Using the Condensation Algorithm to Implement Tracking for Mobile Robots

Esther B. Meier and Frank Ade
Communication Technology Lab, Image Science
Swiss Federal Institute of Technology (ETH)
CH-8092 Zurich, Switzerland
{ebmeier,ade}@vision.ee.ethz.ch

Abstract

The detection of objects in every frame of a sequence is often not sufficient for scene interpretation. Tracking can increase the robustness, especially when occlusions occur or when objects temporally disappear. The standard approach for tracking is to use a Kalman filter for every object. This, however, requires the use of a high complexity management system to deal with the multiple hypotheses necessary to track all anticipated objects.

In this paper we present a stochastic approach which is based on the CONDENSATION *algorithm – conditional density propagation over time – that is capable of tracking multiple objects with multiple hypotheses in range images. A probability density function describing the likely state of the objects is propagated over time using a dynamic model. The measurements influence the probability function and allow the incorporation of new objects into the tracking scheme. Additionally, the representation of the density function with a fixed number of samples ensures a constant running time per iteration step. Results on different data sources are shown for mobile robot applications.*

1. Introduction

This paper is concerned with object hypotheses tracking using a stochastic approach based on the CONDENSATION algorithm. We apply this new technique to mobile robot applications using range image sequences. Handling occlusion or temporally disappeared objects in dynamic scenes are basic conditions to our framework. Therefore, the segmentation results which contain the information at only one time step are not sufficient to establish a safe control system. To increase the robustness, a tracking process which can handle multiple hypotheses has to be included.

Our work has evolved from the CONDENSATION algorithm [3, 8, 9] developed for contour tracking in visual clutter. Outlines and features of moving foreground objects, modeled as curves, are tracked in video sequences. Some elements in the background clutter may consist of objects similar to the foreground object, for instance when a person is moving past a crowd. The resulting ambiguity is solved by hypotheses tracking where probabilistic models of object shape and motion are applied to analyze the video-stream.

We propose to apply the CONDENSATION algorithm instead of Kalman filtering [4, 7]. Avoiding potential obstacles permits an effective path planning for mobile robot systems. For this, we show how the algorithm must be modified to deal with newly appearing objects. The motivation for choosing this new tracking approach is its ability to easily track multiple hypotheses and its simplicity. In addition, a constant computing time can be ensured which is very helpful for real-time applications.

Kalman filtering has been discussed thoroughly in the literature. A feature-based approach is presented in [2] where points or lines are updated with a Kalman filter. This kind of tracking process is well known in the computer vision community. Problematic is the subsequent grouping of the image features. Common motion constraints can be used to group these object features. The advantage of this approach is that even in the presence of partial occlusion, some features of the objects usually remain visible. Tracking based on object models is comparable to the feature-based approach. In [5, 12] Kalman filters are applied to estimate bounding contours of moving objects, where the contour extraction is based on motion and grey value boundaries. In dynamic scenes, the problem of partial occlusion occurs. In order to avoid an erroneous shift in the object trajectory an occlusion detection can be performed by a depth ordered detection of overlapping contours. A more complex approach using 3D model-based tracking is investigated in [1, 10, 11], where a parameterized 3D generic model is used to represent various types of cars. Admittedly it seems a bit risky to expect that the generic models can handle all kinds of cars that can be found on motorways. In addition dynamic models, which describe the motion of an object, can also be incorporated into a Kalman filter. The kinematic behavior of cars is represented in [16] and a multi-state model of the driver's behavior is discussed in [14].

$p(x_t\|\mathcal{Z}_t)$	:	the *a posteriori* density given the measurements
$p(x_t\|\mathcal{Z}_{t-1})$	:	the *a priori* density
$p(x_t\|x_{t-1})$	:	the process density describing the dynamics
$p(z_t\|x_t)$	:	the observation density

Figure 1. The probability distributions

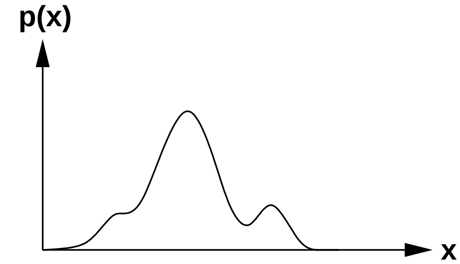

Figure 2. The knowledge of a single object is represented by the probability function $p(x)$. To simplify matters only the one-dimensional case is illustrated here. For realistic problems, the object state x is multi-dimensional and the density function is more complex.

In the case of multiple hypotheses tracking caused by occlusion or disappeared objects Kalman filters can rapidly lead to unwieldy levels of complexity. Hence a mechanism has to be implemented for Kalman filtering to control the evolution of hypotheses. In [15] we present such an aging approach where hypotheses survive until their age exceeds a predefined threshold.

In this paper we first introduce the mathematics needed to formulate the CONDENSATION algorithm. Secondly, we explain how it can be extended to track multiple objects and to cope with newly appearing objects and present applications. Finally, a discussion shows the differences to the original scheme [3, 8, 9] and also compares our approach to Kalman filtering [4, 7].

2. Mathematical Methods

The notation in this paper approximately follows [3, 8, 9]. The terminology is listed in Fig. 1.

Probability Distribution: An object is characterized by a state vector $x \in \mathbb{X}$. Assuming that we are not able to know the exact state, we describe the knowledge about an object by a probability function $p(x)$ (see Fig. 2).

Dynamics: As the observed scene changes over time, the probability function evolves to represent the altered object states. For computational reasons, the propagation is performed in discrete time steps. The dynamics for the evolution is described by a stochastic differential equation where the deterministic part of the equation models the system knowledge. The stochastic part allows us to model uncertainties.

Applying this differential equation, which can be of arbitrary order, the density function $p(x_t)$ depends only on the immediately preceding distribution $p(x_{t-1})$ but not on any function prior to $t-1$. So the dynamics is determined by the process density $p(x_t|x_{t-1})$. The process density used in our application is explained in Section 3.

Measurements: Let z_t be the measurement at time step t with history $\mathcal{Z}_t = \{z_0, \cdots, z_t\}$. For the tracking, we would like to update our estimate at each time step by incorporating the new measurement z_t. So instead of the density $p(x_t)$, we distinguish between the *a priori* density $p(x_t|\mathcal{Z}_{t-1})$ and the *a posteriori* density $p(x_t|\mathcal{Z}_t)$.

We can formulate the *a posteriori* density $p(x_t|\mathcal{Z}_t) = p(x_t|z_t, Z_{t-1})$ using Bayes' rule:

$$\begin{aligned} p(x_t|\mathcal{Z}_t) &= \frac{p(z_t|x_t, \mathcal{Z}_{t-1})\, p(x_t|\mathcal{Z}_{t-1})}{p(z_t|\mathcal{Z}_{t-1})} \\ &= k\, p(z_t|x_t, \mathcal{Z}_{t-1})\, p(x_t|\mathcal{Z}_{t-1}) \\ &= k\, p(z_t|x_t)\, p(x_t|\mathcal{Z}_{t-1}) \end{aligned} \quad (1)$$

where k is a normalization factor. The simplifications can be made using the assumption that the measurements are independent. A proof can be found in [9]. The observation density $p(z_t|x_t)$ gives the likelihood that a state x_t causes the measurement z_t, it is described in more detail in Section 3.

The *a priori* density $p(x_t|\mathcal{Z}_{t-1})$ is the result of applying the dynamic model to the *a posteriori* density $p(x_{t-1}|\mathcal{Z}_{t-1})$ of the previous time step:

$$p(x_t|\mathcal{Z}_{t-1}) = \int_{x_{t-1}} p(x_t|x_{t-1})\, p(x_{t-1}|\mathcal{Z}_{t-1}). \quad (2)$$

The complete tracking scheme first calculates the *a priori* density $p(x_t|\mathcal{Z}_{t-1})$ using the dynamic model and then evaluates the *a posteriori* density $p(x_t|\mathcal{Z}_t)$ given the new measurement:

$$p(x_{t-1}|\mathcal{Z}_{t-1}) \stackrel{dynamics}{\longrightarrow} p(x_t|\mathcal{Z}_{t-1}) \stackrel{measurement}{\longrightarrow} p(x_t|\mathcal{Z}_t). \quad (3)$$

Factored Sampling: In general the *a posteriori* density $p(x_t|\mathcal{Z}_t)$ is too complex and can not be evaluated simply in closed form. Also since $\mathbb{X}$, the space on which x is defined

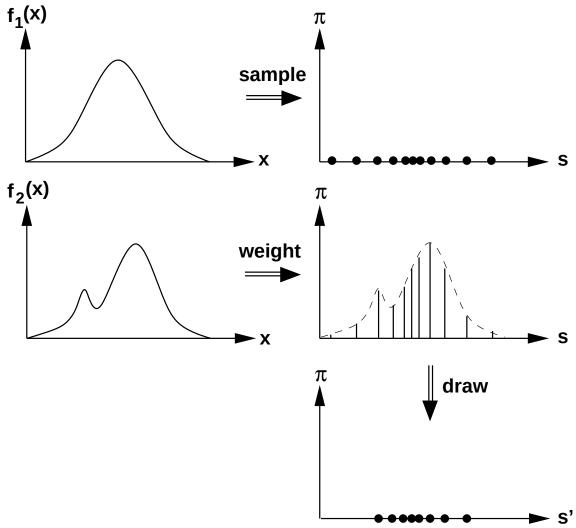

Figure 3. Factored sampling. A point set s is sampled randomly from the density $f_1(x)$. Each sample is assigned a weight $\pi^{(j)}$ in proportion to the density $f_2(x)$. A new sample set s' is generated by choosing N elements according to their weights.

is multi-dimensional and large, we can not sample $p(x_t|\mathcal{Z}_t)$ at regular intervals. Therefore, we must use iterative sampling techniques.

The factored sampling method [6] is used to find an approximation to a probability density

$$f(x) = f_2(x)\; f_1(x), \quad x \in \mathbb{X}. \tag{4}$$

A set of samples $s = \{s^{(1)}, \cdots, s^{(N)}\}$ with $s^{(n)} \in \mathbb{X}$ is drawn randomly from $f_1(x)$. By choosing a sample $s^{(j)}$ with probability

$$\pi^{(j)} = \frac{f_2(s^{(j)})}{\sum_1^N f_2(s^{(j)})}, \quad j = \{1, \cdots, N\} \tag{5}$$

from the sample set s, we calculate a new sample set s'. Its distribution tends to that of $f(x)$, as $N \to \infty$ (see Fig. 3).

The Condensation Algorithm: The CONDENSATION algorithm applies factored sampling iteratively to calculate the *a posteriori* density $p(x_t|\mathcal{Z}_t)$ according to Eq. 1. We always have a sampled distribution of the *a priori* probability $p(x_t|\mathcal{Z}_{t-1})$ ($f_1(x)$ in Eq. 4), so the initial creation of a sample set can be omitted in factored sampling.

An iteration step of the CONDENSATION algorithm starts with a sample set s representing the *a posteriori* density

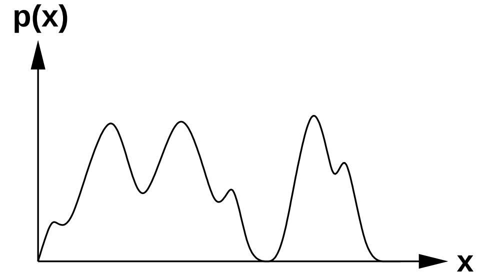

Figure 4. This single, multi-modal probability function represents the states of three different objects.

$p(x_{t-1}|\mathcal{Z}_{t-1})$ from the previous time step. We propagate s to obtain a new sample set s' according to the dynamic model, s' describes the *a priori* density $p(x_t|\mathcal{Z}_{t-1})$. Applying factored sampling, a set s'' is then drawn from s', where each element of the new set is chosen with probability $\propto p(z_t|x_t)$ so that s'' represents

$$p(x_t|\mathcal{Z}_{t-1})\; p(z_t|x_t) \propto p(x_t|\mathcal{Z}_t) \tag{6}$$

finishing the iteration step.

A more detailed treatment of the basic CONDENSATION algorithm can be found in [3, 8, 9].

Multiple Objects: In many applications it is necessary to track several objects simultaneously. The density distribution has the ability to represent the state of multiple objects with a single, multi-modal function. (see Fig. 4).

3. Extending the Condensation Algorithm

The original CONDENSATION algorithm was not designed to track multiple objects, although it is suggested in [3, 8, 9]. However, if we simply apply the method to our application, the results are not satisfactory. Measurements are only utilized to calculate the weights, but do not affect the states directly. So new objects that appear after the initialization can not be tracked. To include the measurements we modify the basic approach: During the factored sampling we select only $N - M$ samples instead of N, but we add M new samples based on the observations.

In our application we are interested in localizing objects to evade potentially harmful situations. As range sensors are used, the state vector for any object at time t contains the distance d_t, the relative velocity $\dot{d}_t$, the horizontal angle ψ_t and its change $\dot{\psi}_t$, the vertical angle θ_t and the corresponding change $\dot{\theta}_t$, as well as the object extensions e_t^0 and e_t^1:

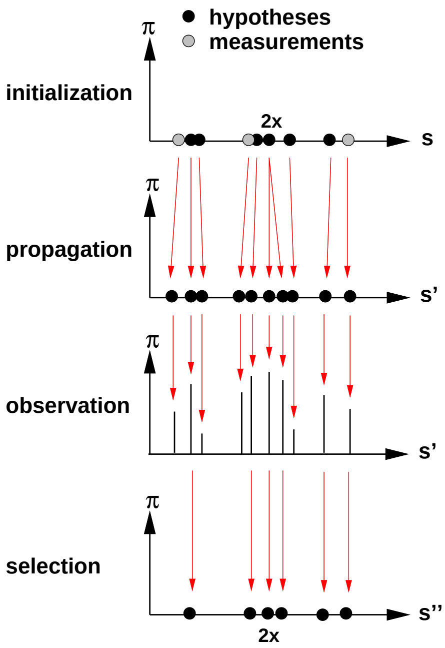

Figure 5. Information flow of one iteration step of the extended CONDENSATION algorithm.

$$x_t^T = [d_t \;\; \dot{d}_t \;\; \psi_t \;\; \dot{\psi}_t \;\; \theta_t \;\; \dot{\theta}_t \;\; e_t^0 \;\; e_t^1]. \tag{7}$$

At each time step a segmentation [13, 15] of the image is used to detect objects and provides the measurement vectors

$$z_t^T = [\tilde{d}_t \;\; \tilde{\psi}_t \;\; \tilde{\theta}_t \;\; \tilde{e}_t^0 \;\; \tilde{e}_t^1] \tag{8}$$

for all of them.

In the state vector we also manage elements which can not be measured directly but are inferred from the other elements.

The extended algorithm used for our application is charted in Fig. 5. For each iteration it can be divided into four parts, which we describe subsequently.

Initialization: A new sample set s is constructed from $N - M$ samples that represent the *a posteriori* density $p(x_{t-1}|\mathcal{Z}_{t-1})$ from the previous time step and from M samples that are added directly based on the measurements at time $t-1$. Using the observations guarantees that newly appearing objects flow into the tracking process. In the first iteration no results from a previous time step are available and therefore, the whole set s is initialized with values from the observations. Elements of the state vectors which can not be measured directly $(\dot{d}_t, \dot{\psi}_t, \dot{\theta}_t)$ are set to zero or to reasonable values based on the ego-motion of the mobile robot.

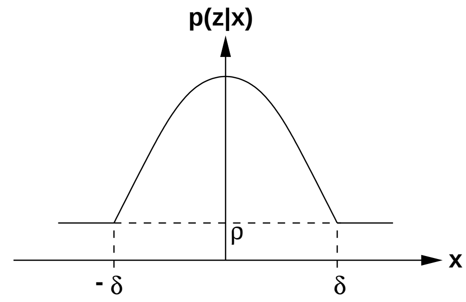

Figure 6. The observation density $p(z|x)$ is described by a truncated Gaussian.

Propagation: The sample set s' that approximates the *a priori* density $p(x_t|\mathcal{Z}_{t-1})$ is generated from s through the application of a dynamic model

$$x_t = A\,x_{t-1} + B\,w_t \tag{9}$$

where A defines the deterministic and Bw_t the stochastic component. This is the same model as would be used for a Kalman filter [4, 7]. In our application we use a first order model for A describing an object moving with constant velocity. Expanding this model to second order is straightforward.

w_t is a vector of normal random variables scaled by B so that BB^T is the covariance of the process noise. The process density $p(x_t|x_{t-1})$ is therefore a Gaussian distribution.

Observation: In the observation step we weight each element of the set s' in terms of the measurements by calculating the Euclidean distance between the common elements of s'_t and z_t. As observation density $p(z_t|x_t)$ we use

$$\pi_t^{(n)} = \begin{cases} e^{-\frac{1}{2\sigma^2}u^2} & u < \delta \\ \rho & otherwise \end{cases} \tag{10}$$

$$u = \min_{(m)} |s_t'^{(n)} - z_t^{(m)}|. \tag{11}$$

The observation density $p(z_t|x_t)$ is a truncated Gaussian and has a minimal constant value ρ (see Fig. 6). This permits that hypotheses of unseen objects might survive the next time step.

Selection: A fixed size of $N - M$ samples can now be selected from the set s'. A particular $s'^{(j)}$ is drawn (with replacement) randomly, by choosing it with probability $\pi^{(j)}$. Some elements, especially those with high weights, may be chosen several times, leading to identical elements in the new set s''. In the propagation step, they will be split

1. **Initialize** the sample set s_t based on the hypotheses and the measurements:
   $s_t^{(n)} = s''^{(l)}_{t-1} \cup z_{t-1}^{(m)}$
   $n = \{1, \cdots, N\}, l = \{1, \cdots, N-M\}$
   and $m = \{1, \cdots, M\}$

2. **Propagate** each sample from the set s_t by a linear stochastic differential equation:

   $s_t'^{(n)} = A\, s_t^{(n)} + B\, w_t^{(n)}$

   where $w_t^{(n)}$ is a vector of standard normal random variables and BB^T is the process noise covariance.

3. **Observe** the measurements:
   (a) weight each sample of the set s_t'
   $\pi_t^{(n)} = \begin{cases} e^{-\frac{1}{2\,\sigma^2}\,u^2} & u < \delta \\ \rho & otherwise \end{cases}$
   $u = \min_{(m)} |s_t'^{(n)} - z_t^{(m)}|$

   (b) calculate the normalized cumulative probabilities
   $c_t^{(0)} = 0$
   $c_t^{(n)} = c_t^{(n-1)} + \pi_t^{(n)}$
   $c_t^{(n)} = \frac{c_t^{(n)}}{c_t^{(N)}}$

4. **Select** $N - M$ samples from the set s_t' with probability $\pi_t^{(n)}$:
   (a) generate a uniformly distributed random number $r \in [0, 1]$
   (b) find, by binary search, the smallest j for which $c_t^j \geq r$
   (c) set $s_t''^{(l)} = s_t'^{(j)}$

Figure 7. An iteration step of the extended CONDENSATION algorithm.

due to the stochastic component of the dynamic model. Others with relatively low weights may not be chosen at all.

The programming details for one iteration step are given in Fig. 7.

4. Comparisons to the Original Condensation Algorithm

To our knowledge this is the first application that uses the CONDENSATION algorithm to track *multiple* objects in a dynamic scene.

The challenge in mobile robot applications is their dynamic character, objects constantly enter or leave the field of view. In order to cope with this situation the original CONDENSATION approach had to be extended to deal with newly appearing objects. We do this by applying a different initialization scheme that incorporates the measurements.

The original CONDENSATION algorithm was used to track contours in grey-scale images where a parametric curve representation describes the state of an object. In our application we utilize range images as we are interested in localizing obstacles. The geometrical parameters characterize our state, respectively measurement vectors. We also manage state elements which can not be measured directly but are inferred from the other elements.

Furthermore, we have only moderately cluttered background in difference to [3, 8, 9], so we are able to detect objects using a high level segmentation scheme. Accordingly we also employ a different observation scheme.

5 Comparisons to Kalman Filtering

In contrast to the CONDENSATION tracker a Kalman filter [4, 7] is unimodal and its density function evolves as a Gaussian. A single Kalman filter is thus able to track only one object.

Tracking several objects can be done by using a Kalman filter for each object. However, if multiple hypotheses are required for each object, this quickly leads to a combinatorial explosion and the need of a complex object management system. As the CONDENSATION algorithm has the ability to deal with multi-modal distributions, multiple hypotheses can be easily tracked simultaneously.

The absence of the Riccati equation – which appears in the Kalman filter – reduces the computational complexity of the extended CONDENSATION tracker. Furthermore, the fixed number of samples leads to a constant running time per iteration step. Of course, for larger N we are able to achieve a better approximation of the density function, although this has its limit determined in accordance to the available processing time.

6. Results

To show the ability of the extended CONDENSATION algorithm we present results on different data sources. We illustrate the method on range image sequences of a line and a matrix sensor. In both applications we are interested in evading potential obstacles for an autonomous navigation. For safety reasons we are dealing with the worst case and will already start an evasive action when one object state is critical. We will demonstrate especially that we can cope

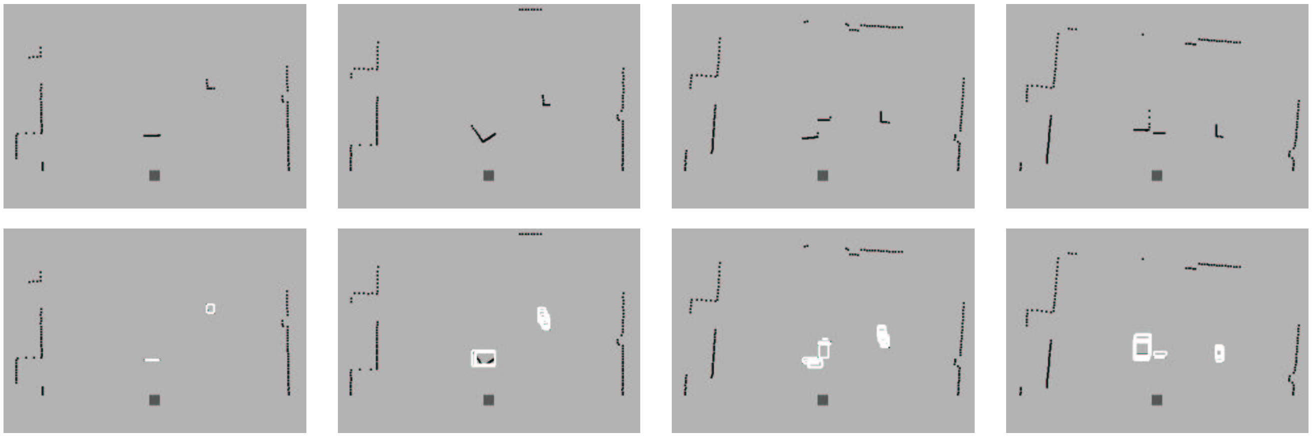

Figure 8. Tracking multiple obstacles which are closer than $5m$ in a real indoor scene with a line sensor. Result of the extended CONDENSATION tracker are illustrated. Each hypothesis of an object state is represented by a bounding box and overlaid to the range data which are plotted as points.

with new or unseen objects.

6.1 Range Images Using a Line Sensor

We tested the extended CONDENSATION algorithm on several image sequences recorded by a line sensor. This SICK LMS 200 was mounted on a mobile robot at a height of $60cm$ above ground for indoor applications. The sequences were produced at the Institute of Robotic at the ETH with a frame rate of 1.5 images per second, an image size of 181 pixels and a field of view of $180°$. The maximal range of the line sensor is $8m$.

Looking at the scene in Fig. 8 from the top view, the upper row indicates the range data plotted as points. The walls are visible as lines along the border where the back wall is not in the range of the sensor at the beginning. The position of the sensor is shown by a grey square in each image. In the lower row, the results of the extended CONDENSATION tracker are overlaid to the range information. The state of each object is represented by a bounding box which can be calculated from the object center (d_t, ψ_t, θ_t) and the object dimension (e_t^0, e_t^1). We are interested in tracking objects within a certain safety zone, which we have chosen as $5m$. In this example an indoor scene with three obstacles is shown where one object is occluded at the beginning. Two of the obstacles are stationary while the robot and one object move with different velocities. The moving object starts a left turn in the second image from the left. The experiment was run using $N = 20$ samples for each iteration step.

We have tested the capability of tracking the multi-modal distribution on the shown sequence. The result for the first 30 images is plotted in Fig. 9. To simplify matters only the distribution of the horizontal image coordinate of the object centers is shown instead of the multi-dimensional object states. The first initialization of the density function was only based on the measurements and therefore sharp peaks characterize our objects. After some time the distribution blurs as several hypotheses of an object develop. This Figure illustrates appropriately the tracking of multiple hypotheses of one object. For example the overtaking evokes several hypotheses but only few survive over time.

6.2 Range Images from a Matrix Sensor

Results of a matrix sensor are presented for a public domain image sequence, which can be found at **http://marathon.csee.usf.edu/range/DataBase.html** in the USF database. This range image sequence was recorded by a Odetics LADAR sensor, which was mounted on the HERMIES III robot at the CESAR lab at Oak Ridge National Labs. The sequence has an image size of 128×128 pixels and a field of view of $60° \times 60°$.

As in the previous application we are interested in tracking obstacles within a certain safety zone. We focus only on objects which have at least a dimension of $5cm \times 5cm$ and which are closer to the sensor than $1m$. In Fig. 10 we track three different objects. From the observed scene a reflectance image (left column) and a corresponding registered range image (right column) was recorded whereas the range is represented by different grey-levels. The object states are indicated by their bounding boxes and overlaid to the corresponding range image. This example illustrates the ability of tracking multiple objects with a single CONDENSATION tracker. In addition new objects which enter the safety zone of $1m$ are incorporated into the tracking process. For example the pyramid in the second image from

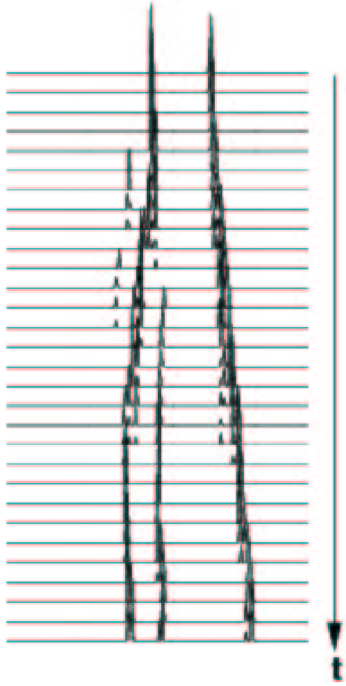

Figure 9. Propagation of a multi-modal density function at discrete time steps. The distribution of the object centers is plotted in the horizontal image dimension.

the top and the cupboard in the third image appear subsequently as new potential obstacles. We can observe that new objects are based on the measurements at the beginning of the tracking but develop multiple state hypotheses over time. For the experiment we used $N = 20$ samples for each iteration step.

The effect of incorporating the measurements into the tracking process is presented in Fig. 11. We show the hypotheses of the object states as white bounding boxes and in black we chart the incorporated measurements. This example illustrates the survival of an object hypothesis even if it was not detected in the current image. The cupboard for example is not marked as measurement and was therefore not in the safety zone for this frame. Nevertheless the object hypothesis is still known in the tracking process and represented by a white bounding box. As we are dealing with occluded and disappeared objects, such states should remain temporary in the tracking process for safety reasons.

7. Conclusion

A stochastic tracking approach has been presented that can handle multiple hypotheses and newly appearing ob-

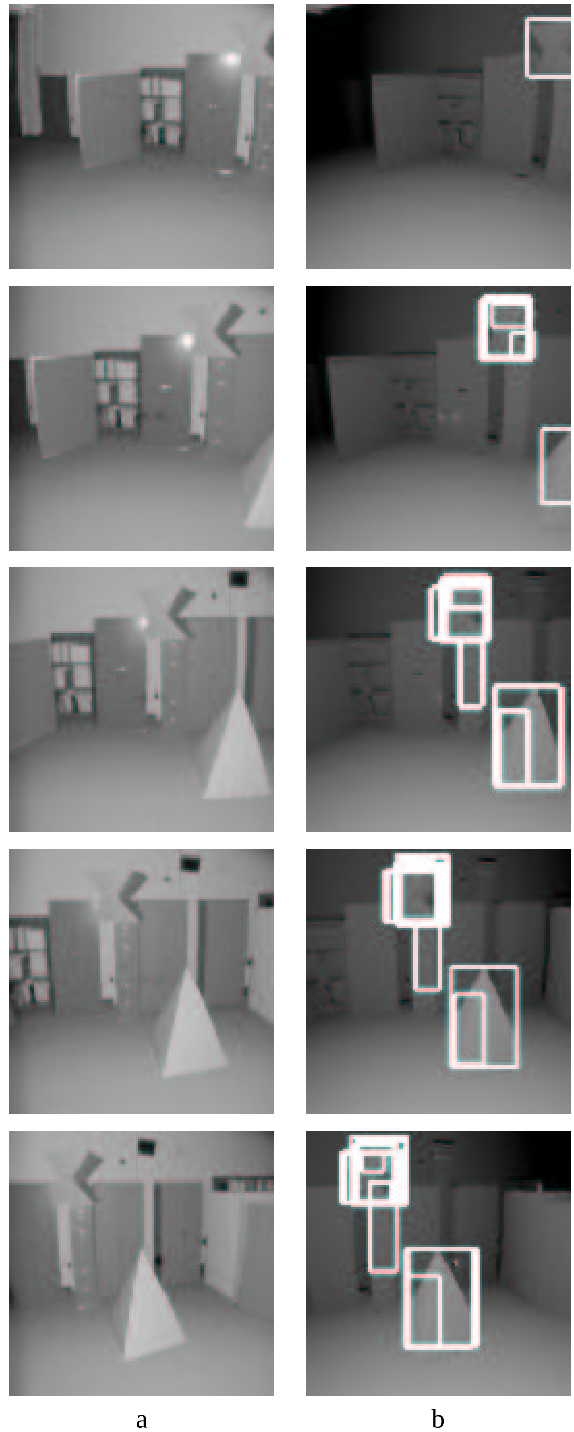

Figure 10. Tracking multiple obstacles of a real indoor scene with a matrix sensor. Only objects closer than $1m$ and larger than $5cm \times 5cm$ are considered. a: The reflectance images recorded by a mobile robot. b: The corresponding registered range images. Each hypothesis of an object state is represented by a bounding box.

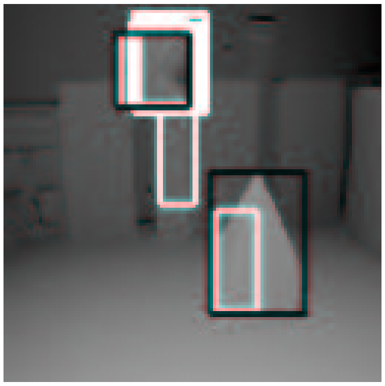

Figure 11. The incorporation of the measurements is illustrated. The white bounding boxes show the hypotheses of the object states and black represents the observations flowing into the tracking process.

jects. A probability density function is used to describe the likely state of the objects. We have shown how the system dynamics and measurements are included in this probabilistic framework and how the extended CONDENSATION algorithm can be applied to represent the density function with a fixed number of samples. A discussion points out the differences to the original CONDENSATION algorithm and also compares our approach to Kalman filtering.

As application we concentrated on mobile robot systems based on line and matrix image sequences. Avoiding potential obstacles permits an effective path planning. Characteristic for such images is the ambiguity caused by several objects within the scene and the incomplete data information by disappeared or occluded objects.

The extended CONDENSATION algorithm can deal with many of the problems that are typically encountered in tracking, such as an arbitrary number of objects, newly appearing, disappear or occluded objects and has real time capability. Additionally this tracking method is simple to implement.

A limitation of the scheme may be the form of the output data. Depending on the application, the probability distribution may have to be interpreted in different ways. For example if the importance of certain object states is needed, it can be determined by the weighting in the observation step or by local clustering.

Acknowledgments

This research was partially supported by MINORA, a project of the Swiss priority program OPTIQUE II, funded by the ETH Council. Further we thank Sjur Vestli and Felix Wullschleger from the Institute of Robotic, ETH Zurich, to provide us with range image sequences from the SICK LMS 200. The authors would like to acknowledge the CESAR lab at Oak Ridge National Labs, especially Judd Jones, Ole Henry Dorum, Fank Sweeney and Reinhold Mann, for making it possible for us to use the range image sequences recorded by the Odetics LRF mounted on the HERMIES III robot system.

References

[1] K. D. Baker and G. D. Sullivan. Performance assessment of model-based tracking. In *IEEE Workshop on Applications of Computer Vision*, pages 28–35, 1992.

[2] D. Beymer, P. McLauchlan, B. Coifman, and J. Malik. A real-time computer vision system for measuring traffic parameters. In *Computer Vision and Pattern Recognition*, pages 495–501, 1997.

[3] A. Blake and M. Isard. *Active Contours*. Springer-Verlag, 1998.

[4] A. Gelb. *Applied Optimal Estimation*. MIT Press, 1996.

[5] S. Gil, R. Milanese, and T. Pun. Combining multiple motion estimates for vehicle tracking. In *European Conference on Computer Vision*, volume 2, pages 307–320, 1996.

[6] U. Grenander, Y. Chow, and D. M. Keenan, editors. *HANDS, A Pattern Theoretic Study of Biological Shapes*. Springer-Verlag, 1991.

[7] M. S. Grewal and A. P. Andrews. *Kalman Filtering*. Prentice Hall, 1993.

[8] M. Isard and A. Blake. Contour tracking by stochastic propagation of conditional density. In *European Conference on Computer Vision*, volume 1, pages 343–356, 1996.

[9] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal on Computer Vision*, 29(1):5–28, 1998.

[10] M. Kilger. A shadow handler in a video-based real-time traffic monitoring system. In *IEEE Workshop on Applications of Computer Vision*, pages 11–17, 1992.

[11] D. Koller, K. Daniilidis, and H.-H. Nagel. Model-based object tracking in monocular image sequences of road traffic scenes. *International Journal of Computer Vision*, 10:257–281, 1993.

[12] D. Koller, J. Weber, and J. Malik. Robust multiple car tracking with occlusion reasoning. In *European Conference on Computer Vision*, pages 189–196, 1994.

[13] E. B. Meier and F. Ade. Object detection and tracking in range image sequences by separation of image features. In *IEEE International Conference on Intelligent Vehicles*, pages 176–181, 1998.

[14] A. Pentland and A. Liu. Toward augmented control systems. In *IEEE Intelligent Vehicles*, pages 350–355, 1995.

[15] K. Sobottka, E. Meier, F. Ade, and H. Bunke. *Modeling and Planning for Sensor Based Intelligent Robot Systems*, chapter Toward Smarter Cars. Springer-Verlag, to appear 1999.

[16] L. Zhao and C. Thorpe. Qualitative and quantitative car tracking from a range image sequence. In *Computer Vision and Pattern Recognition*, pages 496–501, 1998.