

---

# Performance Evaluation of Ultrasonic Arc Map Processing Techniques by Active Snake Contours

Kerem Altun and Billur Barshan

Department of Electrical and Electronics Engineering, Bilkent University, Bilkent,  
TR-06800 Ankara, Turkey  
{kaltun, billur}@ee.bilkent.edu.tr

**Summary.** Active snake contours are considered for representing the maps of an environment obtained by different ultrasonic arc map (UAM) processing techniques efficiently. The mapping results are compared with the actual map of the room obtained with a very accurate laser system. This technique is a convenient way to represent and compare the map points obtained with different techniques among themselves, as well as with an absolute reference. It is also applicable to map points obtained with other mapping techniques.

## 1 Introduction

Ultrasonic sensors have been widely used in robotic applications due to their accurate range measurements, robustness, low cost, and simple hardware interface. When coupled with intelligent processing, they provide a useful alternative to more complex laser and camera systems, especially when it is not possible to use these systems in some environments due to surface characteristics or insufficient ambient light. Despite their advantages, the frequency range at which air-borne ultrasonic transducers operate is associated with a large beamwidth that results in low angular resolution and uncertainty in the location of the echo-producing object. Thus, having an intrinsic uncertainty of the actual angular direction of the range measurement and being prone to various phenomena such as multiple and higher-order reflections and cross-talk between transducers, a considerable amount of modeling, processing, and interpretation of ultrasonic data is necessary.

Most commonly, the large beamwidth of the transducer is accepted as a device limitation that determines the angular resolving power of the system, and the reflection point is assumed to be along the line-of-sight (LOS) of the transducer. According to this naive approach, a simple mark is placed along the LOS at the measured range, resulting in inaccurate maps with large angular errors and artifacts. In earlier work, basically, there have been two approaches to the representation of ultrasonic data: feature-based and grid-based. Grid-based approaches do not attempt to make difficult geometric decisions early in the interpretation process unlike feature-based approaches that extract the geometry of the sensor data as the first step. As a first attempt, several researchers have fitted line segments to ultrasonic data as features that crudely approximate the room geometry [10, 18, 13]. This approach proved to be difficult and brittle because straight

lines fitted to time-of-flight (TOF) data do not necessarily match or align with the world model, and may yield many erroneous line segments. Improving the algorithms for detecting line segments and including heuristics does not really solve the problem. A more physically meaningful representation is the use of *regions of constant depth* (RCDs) as features. RCDs are circular arcs which are natural features of the raw ultrasonic TOF data from specularly reflecting surfaces, first reported in [16], and further elaborated on in [9]. Alternatively, the angular uncertainty in the range measurements has been represented by ultrasonic arc maps (UAMs) [6] that preserve more information (see Fig. 1(c) for a sample UAM). This is done by drawing arcs spanning the beamwidth of the sensor at the measured range, representing the angular uncertainty of the object location and indicating that the echo-producing object can lie anywhere on the arc. Thus, when the same transducer transmits and receives, all that is known is that the reflection point lies on a circular arc of radius  $r$ . More generally, when one transducer transmits and another receives, it is known that the reflection point lies on the arc of an ellipse whose focal points are the transmitting and receiving elements. The arcs are tangent to the reflecting surface at the actual point(s) of reflection.

Several techniques have been proposed to process these UAMs (Table 1), that result in more accurate maps of the environment. These techniques are summarized in Section 2. Each processed UAM results in a collection of (usually a large number of) data points, represented as a black-on-white image. In [4], the DM technique is newly proposed, and a comparison of these techniques is provided based on three different error criteria. In this paper, we propose a method to compactly and efficiently represent the resulting map points that will also make it convenient to assess the accuracy of the different UAM processing techniques. Basically, active snake curves will be fitted to the results of processing the UAM by each technique and a comparison with a very accurate laser map (considered as absolute reference) will be provided.

**Table 1.** Different UAM processing techniques

1	Point marking (PM) [16]	5	Bayesian update (BU) [1]
2	Voting and thresholding (VT) [3]	6	Triangle based fusion (TBF) [15]
3	Directional maximum (DM) [4]	7	Arc transversal median (ATM-org) [8]
4	Morphological processing (MP) [6]	8	Modified ATM (ATM-mod) [4]

## 2 UAM Processing Techniques

This section summarizes various techniques for processing the UAM constructed from raw ultrasonic TOF measurements. Detailed descriptions of the methods can be found in [4], or respective references indicated in the subsections or in Table 1.

### 2.1 Point Marking (PM)

This is the simplest approach, mentioned above, where a mark is placed along the LOS at the measured range [16]. This method produces reasonable estimates for the locations of objects if the arc of the cone is small. This can be the case at higher frequencies

of operation where the corresponding sensor beamwidth is small or at nearby ranges. Since every arc is reduced to a single point, this technique cannot eliminate any of the outlying TOF readings. The resulting map is usually inaccurate with large angular errors and artifacts.

## 2.2 Voting and Thresholding (VT)

In this technique, each pixel stores the number of arcs crossing that pixel, resulting in a 2-D array of occupancy counts for the pixels [3]. By simply thresholding this array and zeroing the pixels lower than the threshold, artifacts can be eliminated and the map is extracted.

## 2.3 Directional Maximum (DM)

This technique is based on the idea that in processing the acquired range data, there is a direction-of-interest (DOI) associated with each detected echo. Ideally, the DOI corresponds to the direction of a perpendicular line drawn from the sensor to the nearest surface from which an echo is detected. However, in practice, due to the angular uncertainty of the object position, the DOI can be approximated as the LOS of the sensor when an echo is detected. Since prior information on the environment is usually unavailable, the DOI needs to be updated while sensory data are being collected and processed on-line [4].

In the implementation, the number of arcs crossing each pixel of the UAM is counted and stored, and a suitable threshold value is chosen, exactly the same way as in the VT method. The novelty of the DM method is the processing done along the DOI. Once the DOI for a measurement is determined using a suitable procedure, the UAM is processed along this DOI as follows: The array of pixels along the DOI is inspected and the pixel(s) exceeding the threshold with the maximum count is kept, while the remaining pixels along the DOI are zeroed out. If there exist more than one maxima, the algorithm takes their median (If the number of maxima is odd, the maxima in the middle is taken; if the number is even, one of the two middle maxima is randomly selected.) This way, most of the artifacts of the UAM can be removed.

## 2.4 Morphological Processing (MP)

The processing of UAMs using morphological operators was first proposed in [6]. This approach exploits neighboring relationships and provides an easy to implement yet effective solution to ultrasonic map building. By applying binary morphological operators, one can eliminate the artifacts of the UAM and extract the surface profile.

## 2.5 Bayesian Update Scheme for Occupancy Grids (BU)

Occupancy grids were first introduced by Elfes, and a Bayesian scheme for updating their probabilities of occupancy and emptiness was proposed in [1] and verified by ultrasonic data. Starting with a blank or completely uncertain occupancy grid, each range measurement updates the probabilities of emptiness and occupancy in a Bayesian manner. The reader is referred to [1] for a detailed description of the method and [4] for its implementation in this work.

## 2.6 Triangulation-Based Fusion (TBF)

The TBF method is primarily developed for accurately detecting the edge-like features in the environment based on triangulation [15]. The triangulation equations involved are not suitable for accurately localizing planar walls.

Unlike the previously introduced grid-based techniques, the TBF method extracts the features of the environment by using a geometric model suitable for edge-like features. In addition, TBF considers a sliding window of ultrasonic scans where the number of rows of the sliding window corresponds to the number of ultrasonic sensors fired, and the number of columns corresponds to the number of most recent ultrasonic scans to be processed by the algorithm. TBF is focused on detection of edge-like features located at  $\leq 5$  m. The other methods consider all of the arcs in the UAM corresponding to all ranges, and are suitable for detecting all types of features.

## 2.7 Arc-Transversal Median (ATM)

The ATM algorithm requires both extensive bookkeeping and considerable amount of processing [8]. For each arc in the UAM, the positions of the intersection(s) with other arcs, if they exist, are recorded. For arcs without any intersections, the mid-point of the arc is taken to represent the actual point of reflection (as in PM). If the arc has a single intersection, the algorithm uses the intersection point as the location of the reflecting object. For arcs with more intersections, the median of the positions of the intersection points with other arcs is chosen to represent the actual point of reflection. In [8], the median operation is applied when an arc has *three or more* intersection points. If there is an even number of intersections, the algorithm uses the mean of the two middle values (except that arcs with two intersections are ignored). It can be considered as a much improved version of the PM approach.

We have also implemented a modified version of the algorithm (ATM-mod) where we ignored arcs with no intersections. Furthermore, since we could not see any reason why arcs with two intersections should not be considered, we took the mean of the two intersection points.

## 3 Fitting Active Snake Contours to UAMs

A snake, or an active contour [14] can be described as a continuous deformable closed curve. It is commonly used in image processing for edge detection or image segmentation [14, 17]. The deformation is controlled by external and internal forces. External forces depend on the image and they try to stretch or shrink the curve to fit to the data, whereas internal forces impose elasticity and rigidity constraints on the curve. We define a snake as a parametrized closed curve  $\mathbf{v}(s) = (x(s), y(s))$ ,  $s \in [0, 1]$ , whose energy is given by the functional

$$E_{snake} = \int_0^1 (E_{int}(\mathbf{v}(s)) + E_{ext}(\mathbf{v}(s))) ds \quad (1)$$

where  $x(s)$  and  $y(s)$  are periodic functions representing the  $x$  and  $y$  coordinates of the snake and  $s$  is the normalized arc length parameter of the snake curve. The internal energy is given by

$$E_{int}(\mathbf{v}(s)) = \frac{1}{2} \left( \alpha \left\| \frac{d(\mathbf{v}(s))}{ds} \right\|^2 + \beta \left\| \frac{d^2(\mathbf{v}(s))}{ds^2} \right\|^2 \right) \quad (2)$$

where  $\alpha$  is the elasticity parameter and  $\beta$  is the rigidity parameter, taken as constants. The external energy will be denoted by  $E_{ext}(\mathbf{v}(s)) = P(\mathbf{v}(s))$ , where  $P$  is a potential function that depends on the image data.

In general, the selection of the potential function varies depending on the application. However, it must be minimum on the image edges if the snake is used for edge detection. Kass *et al.* suggest using the negative of the image gradient magnitude as a potential function [14]. However this is only feasible if the snake is initialized close to the image boundaries. Filtering the image with a Gaussian low-pass filter is also suggested in the same paper to increase the capture range of the snake, but this causes the edges to become blurry, thus reducing the accuracy. If the image is a black-on-white one (as is the case in this study), then the image intensity can be used as the potential function, either as itself or convolved with a Gaussian blur [11]. Obviously this method also suffers from the same drawbacks stated above. Another solution proposed in [12] is using a distance map to increase the capture range of the contour, which is the approach used in this study.

Approaches that do not use a potential function as the external energy term also exist in literature [5]. This relaxes the constraint that the external forces pulling the snake towards the edges should be conservative, i.e., derived from a potential field. Xu *et al.* define a non-conservative force field representing the external forces and use force balance equations rather than energy-based approach to solve the problem [5]. However, this idea is not used in our work.

Having chosen a potential function, the goal is to find the curve that minimizes the energy functional in Eqn. (1). This problem can be solved by using calculus of variations. The minimizing curve must satisfy the following Euler-Lagrange equation [14]:

$$\alpha \frac{d^2(\mathbf{v}(s))}{ds^2} - \beta \frac{d^4(\mathbf{v}(s))}{ds^4} - \nabla P(\mathbf{v}(s)) = 0 \quad (3)$$

For some cases it may be possible to solve this equation analytically, but a general analytical solution does not exist. The general practice is to initialize an arbitrary time-dependent snake curve  $\mathbf{v}(s, t)$ . Eqn. (3) is then set equal to the time derivative of the snake, where a solution will be found when the time derivative vanishes. That is,

$$\alpha \frac{\partial^2 \mathbf{v}}{\partial s^2} - \beta \frac{\partial^4 \mathbf{v}}{\partial s^4} - \nabla P(\mathbf{v}) = \frac{\partial \mathbf{v}}{\partial t} \quad (4)$$

These equations are then discretized for a numerical solution. Furthermore, the snake is treated as a collection of discrete points joined by straight lines, and a snake curve is

initialized on the image. Approximating the derivatives by finite differences, the evolution equations of the snake reduce to [17]:

$$x_{t+1} = (A + \gamma I)^{-1} \left( \gamma x_t - \kappa \frac{\partial P}{\partial x} \Big|_{(x_t, y_t)} \right) \quad (5)$$

$$y_{t+1} = (A + \gamma I)^{-1} \left( \gamma y_t - \kappa \frac{\partial P}{\partial y} \Big|_{(x_t, y_t)} \right) \quad (6)$$

for all points  $(x, y)$  on the snake. Here  $t$  is the current time (or iteration) step,  $\gamma$  is the Euler step size,  $\kappa$  is the external force weight,  $I$  is the identity matrix of appropriate size and  $A$  is a pentadiagonal banded matrix that depends on  $\alpha$  and  $\beta$ . The sizes of the matrices  $A$  and  $I$  are determined by the number of points on the snake, which may change as the algorithm is executed. The variables  $x_t$  and  $y_t$  represent the coordinates of the discrete points on the snake at time  $t$ .

## 4 Experiments

The different techniques, listed in Table 1, are considered for processing the UAMs. Each of these techniques results in a different set of points to which a snake curve is fitted.

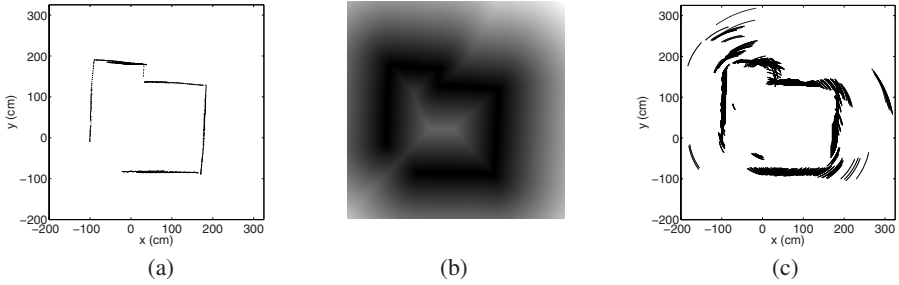
The potential function used in this study is based on the Euclidean distance transform, as suggested in [12]. As stated before, processed UAMs in our case are represented as black pixels (i.e.,  $I(x, y) = 0$ ) on white background (i.e.,  $I(x, y) = 1$ ), where  $I(x, y)$  is the intensity of the image. Euclidean distance transform is defined for all points on the image as the Euclidean distance to the nearest black pixel. That is, the potential function is selected as

$$P(x, y) = \min_{\{(x', y') | I(x', y') = 0\}} \sqrt{(x - x')^2 + (y - y')^2} \quad (7)$$

for all points  $(x, y)$  on the image. Note that the value of the potential function is zero for the points on the extracted map and increases gradually when  $(x, y)$  moves away from the map points. The Euclidean distance transform is computationally costly, and a number of algorithms and other distance transforms have been proposed in the literature to approximate it [7, 2]. However, in this study the Euclidean distance transform is implemented in its original form.

An example image of a room acquired with a laser system is shown in Fig. 1(a). This is the original laser data which is quite accurate, and is used as the absolute reference to compare the methods given in Table 1. The corresponding distance map is shown in Fig. 1(b), which is drawn by scaling the values of the potential function to be between 0 and 255. Fig. 1(c) shows the raw UAM for the room.

The values for the parameters in Eqns. (5) and (6) are selected as  $\alpha = \gamma = 1$ ,  $\beta = 0.1$  and  $\kappa = 2.5$ . Selecting  $\beta = 0.1$  enforces the second derivative in the energy term to have less weight, thus allowing sharp corners in the snake. The snake curves fit to the laser and the processed UAMs are given in Fig. 2. The blue curves are the snakes fitted



**Fig. 1.** (a) Laser map of the room, (b) its distance map, (c) and the UAM

to these UAM data. The red curves are the snakes fitted to the laser data, and it is superimposed on the processed UAMs for better visualization. There was an opening on the lower-left corner of the room from which no ultrasonic data were collected. Therefore, the part of the snake curve in that region is not drawn in the figure and not included in the error calculations.

The snake is initialized as a circle whose center is at (30, 55) having a radius of 185 units so that it encompasses the room boundary. We allow the snake to converge to outlier points caused by crosstalk and/or multiple reflections to provide a better evaluation of the methods. Then, the snake is evolved for a fixed amount of iterations (currently 100), determined experimentally to ensure that each UAM snake converges to the map. After each iteration, the points on the snake are checked for uniformity. The distance between any two neighboring points is maintained between 2–4 units, also determined experimentally. That is, the points are destroyed or created as required by this constraint, after each iteration.

The snake fit to the laser data is referred to as  $C_{laser}$  from now on in this text. The snakes fitted to the processed UAM data will be referred to as  $C_i$ , where  $i$  denotes the index of the method given in Table 1. Thus the  $i^{\text{th}}$  snake is represented as a collection of points  $(x_{ij}, y_{ij}), j = 1, \dots, N_i$  where  $N_i$  is the total number of points on snake  $C_i$ .

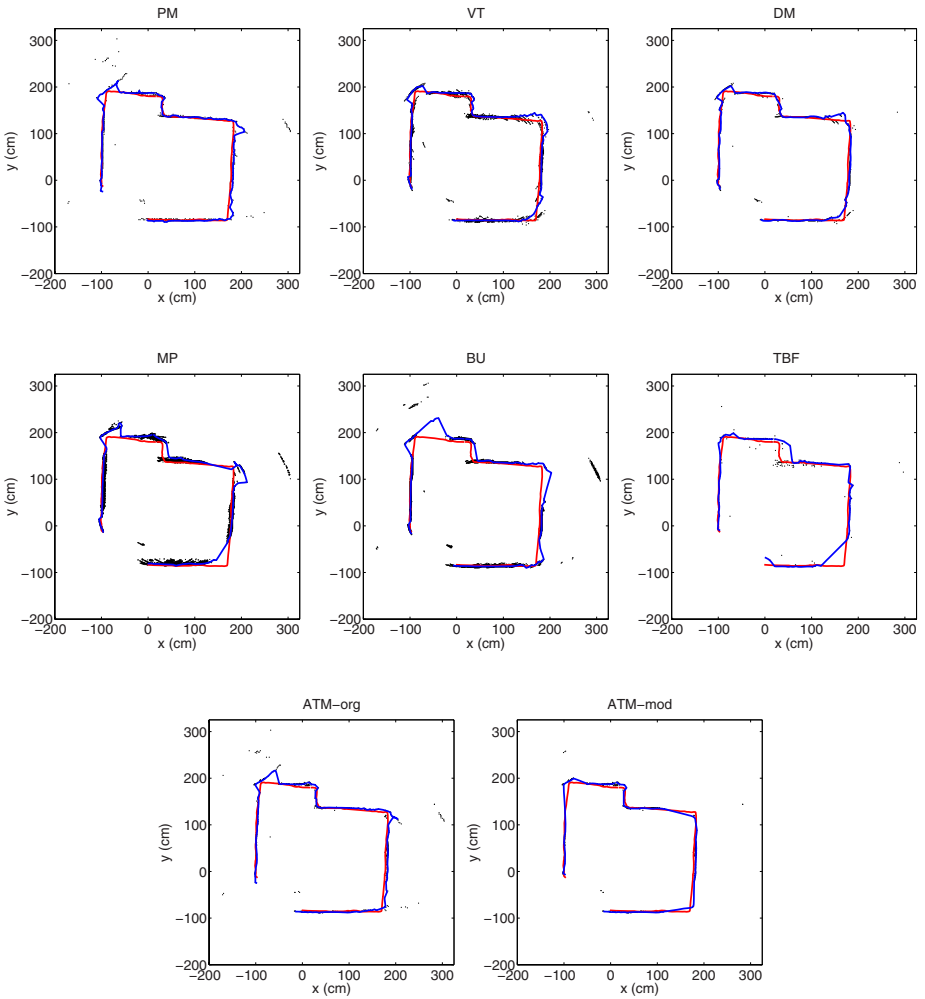
An error measure is defined to determine the closeness between the laser snake  $C_{laser}$  and processed UAM snake  $C_i$ . It is calculated by finding the distance of the nearest point on snake  $C_{laser}$  for all points on snake  $C_i$  and averaging these distances. First, a distance function is defined for points on a given snake  $C_i$  as:

$$d_{i/laser}(x_{ij}, y_{ij}) = \min_{k=1, \dots, N_{laser}} \sqrt{(x_{ij} - x_k)^2 + (y_{ij} - y_k)^2}, \quad j = 1, \dots, N_i \quad (8)$$

where  $k$  is an index for points on snake  $C_{laser}$  and  $N_{laser}$  is the total number of points on the snake  $C_{laser}$ . Then, the error is given as:

$$e_i = \frac{1}{N_i} \sum_{j=1}^{N_i} d_{i/laser}(x_{ij}, y_{ij}) \quad (9)$$

The errors for the different methods are tabulated in Table 2. According to the results, ATM-mod and DM techniques have the smallest errors, and MP and BU perform the worst. The remaining techniques are comparable to the PM method.



**Fig. 2.** Results of snake curve fitting for all the UAM processing techniques. Red curves correspond to the snake fitted to laser data and the blue curves are the snakes fitted to the map points.

**Table 2.** Error values of the different techniques

Method	PM	VT	DM	MP	BU	TBF	ATM-org	ATM-mod
Error	4.321	4.218	3.478	6.806	6.871	4.730	4.427	3.403

DM and ATM-mod methods eliminate most of the artifacts resulting from crosstalk and multiple and higher-order reflections (Fig. 2) so that the corresponding snake curves follow the laser data very closely. MP and BU methods cannot eliminate those artifacts as much, resulting in larger errors. Same applies to the PM method; it can not eliminate the outlier points and could have resulted in large error values. We should note here that



the snake was initialized outside the boundaries of the room. Initializing the snake inside the room would also be possible. In this case, the spurious points outside the boundaries would not affect the snake curve as much, allowing it to follow the boundaries of the room more closely. However, this would not result in a fair comparison between the techniques. In a practical application when comparison and evaluation of the techniques is not an issue, this might be a good choice to eliminate the erroneous points outside the boundary.

## 5 Conclusion

We have presented a technique to compactly and efficiently represent the maps obtained by processing the UAMs using different techniques. The representation of the map points with snake curves makes it easy and convenient to compare maps obtained with different techniques among themselves, as well as with an absolute reference. This approach can be applied to other mapping techniques. Our current work involves using Kohonen's self-organizing maps for the same purpose that takes into account all the outlier points.

## Acknowledgment

Kerem Altun is supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) with a doctoral scholarship. This work is supported by TÜBİTAK under grant number EEEAG-105E065.

## References

1. Elfes, A.: Sonar based real-world mapping and navigation. *IEEE Transactions on Robotics and Automation* RA-3(3), 249–265 (1987)
2. Rosenfeld, A., Pfaltz, J.L.: Distance functions on digital pictures. *Pattern Recognition* 1(1), 33–61 (1968)
3. Barshan, B.: Ultrasonic surface profile determination by spatial voting. *Electronics Letters* 35(25), 2232–2234 (1999)
4. Barshan, B.: Directional processing of ultrasonic arc maps and its comparison with existing techniques. *International Journal of Robotics Research* 26(8), 797–820 (2007)
5. Xu, C., Prince, J.L.: Snakes, shapes and gradient vector flow. *IEEE Transactions on Image Processing* 7(3), 359–369 (1998)
6. Başkent, D., Barshan, B.: Surface profile determination from multiple sonar data using morphological processing. *International Journal of Robotics Research* 18(8), 788–808 (1999)
7. Borgefors, G.: Distance transformations in digital images. *CVGIP* 34(3), 344–371 (1986)
8. Choset, H., Nagatani, K., Lazar, N.: The arc-transversal median algorithm: a geometric approach to increasing ultrasonic sensor azimuth accuracy. *IEEE Transactions on Robotics and Automation* 19(3), 513–522 (2003)
9. Leonard, J.J., Durrant-Whyte, H.F.: *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic Publishers, Boston (1992)
10. Crowley, J.L.: Navigation for an intelligent mobile robot. *IEEE Transactions on Robotics and Automation* RA-1(1), 31–41 (1985)

11. Cohen, L.D.: On active contour models and balloons. *CVGIP: Image Understanding* 53(2), 211–218 (1991)
12. Cohen, L.D., Cohen, I.: Finite element methods for active contour models and balloons for 2-D and 3-D images. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15(11), 1131–1147 (1993)
13. Drumheller, M.: Mobile robot localization using sonar. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-9*(2), 325–332 (1987)
14. Kass, M., Witkin, A., Tersopoulos, D.: Snakes: Active contour models. *International Journal of Computer Vision* 1(4), 321–331 (1987)
15. Wijk, O., Christensen, H.I.: Triangulation-based fusion of sonar data with application in robot pose tracking. *IEEE Transactions on Robotics and Automation* 16(6), 740–752 (2000)
16. Kuc, R., Siegel, M.W.: Physically-based simulation model for acoustic sensor robot navigation. *IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-9*(6), 766–778 (1987)
17. Menet, S., Saint-Marc, P., Medioni, G.: Active contour models: Overview, implementation and applications. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, November 1990, pp. 194–199 (1990)
18. Gex, W., Campbell, N.: Local free space mapping and path guidance. In: *Proceedings of IEEE International Conference on Robotics and Automation*, March 1987, pp. 424–431 (1987)