

Improving area center robot navigation using a novel range scan segmentation method.

José Manuel Cuadra Troncoso¹, José Ramón Álvarez-Sánchez¹, Félix de la Paz López¹, and Antonio Fernández-Caballero²

¹ Dpto. de Inteligencia Artificial - UNED - Madrid (Spain).
(Spain).{jmcuadra,jras,delapaz}@dia.uned.es

² Departamento de Sistemas Informáticos-UCLM-Albacetecaballer@dsi.uclm.es

Abstract. When using raw 2D range measures to delimit the border for the free area sensed by a robot, the noise makes the sensor to yield a cloud of points, which is an imprecise border. This vagueness pose some problems for robot navigation using area center methods, due to free area split points locations. The basic method, when locating split points, does not take into account environmental features, only the raw cloud of points. In order to determine accurately such environmental features we use a novel range scan segmentation method. This method has the interesting characteristic of being adaptive to environment noise, in the sense that we do not need to fix noise standard deviation, even different areas of the same scan can have different deviations, e. g. a wall besides a hedge. Procedure execution time is in the order of milliseconds for modern processors. Information about interesting navigational features is used to improve area center navigation by means of determining safer split points and developing the idea of dynamic split point. A dynamic split point change its position to a new feature if this new feature is considered more dangerous than the one marked by the split point.

1 Introduction

In range sensor measurements noise sources may be very disparate, like: intrinsic sensor error, sensor calibration errors and the environment itself. The noise portion due to the environment cannot be neglected, but it can be the main noise source. For example, laser measures over a garden hedge carry much more error, in standard deviation terms, than measures over a plain wall. Although devices technical specifications usually carry information about intrinsic device noise and calibration procedures, we have no prior information about environmental noise, so we need to make assumptions about its behavior. If we want to develop a procedure being able to handle worlds with surfaces of different characteristics at a time we have to reduce or to relax our assumptions about noise. As more general our noise model as more flexible and robust navigation we get.

Filtering noise we can get an ideal representation of the world as a set of line segments and curved segments, this procedure is known as scan segmentation or line extraction. In section 2 we give a brief review of the subject. The proposed line extraction procedure is explained in section 3 through section 5.

In section 3 a sequential line estimation procedure is presented. Using directly the line equation in polar coordinates, we have developed an Extended Kalman Filter (EKF) for sequential line estimation, being noise variance unknown and depending on measured distance. Our filter formulation is equivalent to regression models formulation, so we get all the statistical properties, test for hypothesis contrast, etc., known for regression models theory.

Kalman filters need initial values for their parameters, the only place for collecting this information are the measures. A set measures, called seed, is used to get an initial estimation for the filter parameters, a clustering procedure, section 4, is used to determine good seeds positions. This clustering procedure is based in scale-space techniques, we have develop a statistic filter to reduce the number of contour curves due to noise, making the procedure more reliable and efficient.

In section 5 the designed EKF ability is used for detecting outliers to detect segments ends, overlapped segments compete for common measures and finally, due to filter statistical properties, test from model regression theory to merge similar segments are applied.

Section 6 presents some uses of segmentation results for making robot navigation safer when using center of area methods: better split points location and moveable split points.

2 Line extraction review

It is not intended in this section to make a detailed review of the subject (interested reader is referred to [9]) but just to highlight the aspects directly motivating our work or more related to it. First question attracting our attention was the existence of distance thresholds and the second one was noise effects and models.

Distance thresholds are usual criteria to determine if a point belongs to a segment based on distance between points. Some known methods using distance threshold are: Split and Merge [5], Iterative End Point Fit [4], Successive Edge Following [3], Random Sample Consensus Algorithm (RANSAC) [6] and Line Tracking [14]. Our objections to distance thresholds are two. First, taking measures radially there is no limit, theoretically, for distance between two consecutive segment points, it depends on incidence angle. Second, points belonging to non concurrent segments could be misclassified if they are close to each other.

Second question to consider is noise models, specially in probabilistic methods. When noise variance is used in the noise model, it is supposed to be constant and known, in section 1 we commented the possible big influence of the reflexion surface on the measure error. We have not found comments in literature mentioning explicitly the possibility for unknown, changing or high variances, but we can give some references where fixed known variance is theoretically required [18], or used in experiments [4,9]. Normally this variance is the value found in device technical specifications plus some constant.

Our work use Kalman filter (KF) estimation as in [18,4,16]. Also it uses a clustering procedure prior to line estimation, this procedure is used in several

works cited previously. Clustering provides a representation of the world at a coarse scale, the problem is to select the proper scale level to represent it in an adequate way. This question leads to the concept of natural (most significant) scales [12]. The selected method has been Interval Tree by Witkin [17], this is the pioneer paper about scale-space. Scale space theory [8] provides a multi-scale representation of a signal formed by smoothed versions of the signal. Smoothing is performed convoluting the signal with gaussian kernels with different variances, the changing scale parameter. Features are normally discovered as extrema or inflexion points of the signal, thus they are represented as zero-crossing level curves in the scale-space representation, see fig. 1 in section 4. Witkin's Interval Tree procedure associates a rectangle to each curve and defines a stability criterion for rectangles, most stable rectangles define natural scales.

3 Line estimation

3.1 Measures model

The problem about using rectangular coordinates with noisy measures taken radially is that both variables, x and y , are random and correlated [7], and there is not a non-stochastic independent variable. In this case many of the properties and results from regression theory. The only case where stochastic independent variables are allowed is when they have to take equally spaced values. This is the case for angles in a laser scan, so polar coordinates are an adequate statistical frame for line estimation from laser measures.

The equation of a line, not crossing the coordinates origin (pole), in polar coordinates is:

$$r = f(\varphi) = \frac{\rho}{\cos(\varphi - \theta)} \quad -\frac{\pi}{2} < \varphi - \theta < \frac{\pi}{2} \quad (1)$$

being r the measured distance, φ the angle of the measure, ρ line distance to pole, and θ angle between x axis and the line.

Our measure model is: $r_n = E[r_n] + \varepsilon_n$, being $E[r_n]$ the expected value of the measure and ε_n the noise. Noise has a gaussian probability distribution with zero mean and variance $\sigma_n^2 \propto h^2(E[r_n])$, being h a known function. In our experiments we use a noise standard deviation proportional to the expected measure, so $\sigma_n^2 = \sigma^2 E^2[r_n]$, being σ an unknown constant representing the deviation per unit measure, or roughly the noise proportion.

3.2 EKF for line estimation in polar coordinates

As we commented in section 2, in order to detect outliers it is necessary to estimate the line sequentially and to have a test for outliers detection. Sequential linear regression estimation studied by Plackett in the 50's [11]. Later Peña [10], saw the equivalence between Plackett formulation and KF formulation, see eq. 2 for EKF formulation, so every result, property, test, etc., from regression theory

is applicable. For example KF formulation is independent of σ^2 and prediction test from regression theory may be used as a outlier detection test. He gave a procedure to sequentially compute the residual sum of squares (RSS), needed for σ^2 estimation, reaching a linear temporal complexity for the process.

To deal with non-constant variances weighted least squares may be used [7]. When $\sigma_n^2 = \sigma^2 h^2(E[y_n])$, being y_n the independent variable and h a known function, a two rounds procedure can be applied due to expected values are usually unknown. Firstly, ordinary least squares, over the whole data set, are applied to obtain estimated measures \hat{y}_i and a second least squares round, over the whole data set, is applied weighting data using $w_i = w(\hat{y}_i) = \frac{1}{|h(\hat{y}_i)|}$. Using Peña's KF formulation and filter independence of σ^2 , matrix covariance noise remains $\frac{1}{w_i^2}$, so data have not to be weighted if this matrix is used. In this work we are going apply the two rounds procedure using KF, but rounds just involve the new incoming data.

During initial experiments, with linear models, we observed that if weights were significantly less than 1 at initial stages could lead to a bad result. In these cases Kalman gains were too small, making filter response to innovation small too. The solution was to scale weights, by a factor ω , in order to make initial weight values near or over 1, for example, $\omega = \max(|h(y_s)|)$, being y_s the seed. Estimation of parameters and their covariances matrix are unaffected by this scale factor. Noise deviation estimation is scaled by the same factor.

Equation 1 represents a non-linear model, so we need an EKF. The EKF formulation, adapted from [10], is simply:

$$\begin{aligned}\beta_n &= \beta_{n-1} \\ r_n &= \frac{\rho_n}{\cos(\varphi_n - \theta_n)} + \varepsilon_n\end{aligned}\tag{2}$$

first equation is the state equation and second one is the measure equation. The state parameters are $\beta = \begin{pmatrix} \rho \\ \theta \end{pmatrix}$ and ε_n is gaussian distributed with zero mean and covariances matrix $\sigma_n^2 = \sigma^2 h^2(E[r_n])$.

4 Clustering

The scale-space procedure, described in section 2, can be applied for range scan segmentation. Fig. 3 a) shows a simulated world composed by plain walls and a hedge, where noise was generated following the model explained in subsection 3.1 with $w_i = |h(\hat{r}_i)| = \frac{1}{\hat{r}_i}$, so noise is proportional to measure. Measures on the walls have a standard deviation proportional to the measure being of 2 cm. at 10 m., standard deviation is 10 times higher on hedge.

In fig. 2 we can see super-imposed the polar representation of that world. Segments edges are extrema and inflexions points of the curve, so first and second derivative can be used to find them. Zero-crossing level curves for first and

second derivative of the scale-space surface are shown in fig. 1. Specially for the second derivative noise yields a lot of curves. This curves could magnify low level environmental features yielding a too fragmented world representation. An adequate selection of scale levels, specially the lowest level could remove eventually the majority of the small curves. We propose a filtering procedure that can adapt itself to the mean amount of environmental noise, removing noise.

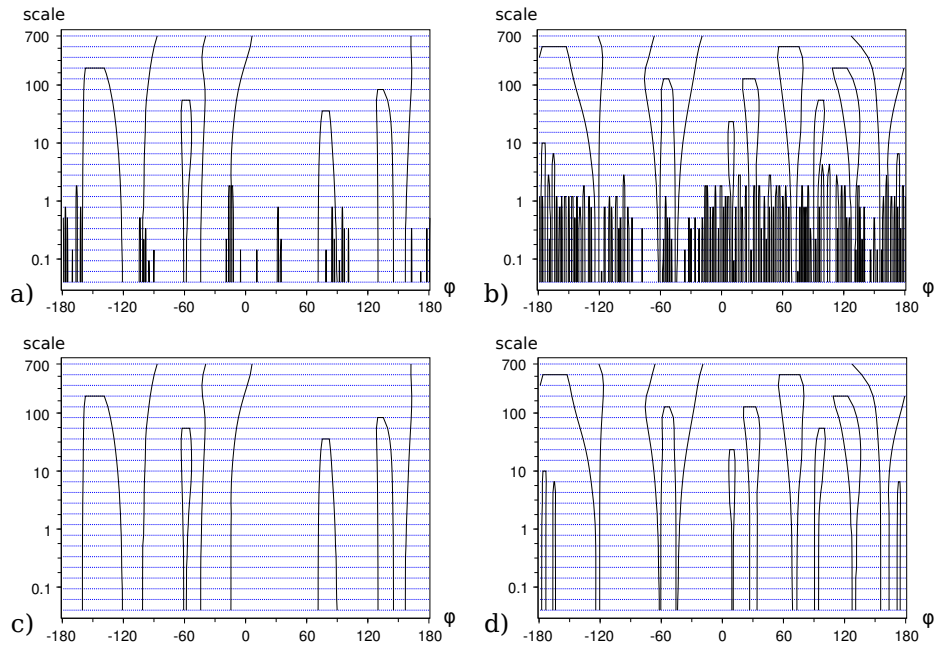


Fig. 1. Zero-crossing level curves for first, a) and c), and second derivatives, b) and d), of scale-space surface. In c) and d) shown the curves remaining after noise filtering. Horizontal blue lines are discretized scale-space parameter values.

The filtering procedure is based on a characterization for the curves generated by noise. In fig. 1 a) and b) we can appreciate that noise generated curves are usually significantly narrower and lower than curves generated by high level features, these are most of curves showed in 1 c) and d). So we looked for a basic characterization based on areas under curves. Experiments using fixed and simple patterns of noise gave an approximate interval for the maximum area of noise generated curves that roughly separates both types of curves. Values for noise deviation in $[0, 0.05]$ were used, this upper ends represents 50 cm . at 10 m . We have choose inside that interval the value 50 as area upper threshold. Curves with area under the threshold are preliminary classified as noise generated. To refine the search for noise generated curves we compute the mean \bar{h} , and the

standard deviation s_h for selected curves heights, setting $\bar{h} + ks_d$, as the definitive upper threshold, usual values for k are between 2 and 3. The value \bar{h} is used as lowest level for the Interval Tree procedure, see section 2.

The results of the filtering procedure are shown in fig. 1 c) and d), only some noise generated curves remain. These curve are located at the hedge segment where the noise is significantly higher than in the rest. After filtering, the Interval Tree procedure is performed in order to find the most stable environmental features, results are shown in fig. 2.

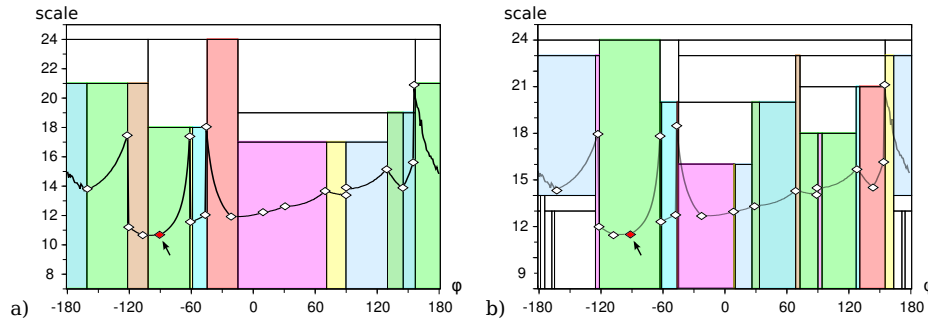


Fig. 2. Stable features, colored rectangles, discovered from level curves after filtering. a) First derivative and b) second one. The difference between lower limits in y axis is due to adaptation to noise, second derivative enhances noise effects. World representation in polar coordinates has been superimposed. No features generated by noise has been discovered and only one real feature has been missed, it is marked with an arrow at -90° .

Features locations discovered using both derivatives have to be merged, nearby features a mixed in one and groups of missing measures are taking in account using their ends as features. Isolated missing values are interpolated and groups are replaced with convenience noiseless values. Finally, the discovered features are considered as fuzzy ends of segments, then sets of consecutive measures are extracted from each segment core to obtain the seeds for KF initial values estimation.

5 Merging

Applying the clustering procedure explained in section 4 we get a set of seeds to obtain initial estimation for KF bootstrapping. In order to achieve a balanced estimation we have to introduce new points in the filter consecutively at both seed ends. Experiments showed a significant reduction in estimation errors maxima when doing balanced estimation.

The selection of an adequate number of measures to include in the seeds poses two questions. First, as longer the seed the smaller the initial error estimation is.

Second, as longer the seed, the lesser the number of points we can test for outlier detection. Take into account that usually test critic values are high if sample is very small, in our case we needed a least 6 measures. Intensive simulations showed that 9 or more measures give a good accuracy, even 5 are not bad. So we need at least 15 measures to start a reliable outlier detection. This number looked to high to us so we studied the possibility for including seed points in KF procedure.

Reusing the seed carries the problem of statistical dependence, thus we considered to obtain the initial estimations from the seed but using a different regression framework other than least squares and, then, to test experimentally the effects. For initial estimation the Repeated Median [15] was chosen, in median based regression the target is to minimize the sum of absolute residuals instead of the squares. The experiments shown a small increment in final estimation errors when including the seed but also a significant increment in the ability for early outliers detection. So we propose to include seeds in KF procedure using median based regression for initial estimations.

Before starting KF estimation procedure a condition for segment end detection has to be proposed. Two types of outliers exist: proper outliers, points belonging to the segment but a bit far from it, and improper outliers, points belonging to other segments. Finding only one outlier as segment end detection is not enough, but waiting for two consecutive outliers is too restrictive and can lead to failing to detect many connected segments ends when noise variance is high. As an intermediate solution our proposed end detection condition is: two outliers in two or three consecutive measures, so that if α is the significance level for outlier detection, then, $2\alpha^2$ is, approximately, the significance level for end detection.

In order to stop estimation, segments ends have to be found at both seed sides. We had to impose two more conditions for estimation stop. First, if process has reached the beginning of the previous seed or the end of the next one, estimation is stopped at the corresponding seed side. These stop points are set in order to avoid excessive overlapping between connected segments as result of high noise levels. Second, if the estimated noise standard deviation is higher than a limit, the filter is considered as out of control, the limit was set to 0.1, equivalent to $1m.$ at $10m.$

When all KF estimations are concluded, a search for groups of points not belonging to any segment, holes, is performed. One new segment is estimated per each hole. At this point, overlaps between segments may exist, so we need a criterion for assigning a group of consecutive points to each segment. Comparing noise variance estimation for each segment, when the point was added to the estimation procedure, was the most accurate criterion for resolving overlaps we have found. Sign of difference between variances at overlap center is computed then the first change in difference sign is searched at both center sides, if it is found, the overlap is split, if not, one of the segments gets the overlap. When overlaps are resolved a new search for holes is performed, but in this case estimation are restricted to holes limits.

Specially when noise is high, clustering procedure could yield more than one seed for a “true” segment, so we need a procedure for merging similar segments. We have selected Chow’s test [7], from regression theory, designed for comparing parameters of regressions over different sets of data. In fig. 3 an example of the whole process is shown.

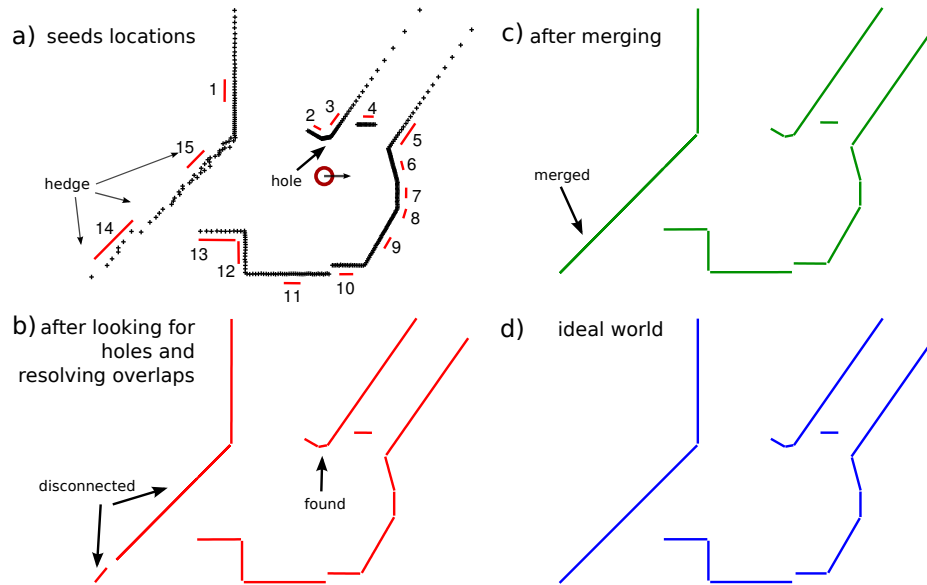


Fig. 3. Pictures sequence showing the application of the estimation and merging procedure described in section 5. World is composed by plain walls and one hedge where the noise is 10 times higher, world dimensions are $17.5\text{ m} \times 12.5\text{ m}$. Red(black) circle in a) represents robot and the arrow at robot center indicates orientation $\varphi = 0$. A picture of the ideal world used to generate the measures is shown for comparison purposes.

To analyze the performance on a computer a preliminary C++ implementation of the procedure, yet in development, was tested on laptop with a Core Duo 2 processor at 2.28 Ghz. New robots are being shipped with similar processors and several well known old models can be now upgraded. For 181 measures with noise standard deviation $\sigma = 0.002$, 2 cm . at 10 m . the procedure takes about 2.5 milliseconds, for 360 about 6 ms. and for 720 about 9.5 ms., these times suggest a complexity near linear order. For $\sigma = 0.02$, times increase about 25 – 30%. Taken into account that an usual laser sampling period takes 100 ms or longer, proposed procedure execution only takes a bit of this time.

6 Improving area center navigation

The basic area center method for robot navigation was proposed previously in [1,2,13]. Basically the robot follows the center of area of its perceived free area while it is accessible. Robot velocity vector is proportional to distance of area center in robot local coordinates. A basic concept in area center navigation is the split points, where the perceived free area is split in two sectors when area center becomes inaccessible and one of them is selected to follow its area center. Successive split points can be set, while selected sector aperture is wide enough, allowing robot to pass across narrow passages and to drive in crowded environments. For goal reaching a virtual split point can be used to select a sector in goal direction.

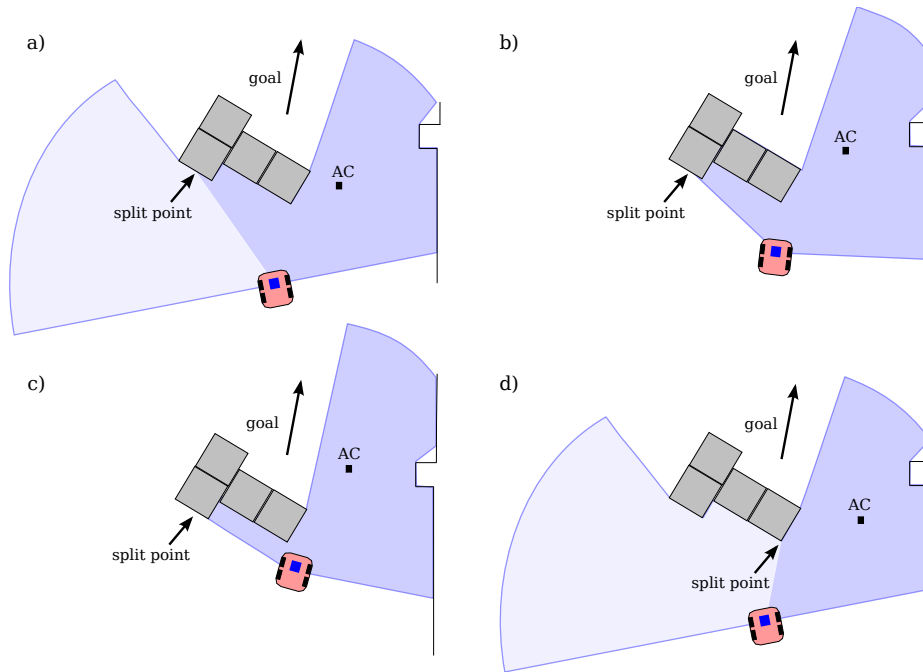


Fig. 4. Bad split point selection example case. In a) the center of area for the full free area becomes inaccessible in the point marked with an arrow and it becomes the split point. Due to goal direction right sector is selected and robot follows the new area center marked with AC b). But area center path is not enough far away from obstacle leading robot to a crash in c). It would be better to set the split point at the nearest corner of the obstacle as in d), so area center path is far away enough from obstacle. The selected sector has dark blue color and the unselected one has a lighter color.

Correct split points localization is very important for a safe navigation. Using raw range measures the basic method usually provides a safe navigation,

but there exists some situations, specially when doing goal tracking, where the position of the split point is not adequate yielding a dangerous situation. Sample situations depicted in figs 4 and 5 are taken from experiments described in [13]. The problem arise from selecting a sector, after split, with highly protruding corners. Selecting as split point the nearest segment end of those segments with some end between the area center and the bisector of original sector plus an offset, we can get sectors with less noticeable protuberances near the robot path, see fig. 4 d).

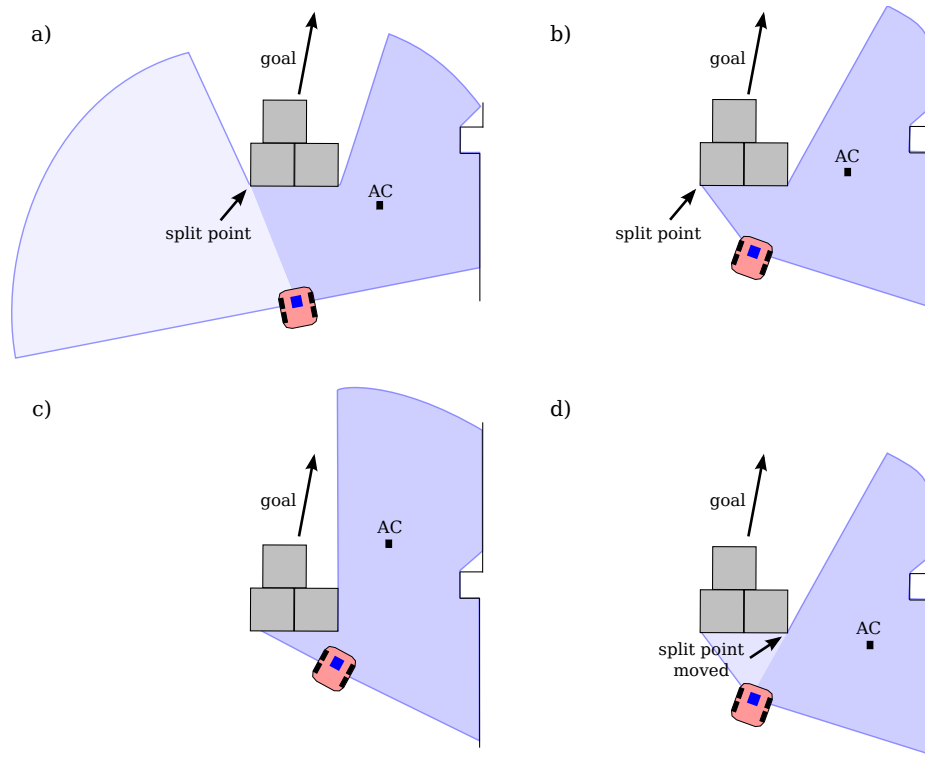


Fig. 5. Dynamic split points. In a) robot is slightly closer to the segment end marked with the arrow than to the other segment end. Then when area center becomes inaccessible inside the obstacle, the marked end is selected as split point and right sector is selected due to goal direction, this would lead to a dangerous situation b) and c), like in fig. 4, with the basic split point strategy. It would be better to change split point to the other end when the other segment end is closer to robot than split point, d), so area center path is far away enough from obstacle. The selected sector has dark blue color and the unselected one has a lighter color.

In order to get a safer and more robust navigation, specially in crowded or dynamic environments, we gave a step forwards enhancing split points behavior

must be improved. Until new split points were static, but to response quickly to environment changes it would be better if they were dynamic. Even if the split point was well located at first time it is possible that a new protuberance appears in robot path without making area center inaccessible. In this case no split process is triggered, a basic solution for this problem is to change the position of a split point. So, if some segment end, located between split point and the bisector of current sector plus an offset, is closer to the robot, such segment end becomes the split point. This situation is depicted in fig. 5, and similar to one described in fig. 4, selected sector contains big protuberances. Changing split point to a new corner avoids the problem, see fig. 5 d).

Test performed in a robot simulator shows a better performance in robot navigation when using the improved method than the basic one. Robot exhibits an improved ability in turning corners in crowded environments and tracking goals, in a safe and efficient way. We reproduced the scenario of our real robot experiments from [13] in the simulator. When using the basic method we can observe many situations as those described previously in this section, a majority of these dangerous situations were avoided using the improved method.

7 Conclusions and future work

In this work we have presented a novel method for line extraction from range measures in polar coordinates. It has been designed for a noise model in which the measure errors have a standard deviation proportional to a known function of the expected measure, the rate constant may be unknown. This last fact allows our method to deal with different noise levels in the same scan. The method may be classified as a Line Regression with Clustering method and it is composed by three elements: a line regression method using EKF, a clustering procedure to look for adequate places in the scan that are useful to start EKF, and a merging procedure for similar adjacent segments.

The key for designing a filter that can deal with our noise model has been filter formulation. We have used and extended a filter formulation that is equivalent to regression models theory, so every result, property, test, etc., from regression theory are at our disposal. This fact provides tools for: line parameters estimation, noise variance estimation, estimators covariances matrix estimation, outliers detection, similar segments merging, etc., grounded in a well known and developed theory. As a future research, tools for scan matching in SLAM can be developed with the help of regression theory. Filter has linear time complexity.

We have used scale-space techniques for clustering by developing a filter for noise in order to get less fragmented features in the environment. Filter is based on a coarse characterization of the set of zero-crossing level curves generated by noise in the scale-space representation, followed by a statistical selection inside the set.

The clustering procedure usually provides adequate places, seeds, to start EKF estimation. Using outliers detection we can establish a segment end detection criterion. Possible overlap between segments are solved and similar segments

are merged. Our C++ implementation provides execution times fast enough for use in robots.

We need to test extensively the whole process, at this moment only EKF has been tested extensively, and to make comparisons with other segmentation methods. Test in real world has been started using laser SICK-LMS 200 and Kinect device. Research on extending the method to curves and to 3D measures are at initial stages of development.

The first use for our segmentation method has been to improve the center of area method for navigation. By means of adequate split points selection, using segments ends, it is possible to get a more reliable and safer navigation.

References

1. José R. Álvarez Sánchez, José Mira Mira, Félix de la Paz López, and José M. Cuadra Troncoso. The centre of area method as a basic mechanism for representation and navigation. *Robotics and Autonomous Systems*, 55(12):860–869, 2007.
2. José Ramón Álvarez Sánchez, Félix de la Paz López, José Manuel Cuadra Troncoso, and José Ignacio Rosado Sánchez. Partial Center of Area Method Used for Reactive Autonomous Robot Navigation. In José Mira, José Manuel Ferrández, José R. Álvarez, Félix de la Paz, and F. Javier Toledo, editors, *Bioinspired Applications in Artificial and Natural Computation*, volume 5602 of *LNCS*, pages 408–418. Springer Verlag, June 2009.
3. G. A. Borges. A split-and-merge segmentation algorithm for line extraction in 2-d range images. In *Proceedings of the International Conference on Pattern Recognition - Volume 1*, pages 1441–, Washington, DC, USA, 2000. IEEE Computer Society.
4. Geovany Araujo Borges and Marie-José Aldon. Line extraction in 2d range images for mobile robotics. *J. Intell. Robotics Syst.*, 40:267–297, July 2004.
5. J. A. Castellanos and J. D. Tardós. *Laser-based segmentation and localization for a mobile robot*, volume 6 of *Robotics and manufacturing: Recent trends in research and applications*. ASME Press, New York, 1996.
6. Martin A. Fischler and Robert C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24:381–395, June 1981.
7. J. Johnston and J. DiNardo's. *Econometric Methods*. McGraw-Hill/Irwin, 4th edition, 1996.
8. Tony Lindeberg. Scale-space for discrete signals. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12:234–254, 1990.
9. Viet Nguyen, Stefan Gächter, Agostino Martinelli, Nicola Tomatis, and Roland Siegwart. A comparison of line extraction algorithms using 2d range data for indoor mobile robotics. *Auton. Robots*, 23:97–111, August 2007.
10. Daniel Peña. On internal robustification of Plackett-Kalman algorithm for recursive estimation of regression linear models. *Trabajos de Estadística y de Investigación Operativa (in spanish)*, 36(1):93–106, 1985.
11. R. L. Plackett. Some theorems in least squares. *Biometrika*, 37((1-2)):149–157, 1950.
12. Paul L. Rosin. Determining local natural scales of curves. *Pattern Recognition Lett*, 19:63–75, 1994.

13. José Ramón Álvarez Sánchez, Félix de la Paz López, José Manuel Cuadra Troncoso, and Daniel de Santos Sierra. Reactive navigation in real environments using partial center of area method. *Robotics and Autonomous Systems*, 58(12):1231–1237, 2010.
14. A. Siadat, A. Kaske, S. Klausmann, M. Dufaut, and R. Husson. An optimized segmentation method for a 2d laser-scanner applied to mobile robot navigation. In *Proceedings of the 3rd IFAC symposium on intelligent components and instruments for control applications*, 1997.
15. Andrew F. Siegel. Robust regression using repeated medians. *Biometrika*, 69(1):pp. 242–244, 1982.
16. R.M. Taylor and P.J. Probert. Range finding and feature extraction by segmentation of images for mobile robot navigation. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 1, pages 95–100 vol.1, April 1996.
17. A. Witkin. Scale-space filtering: A new approach to multi-scale description. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '84.*, volume 9, pages 150–153, March 1984.
18. Sen Zhang, Lihua Xie, and Martin David Adams. Feature extraction for outdoor mobile robot navigation based on a modified gauss-newton optimization approach. *Robotics and Autonomous Systems*, 54(4):277–287, 2006.