[10] Q. Li and D. Rus, "Navigation protocols in sensor networks," *ACM Trans. Sens. Netw.*, vol. 1, no. 1, pp. 3–35, Aug. 2005.

[11] E. Martinson and D. W. Payton, "Lattice formation in mobile autonomous sensor arrays," in *Swarm Robotics* (Lecture Notes in Computer Science Series 3342), E. Sahin and W. M. Spears, Eds. New York: Springer-Verlag, 2004, pp. 98–111.

[12] K. O'Hara and T. Balch, "Distributed path planning for robots in dynamic environments using a pervasive embedded network," in *Proc. Int. Conf. Auton. Agents Multi-Agent Syst.*, Jul. 2004, pp. 1538–1539.

[13] K. O'Hara and T. Balch, "Pervasive *Sensor-less* networks for cooperative multi-robot tasks," presented at the Int. Symp. Distrib. Auton. Robot. Syst., Toulouse, France, Jun. 2004.

[14] K. O'Hara, V. Bigio, E. Dodson, A. Irani, D. Walker, and T. Balch, "Physical path planning using the Gnats," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2005, pp. 709–714.

[15] K. O'Hara, V. Bigio, S. Whitt, D. Walker, and T. Balch, "Evaluation of a large scale pervasive embedded network for robot path planning," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2006, pp. 2072–2077.

[16] L. Panait and S. Luke, "A pheromone-based utility model for collaborative foraging," in *Proc. Int. Conf. Auton. Agents Multi-Agent Syst.*, 2004, pp. 36–43.

[17] H. V. D. Parunak, S. Brueckner, and J. Sauter, "Synthetic pheromone mechanisms for coordination of unmanned vehicles," in *Proc. Int. Conf. Auton. Agents Multi-Agent Syst.*, 2002, pp. 449–450.

[18] D. Payton, M. Daily, R. Estowski, M. Howard, and C. Lee, "Pheromone Robotics," *Auton. Robots*, vol. 11, pp. 319–324, 2001.

Convergent Smoothing and Segmentation of Noisy Range Data in Multiscale Space

Martin Adams, Tang Fan, Wijerupage Sardha Wijesoma, and Chhay Sok

Abstract—With few exceptions, most of the existing noise reduction and data segmentation algorithms are only suited to image data. Therefore, an adaptive smoothing algorithm, with model-based masks, within a scale space framework is proposed for range data in this paper. This algorithm smoothes range data that conform to predefined, geometric models, while leaving other data points unaffected. The convergence of the algorithm in yielding dominant features is shown based on its compliance with the anisotropic diffusion concept. The weights of the smoothing masks are adaptively calculated according to the Mahalanobis distances between range data and model-based predictions. These behave as the diffusion coefficient in the anisotropic diffusion equation, thus satisfying the requirements of the causality criterion that no new features are introduced from fine to coarse scales. The computational complexity of this algorithm is examined and compared to that of the well-known RANSAC feature extraction algorithm. Unlike RANSAC, it has the advantage that the computational complexity is less affected by increasing the order of the model, and is independent of the number of model outliers. The proposed algorithm can be used to smooth range data in multiscale space by increasing the number of smoothing iterations. Robust, robot-occlusion-invariant features are then easily extracted from the smoothed data by least squares fitting algorithms.

Index Terms—Adaptive smoothing, anisotropic diffusion, feature extraction, scale space.

I. INTRODUCTION

Due to their reliability and accuracy, laser range finders are commonly used in mobile robotics. For autonomous navigation, it is necessary to automatically, and continuously, detect a consistent set of landmarks (features) from progressive range-bearing data sets, for data association, and ultimately localization and mapping purposes. Feature detection must be reliable in the presence of sensor noise and data outliers so that false alarms and missed detections are minimized.

In this paper, an adaptive smoothing algorithm, using model-based masks, is proposed to smooth model-compliant range data, while preserving discontinuities. The masks use geometric models to predict range values for each scan point in the observation space by using neighboring measurements. A theoretical justification for the smoothing algorithm is based on its compliance with the anisotropic diffusion concept, originally proposed by Perona and Malik [1]. In vision, this concept utilizes a diffusion coefficient to effectively stop diffusion, or smoothing, across large intensity variations, so that edges are sharpened. The model-based mask proposed here calculates weights according to the Mahalanobis distance between a predicted range value and its measured value. *While in vision, the diffusion coefficient is a nonnegative function of the magnitude of the local intensity gradient (or, put another way, the "noncompliance" of the pixel intensities to a constant, smooth value), it is suggested here that, for range data, this coefficient should be a function of the Mahalanobis distance, since this is a positive number indicating the "noncompliance" of the data to a selected geometric model.* The mask weights become the diffusion coefficient, and will adaptively reflect the suitability of the geometric models used for segmenting range data. However, smoothing only once can fail to find the correct, dominant features due to the difficulty of defining suitable model noise variances and a Mahalanobis distance threshold. The mask can therefore, be used to smooth range data iteratively, in multiscale space. With increase of iteration number, the smoothed range data is represented in a coarse scale, where measurement noise is highly reduced but the dominant features are still preserved.

In this work, all range data predictions, corrections, and error representations are carried out *in the observation space*. The resulting prediction equations are nonlinear, as would also be the case for most models (except straight line) in Cartesian space. The advantage offered, however, is that range variance calculations can be based on known sensor range variances, and that these values, as well as Mahalanobis distances, need not be transformed into other (e.g., Cartesian) spaces.

II. PREVIOUS WORK

Many algorithms formulated for smoothing and segmentation in vision research are general enough to be adapted to range data. This section, therefore, describes some of the literature that has adapted these methods to range data segmentation, and then explains some methods that have been specifically developed for range data only.

The *Hough transform* is a tool that identifies global patterns in an image space, by recognition of local patterns (ideally a point) in a transformed parameter space [2]. Forsberg *et al.* introduced a *range weighted* version in which a window based on the estimated shortest distance between the sensor and the line is defined to improve the efficiency of the computations [3]. Chan and Tam, apply the transform to a predefined grid map and attempt to match cells between a currently sensed local map and the global map [4]. These articles, have collectively demonstrated that its advantages are: 1) it is tolerant to gaps in feature boundary descriptions; 2) it is relatively unaffected by range noise; 3) it is useful for computing a global description of a feature, given local measurements; and 4) The number of features need not be known *a priori*. Its disadvantages are: 1) it is computationally intensive

because of the transformation of each point to, and the quantization of the Hough Space and 2) as the number of parameters of the feature model increases, the processing increases exponentially due to the increase in the dimension of the Hough space.

RANSAC is an algorithm for the robust fitting of models in the presence of data outliers [5]. As well as its extensive use in vision, it has been successfully applied to range data [6], [7], where it has been demonstrated that the advantages of this algorithm are: 1) it has a good tolerance to outliers and 2) it is generic for many types of features once a feature model is defined. There are, however, some disadvantages: 1) it is a randomized estimator—successive runs on the same data set can produce different results and 2) if the threshold on the minimum required number of inliers is set too low (to speed up the algorithm), the estimate can be very poor.

Horowitz and Pavlidis introduced a recursive Split-and-Merge algorithm for both image and range data segmentation [8]. The algorithm first constructs a straight line between the data end-points, and measures the deviation of all points from that line. If that deviation exceeds a threshold, the line is split, and two new lines are formed. The algorithm proceeds until all points have small enough deviations, but it only segments data based on a predefined threshold and does not use any noise reduction.

Mahalanobis distance, recursive,[1] line-model-based methods for range point segmentation were introduced by Adams in [9]. Later, this method was adopted by Sen *et al.* [10] in a Gauss–Newton least squares line detection approach, and Roumeliotis [11] in which different Mahalanobis distance thresholds were applied for feature detection and filter initialization. The Mahalanobis distance represents how close an estimated scan point is to the line defined by its predecessors. Here, the aforesaid work is generalized by providing a smoothing technique based on general geometric masks, which utilize a range point's neighbors. The new algorithm can be iterated in multiscale space,[2] and is proven to converge under the concept of anisotropic diffusion, to finding the dominant geometric features within an environment.

A. Anisotropic Diffusion

Methods to smooth camera images, without removing detail include the *bilateral filter* [12] and the *Mean Shift Algorithm* [13]. These algorithms filter data, based on both the spatial and intensity relationships between pixels, to reduce noise, but preserve image detail. *Anisotropic diffusion*, is also a useful tool for the multiscale description of images and image segmentation, [1]. Anisotropic diffusion offers the advantage over the aforesaid techniques in that a nonnegative monotonically decreasing function $g()$ of the differential of the image intensity can be chosen to control the smoothing [14]. It is this flexibility that we exploit to adapt it in a useful manner, to range data. The anisotropic diffusion equation is given by

$$\frac{\partial I(x,y,t)}{\partial t} = \text{div}[g(\|\nabla I(x,y,t)\|)\nabla I(x,y,t)] \qquad (1)$$

where $I(x,y,t)$ is the image intensity at pixel location (x,y), and t represents the scale. To apply anisotropic diffusion to range data, it is necessary to choose the appropriate decreasing function $g()$ and ∇I, the differential function of I with respect to x, y. In the vision literature, this decreasing function is often a Gaussian. The predicted measure-

Fig. 1. Geometric smoothing at a particular scale (iteration number) t. The black filled circles represent range data, and the light shaded circles represent model-based predictions at the central mask point. (a) Line model [see (2) and (3)]. (b) Circular model smoothing.

ment (pixel intensity) is the variable and the actual measurement, or previously smoothed value, is the mean value of the variable.

III. ADAPTIVE SMOOTHING OF RANGE DATA

This section is divided into six parts. In Sections III-A and III-B, geometric model based masks are derived. The weights of the masks will be adaptively calculated in Section III-C according to the Mahalanobis distance between measured and predicted range data from the models. In Section III-D, it is shown that the masks can be used to smooth range data in multiscale space by translating them iteratively over a scan. Section III-E explains how the necessary, associated variances of each range point can be updated and Section III-F provides proof of convergence of the algorithm, by demonstrating its analogy with anisotropic diffusion.

In an image, the difference between neighboring pixel intensities can be a function of many parameters, such as the direction of the local illumination, the reflectivity of objects in the environment, as well as their shape. For range data acquired from an active sensor, the relationship of adjacent range points is, however, totally dependent on the shape of the scanned environment. Hence, any smoothing mask for range data should reflect the geometric properties of the environment. Simply smoothing range data with a Gaussian mask, based on range changes, results in undesirable regions of constant range, which does not necessarily reflect the true environment. A smoothing mask can be formulated for any chosen surface representation, within an environment, using the method which follows. For demonstration purposes here, masks are derived, which attempt to smooth line and circular segments only, while automatically leaving other parts of the scan intact.

If a range scan can be received as a batch,[3] each scan point in a segment can be predicted by its neighboring scan points, assuming they belong to a predefined segment model.

A. Line-Model-Based Mask

In the proposed *line-based* mask,[4] two pairs of neighboring points (A, B) and (D, E) in Fig. 1(a) are used to calculate two predicted range values $O\widehat{C}_{-1}$ and $O\widehat{C}_{+1}$ for smoothing the observed range value OC. During smoothing, an estimate of the range OC is updated, based

---

[1]The term "recursive" is used to differentiate this work from other published methods, which first record complete data scans, and then perform smoothing/segmentation on the data as a batch. In contrast, this paper estimates whether points are model-compliant as the data is received.

[2]In this paper the scale space corresponds to the iteration number.

[3]As with the data recorded from the commonly used Sick LADARs.

[4]This mask is a modification of the algorithm originally presented by the first author in [9], in which predictions based only on (2) were made for sequentially recorded range points, at a single scale.

on the other mask values, and the mask is shifted from the first to the last scan point. The current, central mask range value will be referred to as $d_0^t(n_\theta)(= OC)$ and the predicted mask range values as $\widehat{d}_{-1}^t(n_\theta)(= O\widehat{C}_{-1})$ and $\widehat{d}_{+1}^t(n_\theta)(= O\widehat{C}_{+1})$.[5] The superscript t indicates the iteration number, and when $t = 0$, the range points are the direct outputs of the range measuring sensor. Weights will be determined for the predicted range values $\widehat{d}_{-1}^t(n_\theta)$ and $\widehat{d}_{+1}^t(n_\theta)$, based on the straight line compliance of point C with the two range values counterclockwise (A and B) and clockwise (D and E) of it, respectively.

It should be noted that any two points within the mask of Fig. 1(a) can be used to predict a third, colinear one along ray OC—e.g. points B and D could be used to predict another point that intersects ray OC, as was considered in [15]. However, only $d_{\pm1}^t(n_\theta)$ are used here, for two reasons: 1) This will be shown to comply with the concept of anisotropic diffusion in Section III-F, so that dominant line detection is guaranteed and 2) Each range point is used only once in the predictions and the correlations between the final estimates can be handled correctly.

The two prediction equations based on the line model are

$$\widehat{d}_{-1}^t(n_\theta) = \widehat{d}_{OC_{-1}}^t = \frac{d_{OA}^t d_{OB}^t}{2d_{OA}^t \cos\gamma - d_{OB}^t} \tag{2}$$

$$\widehat{d}_{+1}^t(n_\theta) = \widehat{d}_{OC_{+1}}^t = \frac{d_{OE}^t d_{OD}^t}{2d_{OE}^t \cos\gamma - d_{OD}^t} \tag{3}$$

where d_{OA}^t = the range from O to A at iteration t, etc., and γ is the assumed constant angular scanning increment. d_{OA}^t, d_{OB}^t and d_{OD}^t, d_{OE}^t (and, in fact, d_{OC}^t) are the previously smoothed range values[6] and $\widehat{d}_{OC_{\pm1}}^t$ are the predicted range values at iteration t. Predictions (2) and (3) can be considered to be the expected values of a dynamic, nonlinear state transition equation, relating the predicted state at scale t, to the previously smoothed state at scale t—i.e.,

$$d_{-1}^t(n_\theta) = F_L(d_{OA}^t, d_{OB}^t, \gamma) + \nu_L^t,$$
$$d_{+1}^t(n_\theta) = F_L(d_{OE}^t, d_{OD}^t, \gamma) + \nu_L^t \tag{4}$$

where F_L is the nonlinear, line transition function, defined by (2) or (3). ν_L^t can be considered to be the system model noise, in this case indicating any unmodelled deviation which we may choose to allow, from the strictly straight line model. Here, ν_L^t will be considered to represent zero mean, Normally distributed noise with variance independent of the scale t, and hence will be referred to as ν_L. Its use will be explained further in Section III-C.

B. Circular-Model-Based Mask

A circular model can also be derived for the smoothing of scan points belonging to circular objects. Fig. 1(b) shows that two triplets of neighboring range points (F, G, H) and (K, L, M) can be used to calculate two predicted range values $d_{-1}^t(n_\theta) = O\widehat{J}_{-1}$ and $d_{+1}^t(n_\theta) = O\widehat{J}_{+1}$ for smoothing the previously smoothed/observed range value OJ.

Again, the prediction equations for $\widehat{d}_{\pm1}^t(n_\theta)$ can be considered to be the expected values of the nonlinear equations

$$d_{-1}^t(n_\theta) = F_C(d_{OF}^t, d_{OG}^t, d_{OH}^t, \gamma) + \nu_C^t$$
$$d_{+1}^t(n_\theta) = F_C(d_{OM}^t, d_{OL}^t, d_{OK}^t, \gamma) + \nu_C^t \tag{5}$$

where F_C is a nonlinear, circular transition function,[7] and ν_C^t is a system model noise term, analogous to ν_L^t.

C. Calculation of the Mahalanobis Distance

Two predictions and a previously smoothed/recorded point are now available for further smoothing. These three points, are to be weighted to produce a more "smoothed" range value. Many ways of calculating these weights exist. In this paper, a weighting method, is devised that:

1) is able to correctly take into account individual range variance values, with each recorded range point;
2) makes use of the variance of the innovation,[8] which is a measure of the "appropriateness" of the model fit to neighboring range data; and
3) provably conforms to the anisotropic diffusion equations, proposed by Perona and Malik [1], so that smoothing takes place only with model inliers, and does not affect the outliers in an adverse way.

The variances of the line and circle predictions, $\Sigma_{\pm1}^t(n_\theta)$, can be found by linearizing (4) and (5) about their current, smoothed values e.g., in the case of the line model

$$\Sigma_{-1}^t(n_\theta) = \left(\frac{\partial F_L}{\partial d_{OA}^t}\right)^2 \Sigma_{OA}^t(n_\theta) + 2\left(\frac{\partial F_L}{\partial d_{OA}^t}\right)$$
$$\times \left(\frac{\partial F_L}{\partial d_{OB}^t}\right) \Sigma_{OA,OB}^t(n_\theta) + \left(\frac{\partial F_L}{\partial d_{OB}^t}\right)^2 \Sigma_{OB}^t(n_\theta) + Q_L \tag{6}$$

where $\Sigma_{-1}^t(n_\theta)$[9] is the predicted variance of $d_{-1}^t(n_\theta)$ at iteration t with a model variance $Q_L = E[(\nu_L)^2]$, which is a design parameter controlling the flexibility of the line model. $E[\cdot]$ is the expectation operator. In general, a low value of $Q_{L/C}$ means that the line/circle model is expected to be strictly adhered to by the data. A high value produces a more tolerant system to model outliers. The effects of $Q_{L/C}$ will be discussed in Section VI. $\Sigma_{OA}^t(n_\theta)$ and $\Sigma_{OB}^t(n_\theta)$ are the previously calculated variances of range values d_{OA}^t and d_{OB}^t, and $\Sigma_{OA,OB}^t(n_\theta)$ is the cross correlation between range d_{OA}^t and d_{OB}^t. For the laser range finder used here, a starting ($t = 0$) range standard deviation of 4 cm is assumed ($\Sigma_{OA/OB}^0(n_\theta) = 0.016$ m^2)[10], and $\Sigma_{OA,OB}^0(n_\theta) = 0$, since the initial range measurements are assumed independent.

Innovation values $v_{\pm1}^t(n_\theta)$ can be derived, which represent the differences between the model-based predicted ranges $E[d_{\pm1}^t(n_\theta)]$ [from (4) or (5)] and the smoothed range values from the previous iteration $d_0^t(n_\theta)$

$$v_{\pm1}^t(n_\theta) = E[d_{\pm1}^t(n_\theta)] - d_0^t(n_\theta). \tag{7}$$

The variances of these innovations, $\Sigma_{v_{\pm1}}^t(n_\theta)$, are simply

$$\Sigma_{v_{\pm1}}^t(n_\theta) = \Sigma_{\pm1}^t(n_\theta) - 2\Sigma_{\pm1,0}^t(n_\theta) + \Sigma_0^t(n_\theta) \tag{8}$$

where $\Sigma_{\pm1,0}^t(n_\theta)$ are the correlations between $\widehat{d}_{\pm1}^t(n_\theta)$ and $d_0^t(n_\theta)$, assumed zero when $t = 0$, to be updated at each iteration. The Mahalanobis distance (variance of the innovation)

$$\epsilon_{\pm1}^t(n_\theta) = \left(v_{\pm1}^t(n_\theta)\right)^2 / \Sigma_{v_{\pm1}}^t(n_\theta) \tag{9}$$

is used as a measure of how well the previously smoothed points conform to each model.

---

[5]For example, if a complete $360°$ scan is taken with 720 points, then $0 \leq n_\theta < 720$, and each range sample is separated by an angle $\gamma = 0.5°$.

[6]Or on the first iteration ($t = 0$), the recorded range data points themselves.

[7]For brevity, this function is omitted here, but the reader may refer to [10].

[8]The difference between the model-based predicted range values, and the smoothed range values from the previous iteration.

[9]A dual equation can be derived to find $\Sigma_{+1}^t(n_\theta)$.

[10]Note that individual range variances for each scan point could be incorporated into $\Sigma_{OA}^0(n_\theta)$ and $\Sigma_{OB}^0(n_\theta)$, etc., if they can be estimated [9].

D. Adaptive Smoothing in Multiscale Space

In vision, the anisotropic diffusion coefficient is a nonnegative function of the divergence of the pixel intensities. It is proposed here, for range data, that this coefficient can be a function of the aforesaid Mahalanobis distance, which is a direct function of the difference between the predicted and central mask range values, since this is a positive number indicating the divergence of the data from the selected model.

Weights are now formed in order to smooth the predicted range points at scale $t + 1$. These weights behave as the diffusion coefficient, $g(\|\nabla I\|)$, in (1). They should be a decreasing function of the Mahalanobis distance and obey the conditions for well-posed anisotropic diffusion, discussed in [16].[11] Here the weights $w_i^t(n_\theta)$ will conform to the diffusion coefficient suggested by Perona and Malik [1][12]

$$w_i^t(n_\theta) = \exp\left(-\frac{\epsilon_i^t(n_\theta)}{2t^2}\right), \quad \text{for } i = \pm 1, \quad w_0^t(n_\theta) = 1 \quad (10)$$

where t is the scale (iteration number). It should be noted that with this choice of function, well-posed anisotropic diffusion is guaranteed if $t \geq \sqrt{\epsilon_i^t(n_\theta)/2}$.

For example, for the line model, either of the points $\widehat{C}_{\pm 1}$, with predicted ranges $\widehat{d}_{\pm 1}^t(n_\theta)$, will have weights $w_{\pm 1}^t(n_\theta)$ according to (10). Hence, these are the two weights that must satisfy the anisotropic diffusion (1). There is still the central mask range value $d_0^t(n_\theta)$, and its weight $w_0^t(n_\theta)$ will be equal to 1. The smoothed range value for this iteration $d_0^{t+1}(n_\theta)$ will be the weighted sum of the two predicted values and the previously smoothed value

$$d_0^{t+1}(n_\theta) = \frac{\sum_{i=-1}^{+1} w_i^t(n_\theta)d_i^t(n_\theta)}{\sum_{i=-1}^{+1} w_i^t(n_\theta)} = \sum_{i=-1}^{+1} \overline{w}_i^t(n_\theta)d_i^t(n_\theta) \quad (11)$$

$$\sum_{i=-1}^{+1} \overline{w}_i^t(n_\theta) = 1 \quad (12)$$

where $\overline{w}_i^t(n_\theta)$ is a normalized weight. For simplicity, only $\overline{w}_i^t(n_\theta)$ is used in the following analysis.

E. Range Data Variance Update

To continue the smoothing iterations, the updated variance of each smoothed point $\Sigma_0^{t+1}(n_\theta)$ must be determined, which, from (11), is given by

$$\Sigma_0^{t+1}(n_\theta) = \sum_{i=-1}^{+1} (\overline{w}_i^t(n_\theta))^2 \Sigma_i^t(n_\theta) + 2(\overline{w}_{-1}^t(n_\theta)\overline{w}_{+1}^t(n_\theta)$$

$$\times \Sigma_{-1,+1}^t(n_\theta)) + 2\sum_{i=-1}^{0} \overline{w}_i^t(n_\theta)\overline{w}_{i+1}^t(n_\theta)\Sigma_{i,i+1}^t(n_\theta)$$

$$(13)$$

where changes in (11), w.r.t. the weights have been ignored, since it can be shown that these terms are negligible, particularly as t increases. The recursive computation of the necessary correlation terms ($\sum_{-1,+1}^t(n_\theta)$ and $\sum_{i,i+1}^t(n_\theta)$ in (13) is possible, if (4) and (5) are reduced to first order vector-matrix system form, and linearized about their predicted values, followed by the prediction of the full system error covariance

[11]The condition given in [16] is that if $\phi(\|\nabla I\|) = \|\nabla I\|g(\|\nabla I\|)$, then well-posed anisotropic is attained if $\phi'(\|\nabla I\|) \geq 0$.
[12]For a discussion of other possible functions, see [16].

matrix, as shown in [9]. Note that the full matrix, is maintained, to ensure that information is not reused either within a particular, or at different scales.

F. Proof of Convergence

According to (11), the change in range w.r.t. scale is

$$d_0^{t+1}(n_\theta) - d_0^t(n_\theta) = \overline{w}_{-1}^t(n_\theta)(d_{-1}^t(n_\theta) - d_0^t(n_\theta))$$
$$+ \overline{w}_{+1}^t(n_\theta)(d_{+1}^t(n_\theta) - d_0^t(n_\theta)). \quad (14)$$

In [1], Perona and Malik proposed the anisotropic diffusion equation (15), which they proved satisfied the requirements of the causality criterion that no new features are introduced from fine to coarse scales in the scale-space

$$\frac{\partial d_0^t(n_\theta)}{\partial t} = \nabla[\overline{w}_{\pm 1}^t(n_\theta)\nabla d_0^t(n_\theta)]$$
$$= \nabla\overline{w}_{\pm 1}^t(n_\theta)\nabla d_0^t(n_\theta) + \overline{w}_{\pm 1}^t(n_\theta)\nabla^2 d_0^t(n_\theta). \quad (15)$$

For the smoothing models proposed in this paper, $\nabla d_0^t(n_\theta)$ is the change in predicted range at scale t w.r.t. index n_θ and $\nabla\overline{w}_{\pm 1}^t(n_\theta)$ is the change in weight w.r.t. n_θ. When considering the recorded scan in the clockwise direction (range $d_{-1}^t(n_\theta)$ predicted before range $d_0^t(n_\theta)$ is considered), $\overline{w}_{\pm 1}^t(n_\theta)$ must be replaced by $\overline{w}_{-1}^t(n_\theta)$ in (15), and

$$\frac{\partial d_0^t(n_\theta)}{\partial t} \triangleq d_0^{t+1}(n_\theta) - d_0^t(n_\theta) \quad (16)$$

$$\nabla d_0^t(n_\theta) \triangleq \widehat{d}_{+1}^t(n_\theta) - d_0^t(n_\theta) \quad (17)$$

$$\nabla \overline{w}_{\pm 1}^t(n_\theta) \triangleq \overline{w}_{+1}^t(n_\theta) - \overline{w}_{-1}^t(n_\theta) \quad (18)$$

$$\nabla^2 d_0^t(n_\theta) = d_{+1}^t(n_\theta) - 2d_0^t(n_\theta) + d_{-1}^t(n_\theta). \quad (19)$$

If the scan is considered in the anticlockwise direction (range $d_0^t(n_\theta)$ considered after $d_{+1}^t(n_\theta)$ is predicted), then $\overline{w}_{\pm 1}^t(n_\theta)$ must be replaced by $\overline{w}_{+1}^t(n_\theta)$ in (15), and

$$\nabla d_0^t(n_\theta) \triangleq \widehat{d}_{-1}^t(n_\theta) - d_0^t(n_\theta) \quad (20)$$

$$\nabla \overline{w}_{\pm 1}^t(n_\theta) \triangleq \overline{w}_{-1}^t(n_\theta) - \overline{w}_{+1}^t(n_\theta). \quad (21)$$

Note that $\overline{w}_0^t(n_\theta)$ is not used as the function $g()$ in the anisotropic diffusion equation or in the expression for $\nabla\overline{w}_{\pm 1}^t(n_\theta)$, as it is only weights $\overline{w}_{\pm 1}^t(n_\theta)$ which are the required functions of the model Mahalanobis distances.

If the aforesaid equations are substituted into (15), it can be seen that the model mask (11) conforms to the anisotropic diffusion concept, meaning that the same causality criterion will apply to the smoothing of the range data also.

IV. SEGMENTATION AND FITTING

Segmentation is also possible, based on the Mahalanobis distance, as shown in a previous paper [9]. For the masks considered here, each smoothed range point will have two associated Mahalanobis distances $\epsilon_{\pm 1}^t(n_\theta)$ (9). Both of these can be compared with a threshold based on a χ^2 distribution to determine whether the center point C^t or J^t (Fig. 1) is a model outlier w.r.t. each neighboring data set. Here, a threshold of 5.02 was chosen theoretically giving a probability of 97.5% correct, model-compliant segmentation. Each segment should then be fitted with its geometric model. The intersections of the lines, or centers and radii of the circles, are then used as occlusion-invariant features.
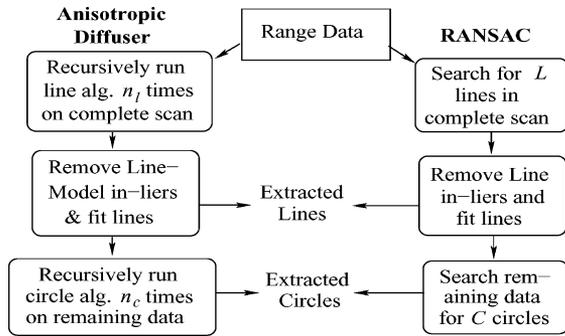
Fig. 2. Schematic diagram showing the feature extraction methods used and compared with the proposed anisotropic diffuser and RANSAC.

V. Computational Complexity

The computational complexity is defined in terms of the number of range points per scan N, the number of points W needed to initiate the model (two for lines, three for circles), and the scale t. At each iteration, (4) [or (5)] and the predicted variances [two lots of (6), or its equivalent] must be calculated. This requires the calculation and storage of $2 \times \frac{W(W-1)}{2}$ cross-correlation terms [e.g., $\Sigma^t_{OA,OB}(n_\theta)$ in (6) per range point N, as well as the updated variance of each range point (13)]. Irrespective of the chosen model, only two Mahalanobis distances are calculated per range point. Therefore, the remaining number of calculations are constant (independent of W). In the worst case (for large W), $\mathcal{O}(NW^2t)$ calculations are required, as well as the model fitting complexity after segmentation.

Comparing this complexity with RANSAC or the Hough transform is not completely informative, since, they perform different tasks. Hough transform calculations increase exponentially when using higher order models [2]. In RANSAC, the computational complexity per feature is approximately $\mathcal{O}(NK)$ + model fitting complexity, where K is the number of trials per feature. An advantage of RANSAC is that K can be chosen as

$$K = \log P_{\text{fail}} / (\log(1 - P_{\text{good}}^W)) \tag{22}$$

where P_{fail} is the probability that the algorithm will fail without finding a successful fit, if one exists, and P_{good} is the probability of a randomly selected point fitting the model. K increases drastically with higher order models, and with a higher probability of data outliers [17]. In the proposed algorithm, the number of iterations t, necessary to achieve a certain level of smoothing can be crudely compared with LK/W^2 in RANSAC, where L is the number of sought features per scan. Contrary to RANSAC, the anisotropic diffuser complexity is independent of the number of model outliers and less sensitive to the model complexity (W).

VI. Experimental Results

As an initial demonstration of the algorithm, a simulated range scan, calculated from a simple, simulated environment containing four line segments and two circles, is used. This is to verify the extracted lines and circles, against known ground truth. The data is simulated by adding zero mean, Gaussian noise, with a constant range variance of 1.6×10^{-3} m^2 to the theoretical range values produced by a ray tracing algorithm. The simulated line intersections and the centers of the two circles (taken as ground truth) are sought as robot occlusion-invariant features. First, the line-model mask smoothing algorithm is used to smooth the range scan and fit lines, as shown schematically in Fig. 2. The results of the line based smoothing are
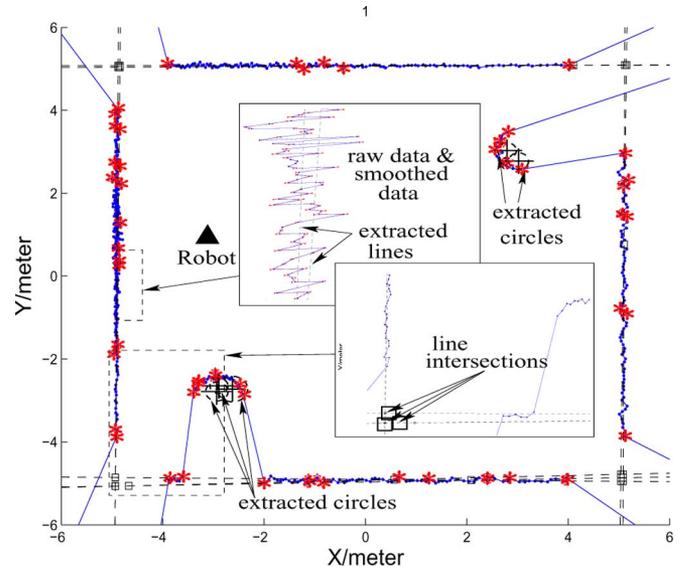
Fig. 3. Smoothing algorithm is applied once to a simulated range scan. The robot is located at the triangle (▲). Detected model outliers are marked as stars (∗). Lines and their intersections (squares (□)) are extracted, followed by circles and their centers (black crosses (+)), labeled "extracted circles."

shown in Fig. 3. For the remaining points, which are rejected by the segmentation method, explained in Section IV, the circular smoothing model is applied to extract circles.[13] When smoothing for just one iteration (Fig. 3), many edge points, shown as stars (∗) are detected (Mahalanobis distance exceeds the threshold explained in Section IV) due to the measurement noise, which can result in oversegmentation (one true line segment being subdivided into many detected ones). Similarly, too many circles are fitted to the remaining data.

After 20 anisotropic smoothing iterations (Fig. 4), only the dominant line intersection points (A, B, C, and D) are extracted. The fitted lines and extracted intersection points match the expected result from the environment (see lower inset for line intersection C). The corrected range data within the segments, is also smoothed (as can be seen in the left inset figure), and the nonlinear segments (e.g., the upper-right circular segment—see upper right inset figure) are left almost unsmoothed by the line based algorithm, but are then smoothed by the circular model, which demonstrates the usefulness of the masks. Close analysis of Fig. 3 shows that after only one iteration, some points within the circular sections are considered not to obey the circular model due to the noisy range data (in the upper-right circular section, too many small circles have been extracted within the true circular region). However, after 20 scale-space iterations (Fig. 4), further smoothing by the circular model removes these points, and the circular segment centers (+ signs) and end points (stars) are successfully determined.

For comparison purposes, Fig. 5 shows the results of running RANSAC on the same data set, again, initially searching for lines, and then using the rejected data to search for circles (see the right-hand column of Fig. 2). To make the comparison as fair as possible, once lines and circles were accepted by the RANSAC algorithm, they were recomputed based on all the accepted inlier data. In this paper,

---

[13]"When" to apply each model is an open question, and the application of the line model, before the circular model is used for demonstration only. In fact, a line can be considered to be a particular case of a circle (with infinite radius); however, its representation then becomes ill-defined. Line extraction, followed by circle extraction gave the best results, for both the proposed algorithm and RANSAC.
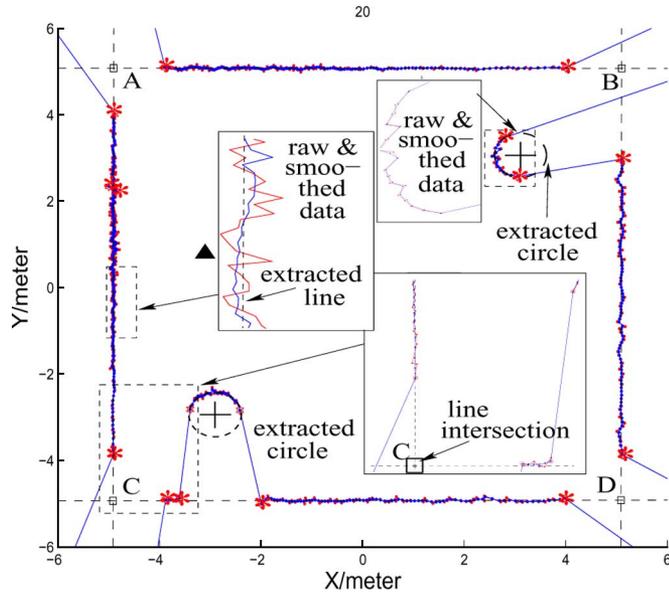
Fig. 4. Both line- and circle-based smoothers are applied to the same data set as used in Fig. 3. The number of smoothing iterations is 20.
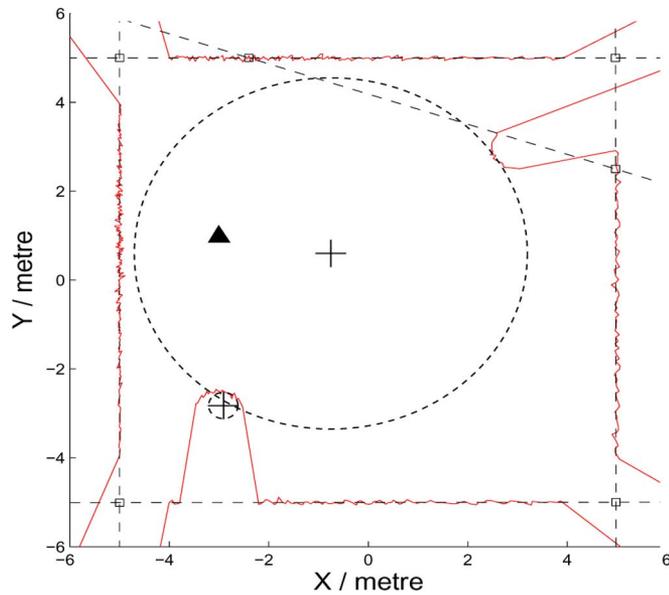
Fig. 5. RANSAC algorithm is used with 180 trials ($P_{\text{fail}} = 0.16$). Note that, with more iterations, more line segments are found in general, which are highly likely to contain the true line segments.

the RANSAC parameters used were: number of inliers necessary for lines = 40 (for circles = 10) $P_{\text{good}} = 0.804/4^{14}$, inlier threshold distance = 0.1 m, and $P_{\text{fail}} = 0.16$. Note that this value of P_{fail} causes the RANSAC algorithm to run 45 times per feature [from (22)], which is computationally approximately equivalent to 20 scale-space iterations of the proposed anisotropic diffuser since $t \approx LK/W^2$, where L is the number of features sought. (To favor the number RANSAC iterations,

[14] Since this is a simulation, it is known that 80.4% of the data points are spread approximately equally among the four lines. For circles, $P_{\text{good}} = 0.114/2$. Note that calculating these values should be favorable to the performance of RANSAC, since in reality, they would not be known.
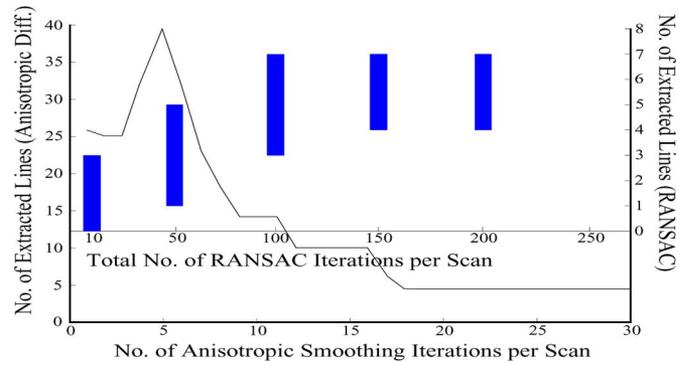
Fig. 6. Number of extracted lines versus the number of smoothing iterations.
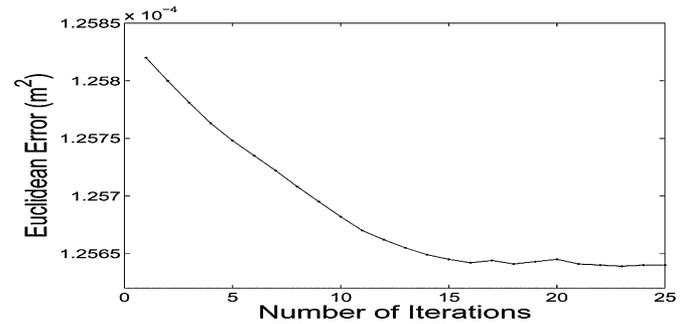
Fig. 7. Euclidean mean-squared error between all four extracted intersection point locations and their respective ground truths versus the number of smoothing iterations beyond those necessary to determine the correct number of lines.

we assumed the worst case value for the anisotropic diffuser of $W = 3$ (circles), but used $L = 4$ (lines)). Under these conditions, RANSAC usually correctly finds all of the line segments, but often detects other, spurious, incorrect lines as shown in Fig. 5. The extraction of circles can, however, pose a problem. As can be seen, the smaller circle is approximately correct, but the larger circle is incorrect, due to the combination of the global nature of the RANSAC algorithm and the noisy data.

In Fig. 6, the number of fitted lines is plotted with respect to the number of anisotropic diffusion smoothing iterations. After 18 iterations, the extracted line number reduces to the correct number, which is four in this environment. With further iterations, the algorithm begins to smooth range data in each segment to improve the accuracy of the extracted intersection points. This is demonstrated in Fig. 7, where the mean-squared Euclidean error between all four extracted intersection point locations and their respective ground truths is plotted.

In the case of RANSAC, the results are quite different. The vertical bars in Fig. 6, correspond to the variation in the number of lines extracted (indicated on the right vertical axis), as a function of the total number of RANSAC iterations. Note that the two sets of data have been aligned in terms of approximately equal computational complexity [e.g. ten anisotropic smoothing iterations $\approx$90 total RANSAC iterations for the four features]. Since RANSAC is a randomized estimator, it can give different results, every time it runs on the same data set. RANSAC was used 20 times in each case and, for example, when P_{fail} was set to give a total of 50 trials for all four line features, between one and as many as five lines could be extracted. However, when P_{fail} was set to give at least a total of 150 trials, RANSAC always produced at least four lines (which were usually the correct ones), but often also found other, incorrect lines—sometimes three more. Another fundamental
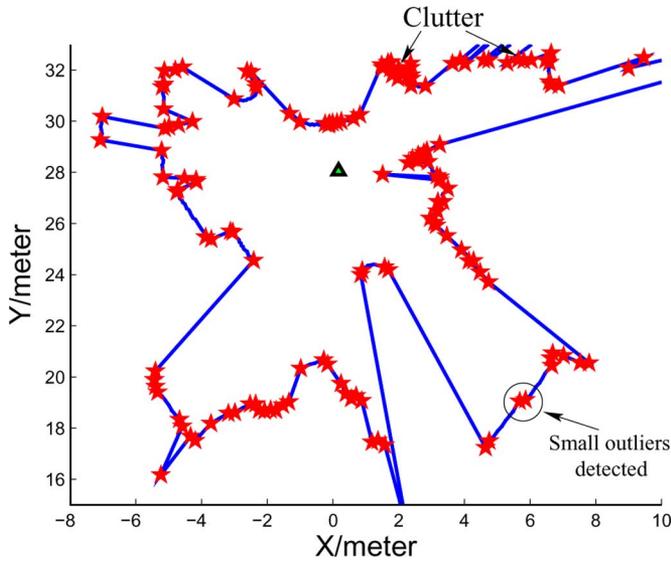
Fig. 8.   Indoor laser scan, segmented after smoothing once by the line mask, without model noise (Q_L set to 0). The lines show the smoothed scan. The stars (⋆) indicate the points, where the calculated Mahalanobis distances are larger than the threshold. These will be used to segment the whole scan.

Fig. 9.   Scan from Fig. 8, segmented after being smoothed once by the line model mask with a high model noise variance of $Q_L = 0.01$ m^2. The dominant edge points A, B, C, and D are all undetected.

difference between the two algorithms, is that with higher iteration number (scale), the anisotropic diffuser smoothes the data, resulting in a lower spatial error between the finally extracted lines and the true ones. In the case of RANSAC, however, as long as the correct lines are extracted, no improvement in their localization takes place. The best Euclidean mean-squared error we could achieve with RANSAC (determined when RANSAC extracted only the four correct lines) was 0.6×10^{-3} m^2, approximately four times higher than that attainable with the proposed smoother.

In Fig. 8, a laser scan was smoothed once by the line model, with model noise variance, Q_L in (6), set to zero. This means that points are to be associated with "perfect" line segments. After smoothing, the laser scan was segmented based on the Mahalanobis distance for each point (Section IV). The stars indicate the detected segment edges.

In Fig. 9, a very high line model noise variance of $Q_L = 0.01$ m^2 was chosen. The obvious difference with Fig. 8 is that there are less edge points detected. The whole laser scan was roughly segmented but many dominant edge points (such as points A, B, C, and D in Fig. 9) have been missed. Due to the high process model noise, the filter is more flexible to stand for small changes in range data that disobey the line model. However, the missed dominant edge points will seriously affect the segmentation of the whole laser scan.

In Fig. 10, the laser scan underwent 20 smoothing iterations with the line model mask, with the same model noise variance (equal to zero) as Fig. 8. The scan was then segmented based on the Mahalanobis distances. Extended line segments and their intersections (shown as squares) have been extracted. The same effect results, as in Fig. 9, in that the number of detected edge points was reduced. However, the algorithm is sensitive enough to detect all of the dominant edge points (in particular A, B, C, and D in Fig. 10), yet still smooth over the small outliers shown in Fig. 8, as it produces only the single-line EF from this segment (Fig. 10). It is clear that smoothing and segmentation at higher scale, is improved. It is interesting to note that, in these experiments, a higher number of iterations indicates that even with strict models ($Q_L = 0$), all of the dominant features are found, and the noise terms in (4) and (5) can be neglected, removing the necessity of parameters Q_L and Q_C.
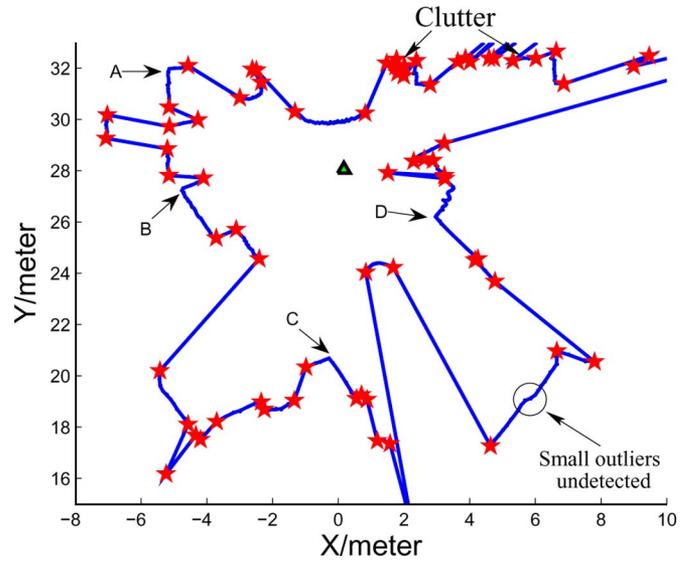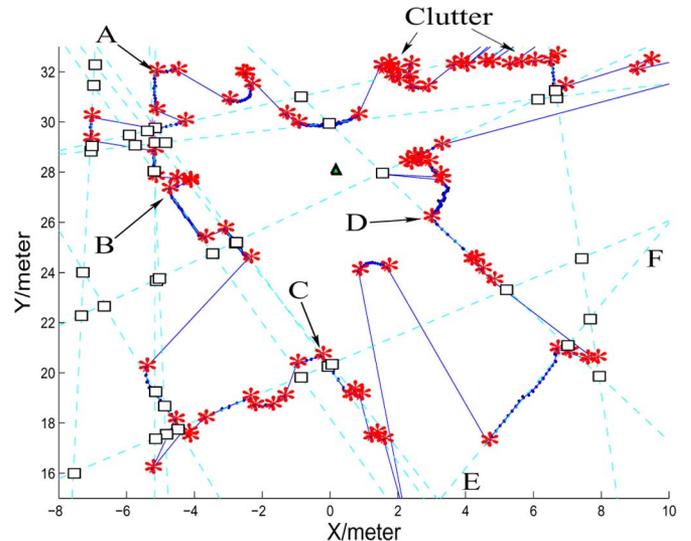
Fig. 10.   Scan from Fig. 8, segmented after smoothing 20 times by the line model mask with $Q_L = 0$. Dominant edge points A, B, C, and D are now all detected, yet the small outliers (Figs. 8 and 9) do not affect the detected line EF. The extracted lines or their intersections can be used as features.

In Fig. 11, two consecutive $0.5°$ angular resolution, outdoor laser range scans have been transformed into global coordinates according to vehicular odometric information. The line model smoothing algorithm has been applied 20 times to each scan and lines were extracted. The intersection points of these lines, denoted by squares (□) in the first scan, and inverted triangles (▼) in the second, can be used as robot-pose-invariant, point features. From a qualitative point of view, even though there is an odometric error between the scans, resulting in the imprecise localization of the sensor, one would expect most of the extracted features from two consecutive scans to be matched by a suitable data association algorithm.
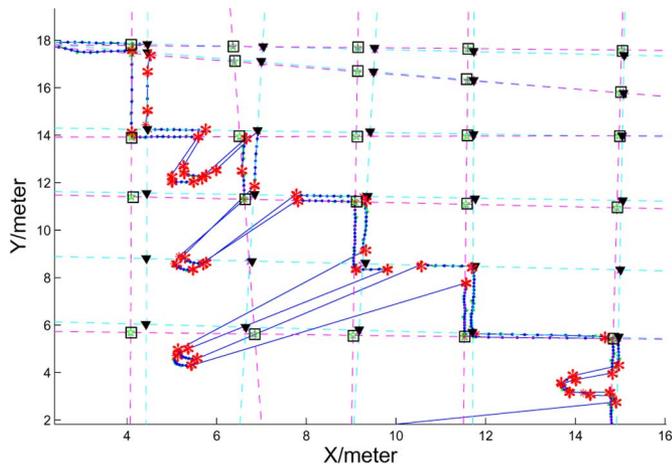
Fig. 11. Two successive outdoor range scans, smoothed and segmented by the line mask. The line intersection points (□) in the first scan, and inverted triangles (▼) in the second, can be used as occlusion-invariant, point features.

VII. CONCLUSION

Unlike RANSAC, the presented algorithm smoothes range data, to produce a local description of features, which, in some circumstances, can be more beneficial than global descriptions for robot navigation. Also, its computational complexity is independent of the number of model outliers, and is less affected by the use of higher order geometric models.

The smoothing and segmentation of range data is fundamentally different from that of image data. Structure preserving, and noise reduction algorithms in vision, use the local intensity gradient as a measure of noise. Range values are completely environment dependent, and not constant between features. Therefore, in this algorithm, the Mahalanobis distance between observed range values and their geometric-model-based predictions is used as the "measure of noise." A mask weighting function of the Mahalanobis distance was derived, which behaves as the diffusion coefficient in the anisotropic diffusion equation, often applied in vision, which guarantees that no new features are introduced with increase of scale. This mask can be applied iteratively, providing smoothing at different scales. The results demonstrated that the number of extracted features (lines or circles) converged to the true number with increase of scale, and the error between the extracted and true feature coordinates converged to a minimum. It has been shown that with increase of scale, the algorithm automatically reduces noise, only within the model-compliant regions of the range scans, yielding superior, postsmoothing, segmentation possibilities.

REFERENCES

[1] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 12, no. 7, pp. 629–639, Jul. 1990.

[2] P. V. C. Hough, "Machine analysis of bubble chamber pictures," presented at the Int. Conf. High Energy Accelerators Instrum., Geneva, Switzerland, 1959.

[3] J. Forsberg, U. Larsson, and A. Wernersson, "Mobile robot navigation using the range-weighted Hough transform," *IEEE Robot. Autom. Mag.*, vol. 2, no. 1, pp. 18–26, Mar. 1995.

[4] R. H. T. Chan and P. K. S. Tam, "A new Hough transform based position estimation algorithm," *Intell. Inf. Syst.*, vol. 29, pp. 140–144, 1994.

[5] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Commun. Assoc. Comput. Mach.*, vol. 24, no. 6, pp. 381–395, Jun. 1981.

[6] R. Martinez-Cantin, J. A. Castellanos, J. D. Tardos, and J. M. M. Montiel, "Adaptive scale robust segmentation for 2D laser scanner," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2006, pp. 796–801.

[7] V. Nguyen, A. Martinelli, N. Tomatis, and R. Siegwart, "A comparison of line extraction algorithms using 2d laser rangefinders for indoor mobile robots," in *IEEE Int. Conf. Intell. Robots Syst.*, 2005, pp. 1929–1934.

[8] S. L. Horowitz and T. Pavlidis, "Picture segmentation by a tree traversal algorithm," *J. ACM*, vol. 23, no. 2, pp. 368–388, 1976.

[9] M. Adams. *Sensor Modelling, Design and Data Processing for Autonomous Navigation*. Singapore: World Scientific, 1999.

[10] S. Zhang, L. Xie, M. Adams, and F. Tang, "Geometrical feature extraction using 2d range scanners," presented at the Int. Conf. Cont. Autom., Montreal, Canada, Jun. 2003.

[11] S. I. Roumeliotis and G. A. Bekey, "Segments: A layered, dual-Kalman filter algorithm for indoor feature extraction," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2000, pp. 454–461.

[12] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Proc. Sixth Int. Conf. Comput. Vis.*, 1998, p. 839.

[13] D. Comaniciu and P. Meer-Mean, "Shift analysis and applications," in *Proc. IEEE Int. Conf. Comput. Vis.*, 1999, pp. 1197–1203.

[14] M. J. Black, G. Sapiro, D. H. Marimont, and D. Heeger, "Robust anisotropic diffusion," *IEEE Trans. Image Process.*, vol. 7, no. 3, pp. 421–432, Mar. 1998.

[15] F. Tang, M. Adams, J. Ibanez-Guzman, and W. S. Wijesoma, "Pose invariant, robust feature extraction from range data with a modified scale space approach," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2004, pp. 3173–3179.

[16] Y. L. You, W. Xu, A. Tannenbaum, and M. Kaveh, "Behavioral analysis of anisotropic diffusion in image processing," *IEEE Trans. Image Process.*, vol. 5, no. 11, pp. 1539–1553, Nov. 1996.

[17] O. Chum and J. Matas, "Randomized RANSAC with Td,d Test," presented at the Br. Mach. Vis. Conf., Cardiff, U.K., Sep. 2002.

A Globally Stable PD Controller for Bilateral Teleoperators

Emmanuel Nuño, Romeo Ortega, Nikita Barabanov,
and Luis Basañez

Abstract—In a recent scheme, with delayed derivative action [Lee and Spong, *IEEE Trans. Robot.*, vol. 22, no. 2, pp. 269–281, Apr. 2006], it is claimed that a simple proportional derivative (PD) scheme yields a stable operation. Unfortunately, the stability proof hinges upon unverifiable assumptions on the human and contact environment operators, namely, that they define $\mathcal{L}_\infty$–stable maps *from velocity to force*. In this short paper, we prove that it is indeed possible to achieve stable behavior with simple PD-like schemes—even without the delayed derivative action—under the classical assumption of passivity of the terminal operators.

Index Terms—Bilateral teleoperation, communication delays, passivity, proportional derivative (PD) control.