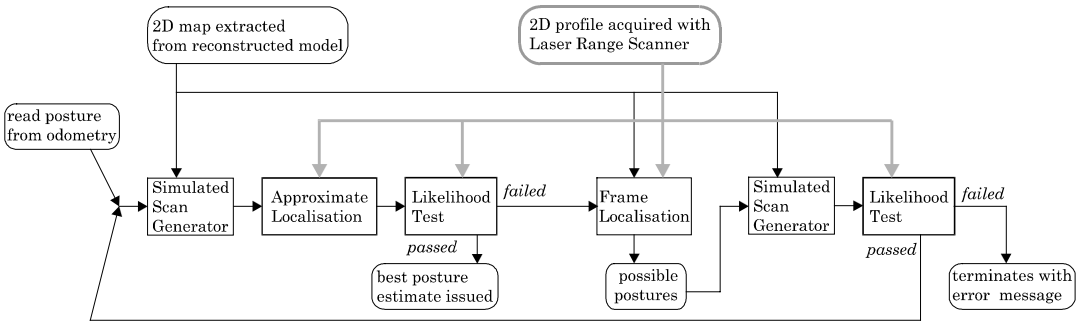


Figure 8 Compound algorithm operation flow

FrameLoc operation is based on the environment 3D model and the 2D laser range scan only, thus, it is irrelevant whether ApproxLoc has been called before. The results from FrameLoc are a list of possible postures which are passed, one by one, to the Simulated Scan generator and then to Likelihood Test. The approved postures are sent to ApproxLoc, while the others are discarded.

To avoid an endless loop after ApproxLoc, a flag indicates if FrameLoc has already been called; in that case the algorithm ends with failure. In future extensions a more sophisticated control loop would be inserted at this stage.

It is interesting to notice that a single range scan is used for all the operation loop. However, if the algorithm fails, it is possible to repeat it taking a scan at a different height. This is a likely extension to the compound algorithm and will be discussed in the next section.

8. Conclusions, known issues and extensions

This paper presents a novel approach for indoor localisation using Laser Range Scanner. It requires only a partial map and no initial localisation estimate. Because one single algorithm would fail to operate properly in some conditions, two complementary algorithms were developed. The presented solution proved adequate for 3D model reconstruction, because it ignores large unknown areas, keeping its "anchor" on the available reconstructed 3D model. Accuracy requirements were exceeded and (2 cm, 0.01 rad (0.6º)) maximum errors are expectable in standard conditions.

The algorithm uses computer resources with modesty. There are no large arrays, only comprehensive object lists. Since all repetitive tasks include tests and processing of linear equations and systems of reduced dimension, the algorithms are swift. The only computation intensive task is the line extraction algorithm (not described) even though it was developed in an efficient iterative form.

However, some pitfalls subsist. One major cause of concern is the akin nature of the localisation sensor and the map acquisition sensor. If the Laser Scanner fails to detect a mirror and maps a second scene *behind* the mirror, it is quite likely that the localisation will repeat the error. Strictly

speaking, this is not a localisation error since the model includes the *virtual* scene. Nevertheless, it is an undesirable feature. The line extraction algorithm (not described) includes some statistical analysis using laser reflectance data to overcome this problem, in particular for mirrored windows. Also, if some surfaces in the scene are too dark, smooth or very detailed, the 3D reconstruction program will fail to extract these surfaces, leading to void areas in the 3D model. However, if there is a sufficient number of well defined surfaces the algorithm will "anchor" to them for localisation.

A subtler problem arises on non-horizontal floors. The 2D map extracted from the 3D model is independent of the AEST location iff the plane intersection is parallel to the ground. So, unless the AEST is equipped with an inclinometer of any type, it is restricted to piece-wise horizontal floors.

Finally, problems arise in highly symmetric environments, such as square or round rooms or long corridors with evenly spaced doors, which occur more often. If the corridor length is greater than the Laser Scan operating range, the algorithm will find more than one equally likely solution separated by the distance which correspond to the corridor's repeating feature period.

The algorithm presented may be extended in several ways. One natural extension is the generalisation of the frame elements to biquadratic curves. On another direction, it is also desirable to include a richer control structure in the loop, such as automatically choosing the scan height, repeating localisation at different heights and matching the results, etc.

The major extension to the algorithm is to compute the localisation while the AEST is moving. The author is convinced that this task will require a new approach, using a narrow laser scan, anchoring the localisation to a few carefully selected landmarks.

References

[Arras,Vestli,98] Kai O. Arras and Sjur J. Vestli, *Hybrid, High-Precision Localisation for the Mail Distributing Mobile Robot System MOPS*, Proc. IEEE Int. Conference Robotics and Automation, pp. 3129-3134, Leuven, Belgium, May 1998.

[Butterfield *et al*,97] Stuart Butterfield, Kia Ng, David Hogg, *The RESOLV Texture-Mapping Module*, School of Computer Science, University of Leeds. Report 97.48, December 1997.

[Castro *et al*,98] José Castro, Vítor Santos, M.Isabel Ribeiro, *A Multi-Loop Robust Navigation Architecture for Mobile Robots*, Proc. IEEE Int. Conference Robotics and Automation, pp. 970-975, Leuven, Belgium, May 1998.

[Crowley *et al*,98] Crowley, J.L, F. Wallner, B. Schiele, *Position Estimation Using Principal Components of Range Data*, Proc. IEEE Int. Conference Robotics and Automation, pp. 3121-3128, Leuven, Belgium, May 1998.

[Dubrawski,Semiatkowska,98] Artur Dubrawski and Barbara Siemiatkowska, *A Method for Tracking Pose of a Mobile Robot Equipped with a Scanning Laser Range Finder*, Proc. IEEE Int. Conference Robotics and Automation, pp. 2519-2523, Leuven, Belgium, May 1998.

[Leevers *et al*,98] David Leevers *et al*, *An Autonomous Sensor for 3D Reconstruction*, 3rd European Conf.erence on Multimedia Applications, Services and Techniques, ECMAST 98, Berlin, Germany, May 98.

[RESOLV,Web] RESOLV Web site at http://www.scs.leeds.ac.uk/resolv/, includes publications, description of the platforms and tools and also sample reconstructed environments.

[Sequeira,96] Vítor Sequeira, *Active Range Sensing for Three Dimensional Environment Reconstruction*, *Ph.D. Thesis*, Instituto Superior Técnico, Technical University of Lisbon, Portugal, December 1996.

[Sequeira,Ng,Butterfield *et al*, 98] V. Sequeira, K. C. Ng, S. Butterfield, J. G. M. Gonçalves, D. C. Hogg, *Three-Dimensional Textured Models of Indoor Scenes from Composite Range and Video Images*, In R.N. Ellson and J.H. Nurre, editors, Proc. SPIE, Three-Dimensional Image Capture and Applications, Vol. 3313, to appear 1998.

[Vestli,95] Sjur Vestli, *Fast, accurate and robust estimation of mobile robot position and orientation, Ph.D Thesis*, Eidgenössische Technische Hochschule Zürich, Switzerland, December 1995.

[Weckesser,Dillmann,97] P. Weckesser, R. Dillman, *Modeling Unknown Environments with a Mobile Robot*, Proc. Symposium on Intelligent Robotic Systems SIRS'97, pp. 69-76, Stockholm, July 1997.