

Tracking Cars in Range Images Using the Condensation Algorithm

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 temporarily disappear.

In this paper we present a stochastic tracking 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 with data from different sources are shown for automotive applications.

1. Introduction

This paper is concerned with object tracking using a stochastic approach based on the CONDENSATION algorithm. We apply this new technique to driver assistance systems using coarse range image sequences. Handling occlusion or unseen objects due to coarse or incomplete data 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 hypotheses has to be included.

Our work has evolved from the CONDENSATION algorithm [7, 8] 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 [3, 6] which 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. Problems arise in the subsequent grouping of image features in this tracking process. Common motion constraints can be used to group the 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 [10, 4] Kalman filters are applied to estimate bounding contours of moving objects, where the contour extraction is based on motion and grey value boundaries. In highway scenes with heavy traffic, 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 [9, 1], 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 [16, 12].

In the case of multiple hypotheses tracking caused by occlusion or temporarily 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 [13] 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 [7, 8] and also compares our approach to Kalman filtering [3, 6].

2. Mathematical Methods

The notation in this paper approximately follows [7, 8]. 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

$p(x_t \mathcal{Z}_t)$:	the <i>a posteriori</i> density given the measurements
$p(x_t \mathcal{Z}_{t-1})$:	the <i>a priori</i> 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.

the exact state, we describe the knowledge about an object by a probability function $p(x)$. However, in our application it is necessary to handle several objects simultaneously. The density distribution has even the ability to represent the state of multiple objects with a single, multi-modal function.

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 of 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: We would now like to introduce measurements into our scheme. Let z_t be the measurement at time step t with history $\mathcal{Z}_t = \{z_0, \dots, z_t\}$. 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 $p(x_t | \mathcal{Z}_t) = p(x_t | z_t, \mathcal{Z}_{t-1})$ using Bayes' rule:

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

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 [8]. 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:

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

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

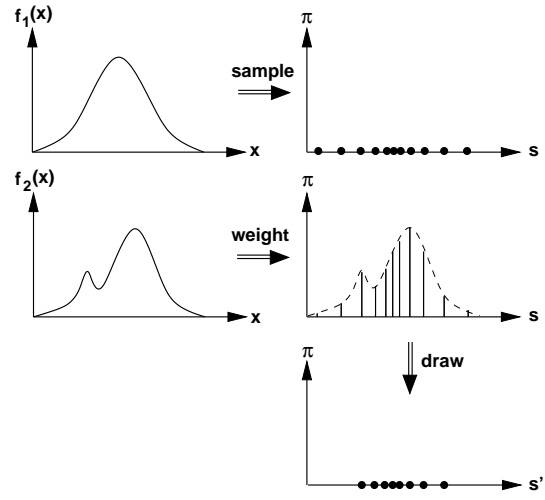


Figure 2. Factored sampling. A point set s is sampled randomly from the density $f_1(x)$. Each sample is assigned a weight $\pi^{(j)}$ in proportion to the density $f_2(x)$. A new sample set s' is generated by choosing N elements according to their weights.

evaluates the *a posteriori* density $p(x_t | \mathcal{Z}_t)$ given the measurements:

$$p(x_{t-1} | \mathcal{Z}_{t-1}) \xrightarrow{\text{dynamics}} p(x_t | \mathcal{Z}_{t-1}) \xrightarrow{\text{measurement}} p(x_t | \mathcal{Z}_t). \tag{3}$$

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 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 [5] 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 $\{s^{(1)}, \dots, s^{(N)}\}$ with $s^{(n)} \in \mathbb{X}$ is drawn randomly from $f_1(x)$. By choosing a sample $s^{(j)}$ with probability

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

from the sample set s , we calculate a new sample set s' . Its distribution tends to that of $f(x)$, as $N \rightarrow \infty$ (see Fig. 2).

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 $p(x_{t-1}|\mathcal{Z}_{t-1})$ 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

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

finishing the iteration step.

A more detailed treatment of the basic CONDENSATION algorithm can be found in [7, 8].

3. Extending the Condensation Algorithm

The original CONDENSATION algorithm was not designed to track multiple objects, although it is suggested in [7, 8]. 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 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 a driver assistance system 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 to the object and its change $\dot{\psi}_t$, the vertical angle θ_t and the corresponding change $\dot{\theta}_t$, as well as the width w_t and the height h_t :

$$x_t^T = [d_t \ \dot{d}_t \ \psi_t \ \dot{\psi}_t \ \theta_t \ \dot{\theta}_t \ w_t \ h_t]. \quad (7)$$

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

$$z_t^T = [\tilde{d}_t \ \tilde{\psi}_t \ \tilde{\theta}_t \ \tilde{w}_t \ \tilde{h}_t] \quad (8)$$

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 shown in Fig. 3. 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

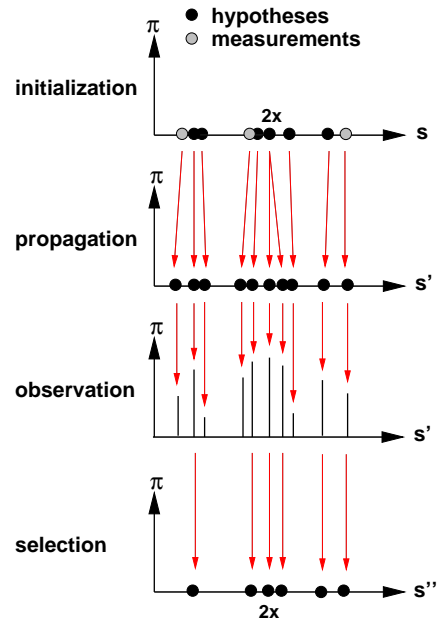


Figure 3. Information flow of one iteration step of the extended CONDENSATION algorithm.

the observations.

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

$$x_t = A x_{t-1} + B w_t \quad (9)$$

where A defines the deterministic and Bw_t the stochastic component. This is the same model as would be used for a Kalman filter [3, 6]. 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

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

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

The observation density $p(z_t|x_t)$ is a truncated Gaussian and has a minimal constant value ρ . 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

1. **Initialize** the sample set s_t based on the hypotheses and the measurements:

$$s_t^{(n)} = s_{t-1}^{(l)} \cup z_{t-1}^{(m)}$$

$$n = \{1, \dots, N\}, l = \{1, \dots, N - M\}$$

$$\text{and } m = \{1, \dots, M\}$$

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' :

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

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

- (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 4. An iteration step of the extended CONDENSATION algorithm.

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

The programming details of one iteration step are given in Fig. 4.

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 traffic scenes 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 the width, height and position of an obstacle. These geometrical parameters characterize our state vector, 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 contrast to [7, 8], 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 the density function used by a Kalman filter [3, 6] is unimodal and 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 – and the fixed number of hypotheses reduce the computational complexity and lead to a constant running time per iteration step.

6. Results

In the project MINORA a universal miniaturized optical range camera [15, 14] is being developed. Since this range sensor is still under development, range image sequences of real traffic scenes are not yet available. Therefore, we use simulated range image sequences and scaled range images recorded by an ABW range scanner (ABW GmbH, Germany) to develop and test our tracking algorithm.

We have tested the capability of tracking a multi-modal distribution on a 100-frame sequence with a frame rate of 25 images per second. This simulated sequence has an image size of 16×64 pixels and a field of view of $2.38^\circ \times 10.0^\circ$. In Fig. 5 the result of the first 30 images is plotted. On the right, a subset of the range images is represented by different grey-levels; black represents undefined pixels where no meaningful measurement was obtained. This typical traffic scene contains several road posts and two cars. One car is occluded at the beginning by another passing vehicle. On the left, the propagation of the density function is plotted. To simplify matters only the distribution of the horizontal image coordinate of the object centers is shown instead of the

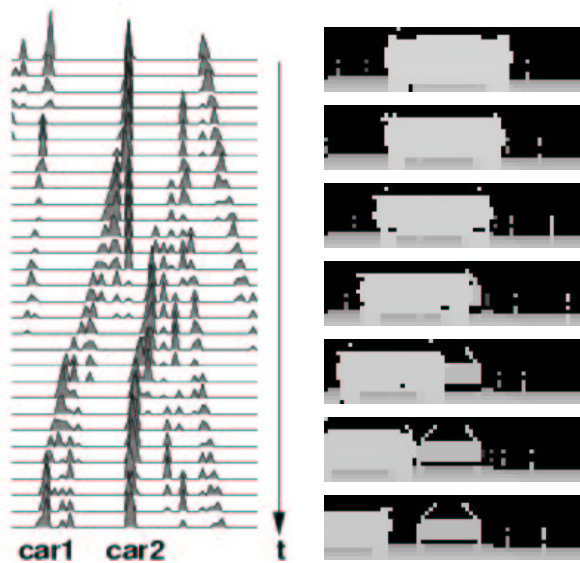


Figure 5. Propagation of a multi-modal density function at discrete time steps. The distribution of the object centers is plotted in the horizontal image dimension.

multi-dimensional object state. The first initialization of the density function was based on the measurements and therefore sharp peaks characterize our objects. After some time steps the distribution blurs as several hypotheses of an object develop. As our data has a low resolution, small objects – like a road post – are not visible in each frame but are tracked nevertheless. The corresponding peaks of these objects in the probability function continuously get smaller. Furthermore, this Figure illustrates appropriately the tracking of multiple hypotheses of one object. For example the lane changing of the first car evokes several hypotheses but only few survive over time. The experiment was run using $N = 40$ samples for each time step.

Traffic scenes are distinguished by their high dynamic character. The number of tracked objects changes permanently as objects enter or leave the sensor’s field of view. Our tracking algorithm must be able to handle such modified conditions. In Fig. 6 two cars, one driving ahead and the other in the oncoming traffic, can be observed. The range sensor has a relatively narrow field of view whereas the driver has a wider field of view. The state of each hypothesis is represented by a bounding box which can be calculated from the object center (d_t, ψ_t, θ_t) and the object dimensions (w_t, h_t) . By projecting the bounding boxes into the image several hypotheses collapse into a single box. The third image from the top shows that the hypothesis of an invisible object – the road post on the left side – survives some time steps.

In Fig. 7 an example of a range image sequence of a scaled toy traffic scenario acquired with a structured light sensor is shown. The scene contains an oncoming car and several trees in the background. This sequence has a length of 25 images and a frame rate of 25 images per second. The image size is

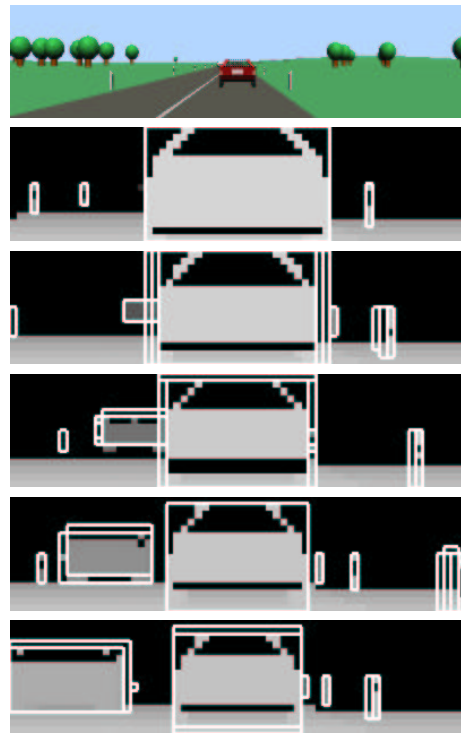


Figure 6. Tracking multiple obstacles in a simulated traffic scene. The state of each hypothesis is pictured by a bounding box in the corresponding range image where the top image shows the view from the driver’s seat.

65×144 pixels and the field of view is $6.04^\circ \times 14.1^\circ$. A robust tree tracking is a demanding task in this sequence. Sparse information regarding the tree tops make an exact localizations impossible. Several hypotheses of an object represent this uncertainty. In this experiment $N = 60$ samples were used at each time step.

More examples on real data can be found in [11].

7. Conclusion

A stochastic tracking approach has been presented, that can handle multiple hypotheses and newly appearing objects. 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 driver assistance systems based on low-resolution range image sequences. Characteristic for such images is the ambiguity caused by several objects within the scene and the incomplete data information by unseen objects.

The extended CONDENSATION algorithm can deal with

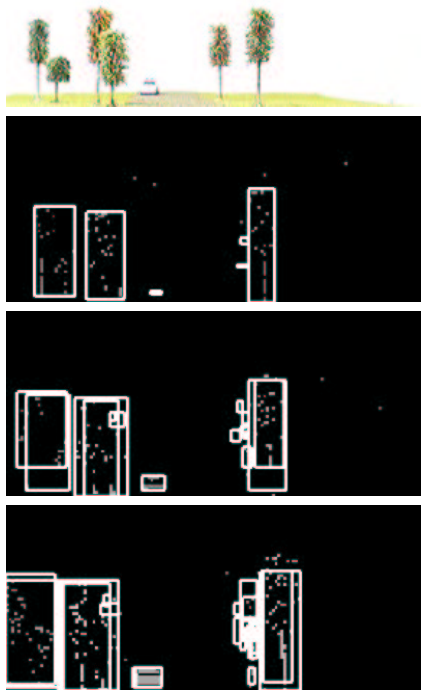


Figure 7. Tracking multiple obstacles in a toy traffic scenario. The images have been acquired by a structured light sensor.

many of the problems that are typically encountered in tracking, such as an arbitrary number of objects, newly appearing, disappearing or occluded objects and it 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 Prof. Horst Bunke, Dr. Xiaoyi Jiang and Karin Sobottka of the Institute of Computer Science and Applied Mathematics, University of Bern, Switzerland, for providing us with range image sequences of toy traffic scenes to test our approach.

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. Gelb. *Applied Optimal Estimation*. MIT Press, 1996.
- [4] 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.
- [5] U. Grenander, Y. Chow, and D. M. Keenan, editors. *HANDS, A Pattern Theoretic Study of Biological Shapes*. Springer-Verlag, 1991.
- [6] M. S. Grewal and A. P. Andrews. *Kalman Filtering*. Prentice Hall, 1993.
- [7] 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.
- [8] M. Isard and A. Blake. Condensation – conditional density propagation for visual tracking. *International Journal on Computer Vision*, 29(1):5–28, 1998.
- [9] 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.
- [10] D. Koller, J. Weber, and J. Malik. Robust multiple car tracking with occlusion reasoning. In *European Conference on Computer Vision*, pages 189–196, 1994.
- [11] E. B. Meier and F. Ade. Using the condensation algorithm to implement tracking for mobile robots. In *3rd European Workshop on Advanced Mobile Robots*, 1999.
- [12] A. Pentland and A. Liu. Toward augmented control systems. In *IEEE Intelligent Vehicles*, pages 350–355, 1995.
- [13] K. Sobottka, E. Meier, F. Ade, and H. Bunke. *Modeling and Planning for Sensor Based Intelligent Robot Systems*, chapter Toward Smarter Cars. Springer-Verlag, 1999.
- [14] T. Spirig, M. Marley, and P. Seitz. The multi-tap lock-in CCD with offset subtraction. *IEEE Transactions on Electron Devices*, 44(10):1643–1647, October 1997.
- [15] T. Spirig, P. Seitz, O. Vietze, and F. Heitger. The lock-in CCD — two-dimensional synchronous detection of light. *IEEE Journal of Quantum Electronics*, 31(9):1705–1708, September 1995.
- [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.