

Towards Smarter Cars

Karin Sobottka¹, Esther Meier², and Frank Ade² and Horst Bunke¹

¹ Institute of Computer Science and Applied Mathematics, University of Bern,
CH-3012 Bern, Switzerland, {sobottka,bunke}@iam.unibe.ch

² Communication Technology Lab, Image Science, Swiss Federal Institute of
Technology, CH-8092 Zurich, Switzerland, {ebmeier,ade}@vision.ee.ethz.ch

Abstract. Most approaches for vision systems use greyscale or color images. In many applications, such as driver assistance or presence detection systems, the geometry of the scene is more relevant than the reflected brightness information and therefore range sensors are of increasing interest.

In this paper we focus on an automotive application of such a range camera to increase safety on motorways. This driver assistance system is capable of automatically keeping the car at an adequate distance or warning the driver in case of dangerous situations.

The problem is addressed in two steps: obstacle detection and tracking. For obstacle detection two different approaches are presented based on slope evaluation and computation of a road model. For tracking, one approach applies a matching scheme, the other uses a Kalman filter. Results are shown for several experiments.

1 Introduction

Modern everyday life, characterized by an ever increasing interaction between man and machine, leads to a growing number of potentially harmful situations through accidents, malfunctions or human oversight. Related to this is our investigation into reliable presence detection systems based on coarse range images.

In the project MINORA, which is part of the Swiss priority program OPTIQUE II, a universal miniaturized optical range camera is developed [18, 19]. This new sensor, working in the near infrared, will be fast, cheap and can supply 3D information with high accuracy. Necessary trade-offs however, mean that the sensor provides range images which are coarse (resulting from the need for inexpensive sensor and computing hardware) and incomplete (due to insufficient or saturated reflections from targets). For measuring distances ranging from a few meters to a few hundred meters the sensor is restricted to a narrow field of view. As a consequence the entire road width is within the sensor's field of view at a distance of about 20 meters. Accordingly, we focus on well ordered environments such as motorways where this restriction is tolerable.

Two main applications of the universal range camera are targeted in the project MINORA:

1. Automotive applications such as driver assistance systems with distance measurements of some hundred meters and a relatively narrow field of view to increase safety on motorways.
2. Safety and surveillance applications, such as the control of automatic sliding or revolving doors with distances of a few meters and a wide field of view.

While other project partners work on the electro-optical and computing components of the proposed sensor, our task is the development of algorithms for the interpretation of coarse and incomplete range image sequences. Although the sensor used in both the safety/surveillance and the automotive applications is the same, a discussion of the theory behind both applications is beyond the scope of a single paper. In the following we focus on the automotive application.

The aim of this paper is to present and discuss different methods for obstacle detection and tracking for a driver assistance system based on coarse range image sequences. Robust obstacle detection and tracking is performed to ensure collision avoidance. All potential obstacles have to be recognized within the scene. Two obstacle detection methods based on vertical slope evaluation and computation of a road model are presented. Furthermore, obstacles have to be tracked over time employing a matching scheme or Kalman filtering to predict dangerous situations. Due to the availability of 3D information and low resolution of range images a processing in real-time is ensured for all approaches at any discrete point of time.

We show results with simulated and scaled range images. Real data will be available later in the project.

2 Types of Range Images

Range imaging is a key technology for many applications where it is important to know the geometrical description of the observed scene. Although several measuring techniques exist, for example, stereo, structured light, triangulation, structure from motion, range from focusing, and time of flight, the literature is not too abundant. The techniques can be classified as either active or passive, where active methods involve a controlled energy source. Some ranging systems have already been successfully applied to niche applications, but none can claim to be a universal sensor.

Only little research was done on the interpretation of range image sequences for outdoor applications such as intelligent cruise control. Table 1 gives examples of current research on this topic and summarizes the advantages and disadvantages of different kinds of range sensors. To the knowledge of the authors, they have not been described and discussed in sufficient details in the open literature.

As can be seen in Table 1, the optical range camera that is developed within the MINORA project is, due to synchronous image sensing, well suited for automotive applications. Like other laser range scanners it is insensitive to ambient light and weather conditions and provides range data with a high accuracy in a short acquisition time. But novel algorithms have to be developed to handle the low resolution and incompleteness of the range data.

Table 1. Comparison of different kind of sensors

Sensor	Lit.	Advantages	Disadvantages
Stereo vision	[3]	- high resolution - brightness and range information available	- sensitive to illumination - calibration necessary - correspondence problem
Light stripe range scanner	[1]	- high resolution - dense depth map	- restricted to indoor appl. - calibration necessary - sensitive to illumination - long acquisition time
Depth from motion	[2]	- low cost	- low accuracy - dependent on motion segm. - sparse depth map
Millimeter-wave radar	[15], [21]	- long distance detection - high accuracy - insensitive to weather cond.	- long acquisition time (for imaging radar) - incomplete data - high cost
Ultrasonic sensor	[14]	- low cost	- low lateral accuracy - sensitive to wind - short distance measurement - incomplete data
Mechanically scanned optical radar	[5], [20]	- long distance measurement - high accuracy - insensitive to weather cond.	- long acquisition time - incomplete data - high cost
Optical range camera with synchronous image sensing	[11],[16],[17]	- low cost - high accuracy - insensitive to weather cond. - short acquisition time	- low resolution - incomplete data

3 An Automotive Application

Vision based driver assistance improves traffic safety by warning the driver, e.g. by an acoustic or visual signal, in case of dangerous situations. An example for a scenario, in which range imagery is used for vision based driver assistance, is depicted in Fig. 1. As can be seen, a range camera is fixed near the front bumper of a vehicle and acquires a stream of range images over time. The field of view of the range sensor is described by the angular width of image sensing. Assuming a geometry of the sensor as shown in Fig. 2, ϕ denotes half of the angular width in vertical direction and θ denotes half of the angular width in horizontal direction. Collision avoidance, then, is a sequence of operations including obstacle detection and tracking. In obstacle detection, the vision system checks if there is any object in the sensor's field of view that interferes with the path of the sensor vehicle. Such obstacles may be static or mobile. For obstacle tracking a sequence of frames is considered and feature correspondences are determined to



Fig. 1. Using range imagery for vision based driver assistance (with friendly permission of A.D.C. GmbH, Germany)

derive a robust estimation of the positions of obstacles and information about their temporal behaviour.

3.1 Obstacle Detection

Obstacle detection schemes for natural environments should be able to detect obstacles of any shape, e.g. cars, pedestrians, posts, trees, bushes and traffic signs. For that purpose a general obstacle model seems to be more appropriate than several specific 3D object models.

In the context of automotive applications, an obstacle can be defined as any region the sensor car should not traverse. This criterion can be examined in two

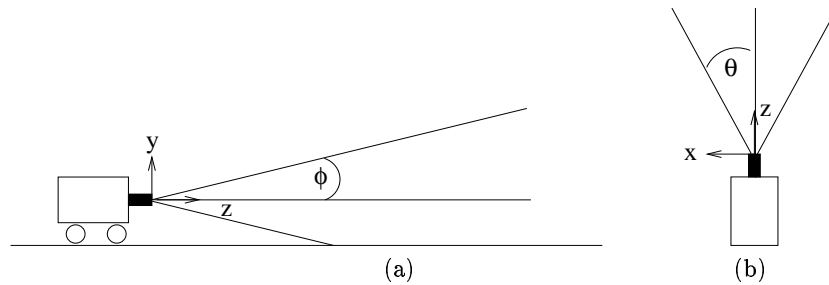


Fig. 2. Geometry of sensor: (a) vertical angular extent and (b) horizontal angular extent (top view)

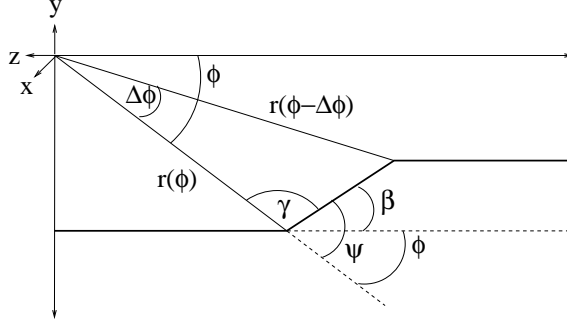


Fig. 3. Geometrical relationships for radial slope computation

different ways. On the one hand, the traversability of the environment in the sensor's field of view can directly be checked by slope evaluation. In this case a region with steep slope indicates the presence of an obstacle. On the other hand obstacles lie on top of the road. From an estimate of the location of the road we can separate the image into ground and obstacles. Both approaches allow robust obstacle detection and, combined with a tracking scheme, they can be employed for collision avoidance.

Radial Slope A straightforward approach to checking the traversability in the sensor's field of view is the evaluation of the slope of a region. It can be computed very efficiently in radial direction. Thus determining a region with steep slope indicates the presence of an obstacle.

Also in [5] obstacles are defined as untraversable areas. An ideal obstacle detection method which uses a complete vehicle model is applied to continuous range data. Specific features, such as discontinuities and slope, are extracted to indicate an obstacle. In [22] partial derivatives of the range in a spherical coordinate system are used to indirectly measure the Cartesian slopes. This approach is used for obstacle detection by an autonomous car.

Based on the sensor geometry illustrated in Fig. 2, for each pair of two vertically adjacent laser beams with depth $r(\phi)$ and $r(\phi - \Delta\phi)$, respectively, slope β can be computed based on the following relationships:

$$\beta = \psi - \phi \quad (1)$$

$$\psi = 180^\circ - \gamma \quad (2)$$

$$\tan\psi = -\tan\gamma = \frac{r(\phi - \Delta\phi) \cdot \sin\Delta\phi}{r(\phi - \Delta\phi) \cdot \cos\Delta\phi - r(\phi)} \quad (3)$$

The geometrical relationships for radial slope computation are illustrated in Fig. 3. In our approach obstacles are detected in range image sequences by first computing the radial slope in vertical direction for each pixel. If the slope value of a pixel exceeds a predefined threshold, the pixel is labeled as belonging to

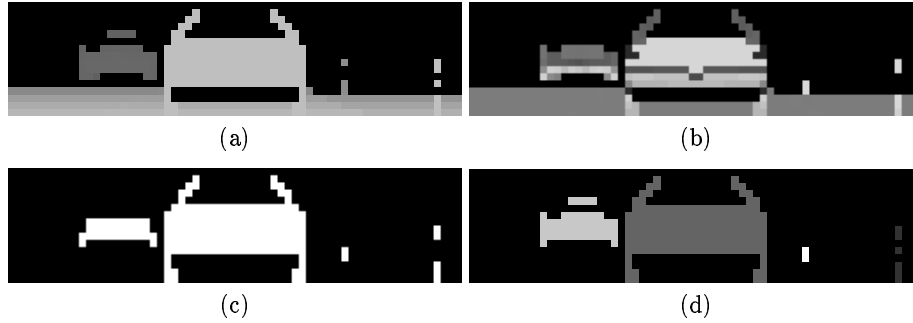


Fig. 4. (a) Range image, (b) radial slope image, (c) thresholded slope image, (d) connected components

an obstacle. Experiments showed that the definition of this threshold, which characterizes non-traversable object areas, is not critical. We have decided to define it as 25 degrees. A region growing step is applied next to determine regions of steep slope. Since obstacles may have flat surface parts in between steep surface parts, the detected regions are taken as start regions for a connected component analysis. The objective of our connected component analysis scheme is to determine point clouds that are connected in 3D space and that have at least one part with steep slope. For that we consider neighborhood in depth and neighborhood in 2D image space as well as height information as features. Thus flat surface parts of an obstacle are merged as well as steep surface parts.

Results for an example scene are shown in Fig. 4. In Fig. 4a the original range image, with range values encoded as greylevels, is shown. The scene contains two cars with one of them driving ahead of the sensor car and the other car driving in the oncoming traffic. On the right side of the road, two posts can be seen. The computed radial slope image is illustrated in Fig. 4b. In this image, slope is encoded as greylevel: A slope of 0° is represented as medium greylevel, and positive (negative) slope as brighter (darker). It can be seen that, depending on the shape of a car, different parts of the same car may have different slope values. The road has a slope value of almost zero degrees. The thresholded slope image is shown in Fig. 4c. As can be seen, pixels belonging to the cars are labeled as obstacles. Also the posts are detected as obstacles. Road pixels are eliminated and thus correctly classified as belonging to a traversable object. The detected connected components are shown in Fig. 4d. Both cars and the posts are correctly detected as obstacles. Due to the fact that slope values can be computed in radial direction, the obstacle detection scheme using radial slope is of low computational complexity. By performing a connected component analysis it is ensured that obstacles are detected in their entire size even if obstacles have flat surface parts in between. Furthermore the connected component analysis in 3D space increases the robustness against noise since simple slope computations are sensitive to noise. However at least two valid depth measurements in neighboring

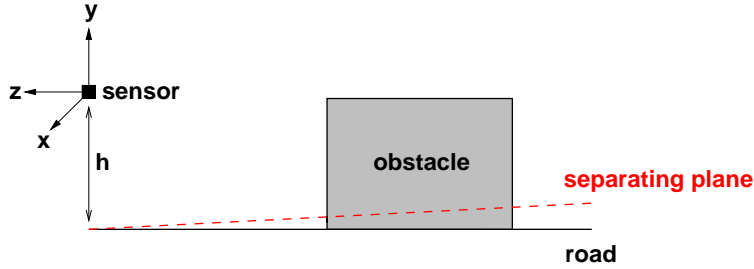


Fig. 5. Separation of image feature

rows are necessary to detect an obstacle. A more detailed description of the radial slope approach for obstacle detection can be found in [17].

Separating Plane In this alternative approach, the knowledge of the geometry of the sensor with respect to the road is used to separate image features into ground and obstacles introducing a separating plane. We make the assumption that the road can be locally modeled as a plane, and determine a depth map of a slightly steeper plane than the current road model (rotated by 0.5°). The center of rotation is placed vertically under the range sensor (see Fig. 5). Upon initialization the road model is assumed to be horizontal and the sensor has a height h . Using an inclined plane has the advantage that noisy ground pixels and small road elevations are less likely erroneously identified as obstacles. It is better to use a rotated instead of a vertically shifted plane as our expected noise depends on the depth. Consequently, near and potentially dangerous obstacles are completely detected whereas objects far away may be lost. Although hardware limitations allow the development of only a simple road model (e.g. limited field of view and range), this is sufficient for our purpose.

Similar systems, also using a road model, have been proposed in the literature. The approaches described in [4] and [10] use the disparity obtained from stereo image pairs to distinguish features painted on the road and obstacles lying on the ground. The lane markings are detected and used to update the road model and the geometric parameters of the stereo cameras. In comparison to these methods we process much smaller images and 3D information for each pixel is directly available. On this basis, all the extracted 3D ground pixels are used to update our simple road model. Given our coarse images, it is not feasible to use a more complex road model like the one described in [6].

We determine a depth map by calculating the distances the sensor would measure if only the separating plane were seen. By comparing the measured depth values (see Fig. 6a) with those from the depth map, we can separate image features into ground and obstacles. Pixels with a smaller distance belong to an obstacle, while larger distances are assigned to the ground (see Fig. 6b,c).

However, the car suspension and the road inclination may alter the geometry of the road model, with small differences potentially producing large errors in

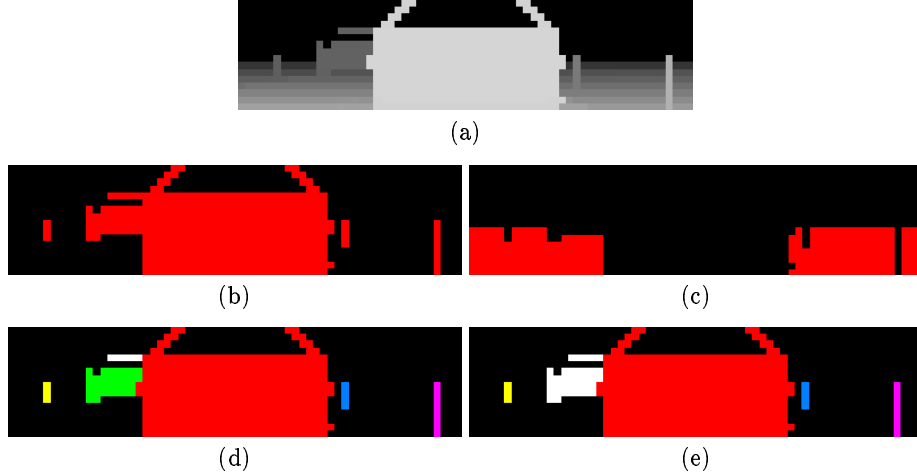


Fig. 6. (a) Range image, (b) obstacle detection, (c) road/ground detection, (d) connected components, (e) connected regions

the separation of the image features. Therefore, the parameters are updated by fitting a plane through all pixels marked as ground. We use the least squares method which minimizes the sum of the squares of the residuals.

As we are dealing with incomplete data, we might not always have enough pixels to get a robust result. Even if we have enough pixels to perform the fit (> 3) it can still be insufficient because of incorrectly classified pixels. Therefore, we calculate the mean distance q of the extracted ground pixels to the fitted plane

$$q = \sqrt{\frac{\sum_{i=1}^N d_i^2}{N}}, \quad (4)$$

where N is the number of defined ground pixels and d_i is the distance of each pixel to the plane. If this measure is higher than a specified threshold, the plane fit is insufficient and the update of the geometric parameters is not reasonable. In this approach the threshold is set to 0.5m.

An update of the current road model is not necessary for each frame. Noise and roughness of the road should not cause an adaption of the geometric parameters. The following two properties are used to decide if an update should be made:

1. the number of ground points
2. the quality of the plane fit.

All distances which are smaller than the depths of the separating plane characterize obstacles which we group into different objects. This is done by connected component analysis (see Fig. 6d). We use spatial coherence to identify

regions from the depth information. First, we are looking for connected components in the 8-neighborhood in the image plane. In the depth dimension a neighboring pixel is connected if the difference in depth is less than a threshold.

As we have to deal with incomplete data and occlusion, it is possible that an obstacle will be split into different disjoint regions. If two regions belong to the same obstacle, they must be close to each other in 3D space. For each pair of regions we determine a probability measure by the *proximity* and *depth similarity*. This measure corresponds to the likelihood that the regions belong to the same obstacle. They are merged if this probability measure is larger than a specified threshold. In Fig. 6e we see the result after combining close regions where the roof and the body of the oncoming car are merged to one object.

Within the separating plane method, the choice of a rotated plane eliminates the problems of noise and uneven roads. The method can also adapt itself to changes in the environment. On the other hand valid ground pixels are required to perform an update of the current road model. A flat angle of incidence or a rough road surface may cause a insufficient reflection of the light. More detailed information about this approach can be found in [12].

3.2 Tracking

Robust tracking schemes for driver assistance and traffic surveillance have to cope with several problems, such as occlusion, new appearance, and disappearance of scene objects. Furthermore, system and measurement noise corrupts the data and thus a tracking mechanism should not rely on perfect data or perfect low level processing.

However, the use of 3D data allows to treat these problems since accurate information about the geometrical arrangement is available. Thus the search space for finding correspondences between consecutive frames can be restricted to a small depth slice. In this paper two tracking schemes are reported. The first is a matching scheme using temporally local and global processes. Secondly Kalman filtering is described. It finds the most likely estimates of the states of obstacles. Both approaches results in a list of tracked obstacles.

Due to the low resolution of our data, the tracking schemes have to deal with the problem that narrow obstacles, e.g. posts and trunks, are only visible at times. Such obstacles can be tracked by means of aging. That means each obstacle is attached an age as attribute. Age is initialized by zero and reset to zero if a correspondence is found. If an obstacle disappears, its age is increased by one and its parameters are updated with estimated values which can be computed by the Kalman filter or by a weighted regression method. The tracking process continues if the obstacle reappears within a certain time interval. Otherwise, if its age has exceeded a certain threshold, it will be removed from the list of tracked obstacles.

Results of both tracking schemes show that obstacles can be tracked even under difficult conditions.

Matching features A successful way of determining the position of an obstacle at the next frame is to find feature correspondences in consecutive frames. In this procedure the use of 3D data is an enormous advantage since the search for correspondences can be restricted to a small depth slice. To allow robust long-term tracking of obstacles, knowledge over larger time intervals has to be acquired as well. Thus our tracking scheme employs temporal global processes, such as the prediction of 3D positions, aging, and global merging of obstacles.

A region-based matching approach is also presented in [1]. At each point of time height maps are computed and segmented into unexplored, occluded, traversable or obstacle regions. Then height maps of consecutive frames are matched in order to estimate the vehicle motion and to find moving objects. Also in [13] feature correspondences are determined to track independently moving and deformable objects. The correspondence problem is solved locally by assigning individual trackers to each feature and searching for the correspondence in a small region-of-interest around the feature's next predicted position.

In our approach correspondences between obstacle hypotheses H_i^{t-1} , $i = 1, \dots, m$ that were stored at time $t - 1$ and connected components C_j^t , $j = 1, \dots, n$ that were detected as obstacles at time t , are determined. To simplify the correspondence problem, we make use of the maximal possible displacement (Fig. 7a), of similarity in object parameters (Fig. 7b), of continuity in motion (Fig. 7c) and consistency of matching (Fig. 7d). In the first step, connected components that are located in the relevant depth slice and image space are determined for each hypothesis H_i^{t-1} . The boundaries of the relevant depth slice can be determined based on the maximum speed of cars and the frame rate. Assuming a maximum speed of 200 km/h and a frame rate of 25 frames/sec, a displacement of 2.22 meters for vehicles driving ahead and a displacement of 4.44 meters for oncoming vehicles is possible. Thus the relevant depth slice can be restricted to $[-4.44 \text{ meters}, 2.22 \text{ meters}]$ surrounding the obstacle hypothesis.

The definition of the relevant image space depends on the sensor geometry and the distance of the hypothesis. The search window is large for close objects and small for distant objects. After all relevant connected components are determined, a grouping step is applied next. Thus groups G_k^t of connected components are built, which probably belong to the same object. This is done to take into account that, due to segmentation errors, noise or poor reflection properties, objects could be split into several parts over time and then are separately detected as connected components. Afterwards all relevant matches $H_i^{t-1} \rightarrow G_k^t$ are assessed. The assessment is defined based on the weighted sum of individual assessments, which describe the distance in horizontal and vertical direction, the similarity with the predicted position, the similarity in object height, the similarity in object width, and the importance of the match based on the volumes of both. The best five matches are stored for each hypothesis H_i^{t-1} .

Because each hypothesis is considered independently up to this step, it may happen that the same connected component is assigned to different hypotheses. To ensure consistent matching, a consistent subset of assignments is selected. Based on this subset, a list of hypotheses for obstacles at time t is stored. Con-

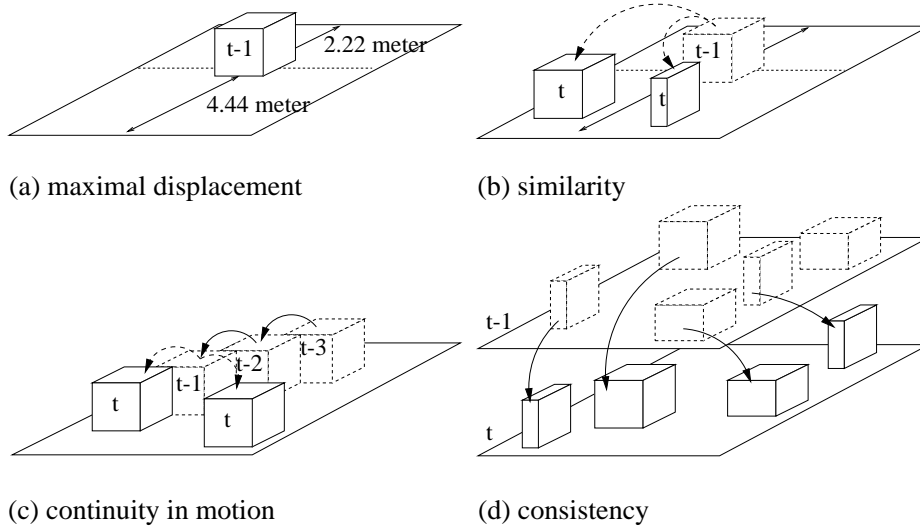


Fig. 7. Constraints for finding feature correspondences

nected components that were detected at time t , but that were not assigned to one of the hypotheses at time $t - 1$, are added to the list as new hypotheses. Temporally global processes increase the robustness of tracking significantly by considering larger time intervals. In our tracking scheme we employ aging, the prediction of 3D positions of obstacles, and global merging of hypotheses as global processes.

As stated above, aging means that hypotheses for which no correspondence was found by local matching are kept for a few frames. Thus at each discrete point of time, it is checked for a hypothesis, if its age has exceeded a maximum threshold. If this is the case, the hypothesis is deleted. An example is shown in Fig. 8. Four consecutive frames are illustrated showing one obstacle in the tracking process. For clarity other detected obstacles are not marked. Focusing on the white bounding box (label 1, Fig. 8b), which represents a car, it can be seen that the bounding box is empty at time $t + 1$ and $t + 2$. This means that no correspondence was found in the corresponding frames. The reason is that the car is driving behind bushes (Fig. 8a) and thus no depth data are measured during this period. But at time $t + 3$ the tracking process recovers and the correct assignment for the car is found.

The 3D position of an object can be predicted based on observations in the past. It is assumed that the object moves in a continuous way during the observed period. This assumption is valid in our case, because cars do not change motion abruptly and the range images are captured with a high frame rate by our sensor. To determine the predicted object position we use weighted regression as prediction procedure; $k = 4$ observations are taken into account for the computation of the predicted value. The observations are not required to occur

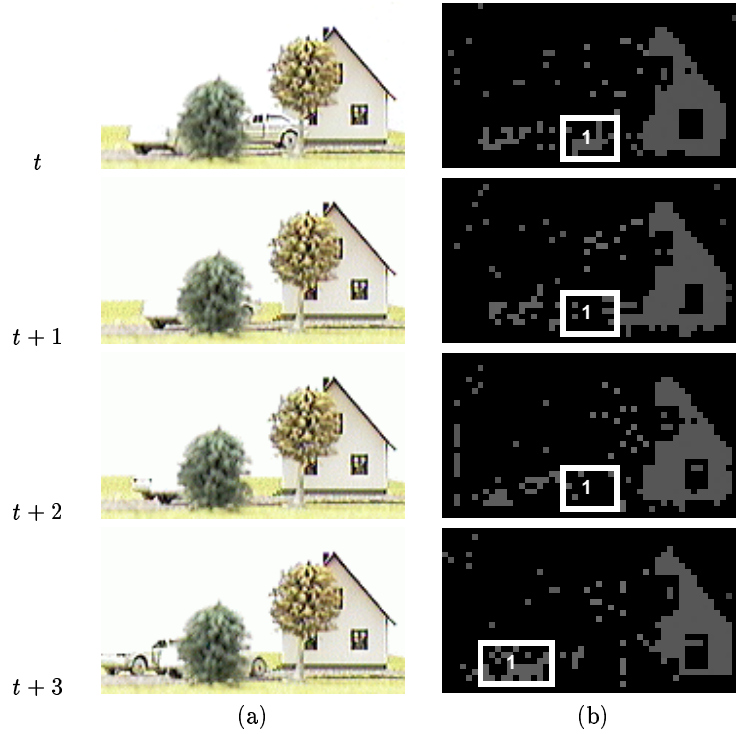


Fig. 8. Example for aging mechanism

in consecutive frames. Due to aging it is possible that measurements are missing in between. Using weighted regression for prediction, observations that were made longer ago are less weighted than observations from the near past.

The global merging process is based on the assumption that hypotheses which move similarly over time and which are adjacent to each other in space probably belong to the same obstacle. Thus at the beginning of a time point it is checked for a hypothesis if adjacent hypotheses have similar motion characteristics. Using range data as input, it is not possible to prevent static objects that are very close to each other from being merged together, because, due to egomotion, static objects will have a similar motion characteristic. But this kind of error is not critical for collision avoidance applications, since obstacle detection and not object recognition is the task to be solved. An example of the global merging of hypotheses is shown in Fig. 9. Due to occlusion by a tree, the house is split into two parts (time t , label 1 and 2). By means of global merging, the hypotheses corresponding to the two house parts are merged into one hypothesis and tracked successfully as one hypothesis along the time axis (time $t + 3$, label 1).

Matching features in consecutive frames is a well-known method for object tracking. To be robust against splitting and merging of objects, multiple matches have to be realized. Unlike the fact that in general the integration of multiple

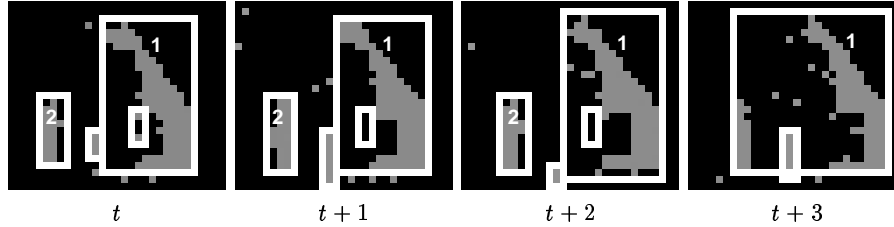


Fig. 9. Example for global merge of obstacle hypotheses

matches increases the computational costs significantly, range imagery allows to restrict the search for correspondences to a small depth slice. Thus only a few possible matches have to be evaluated to select the best match. Furthermore temporally global processes such as aging, prediction and global merge allow robust long-term tracking of obstacles. In addition, it is important to emphasize that scene objects that are located in the same depth slice and move in a similar way may be erroneously merged and tracked as one obstacle. More details about our tracking scheme based on feature correspondences can be found in [16].

Kalman Filtering An optimal estimator is implemented by an algorithm that processes measurements to deduce a minimum error estimate of a system by utilizing knowledge of system and measurement dynamics, assumed statistics of system noise and measurement errors, and initial condition information. The optimal estimator for a quadratic error function is the Kalman filter [7, 9].

For driver assistance and traffic surveillance systems the Kalman filter is an already known approach to deal with system and measurement noise. An extended Kalman filter can be used to perform quantitative tracking [23]. Three different models are constructed to describe the motion of a moving car and are incorporated into a Kalman filter. The authors in [8] propose two methods to combine the estimates of two tracking systems that provide motion parameters: the first method applies a Kalman filter, while the second method assigns weights to the individual estimates based on the covariance matrix.

The model of a linear dynamic process is defined by

$$x_{t+1} = \Phi_t x_t + w_t, \quad w_t \sim N(0, Q_t) \quad (5)$$

where the *transition matrix* Φ_t models the evolution of the state vector x_t at time t and the measurement model

$$z_t = H_t x_t + v_t, \quad v_t \sim N(0, R_t) \quad (6)$$

determines the measurements z_t as a function of the state x_t . H_t is called the *measurement sensitivity matrix*. The system noise w_t and measurement noise v_t are zero-mean Gaussian sequences with given covariance-matrices Q_t and R_t , respectively. These equations, shown in the dashed-line box in Fig. 10, are simply a mathematical abstraction – a model of what we think the system and

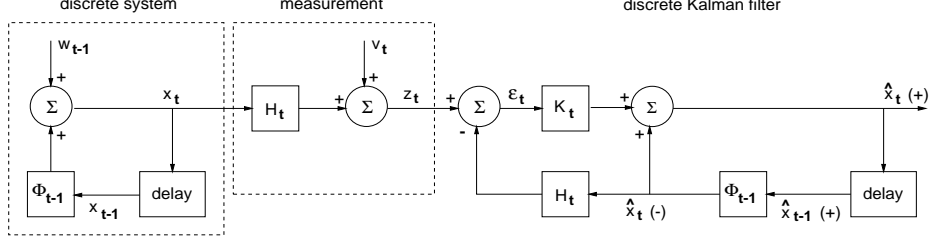


Fig. 10. System model and discrete Kalman Filter

measurement processes are – and the Kalman filter is based upon this model. In the linear, discrete Kalman filter, calculations at the covariance level ultimately serve to provide K_t , which is then used to determine the estimated state \hat{x}_t . K_t is called the *Kalman gain matrix*. The estimated state before the measurement at time t is known as the *a priori* estimate $\hat{x}_t(-)$ and the estimated state after the measurement as the *a posteriori* estimate $\hat{x}_t(+)$. This is illustrated in Fig. 10, which is essentially a simplified flow diagram of the discrete Kalman filter. The basic steps of the computational procedure for the discrete-time Kalman estimator are as follows:

1. a priori state estimate extrapolation:

$$\hat{x}_t(-) = \Phi_{t-1} \hat{x}_{t-1}(+)$$

2. error covariance extrapolation:

$$P_t(-) = \Phi_{t-1} P_{t-1}(+) \Phi_{t-1}^T + Q_{t-1}$$

3. Kalman gain matrix :

$$K_t = P_t(-) H_t^T [H_t P_t(-) H_t^T + R_t]^{-1}$$

4. error covariance update:

$$P_t(+) = [I - K_t H_t] P_t(-)$$

5. a posteriori state estimate observational update:

$$\hat{x}_t(+) = \hat{x}_t(-) + K_t [z_t - H_t \hat{x}_t(-)]$$

We employ Kalman filtering to provide most likely estimates of the state of each obstacle from measurement data corrupted by noise. In our system the state vector of each obstacle contains the depth d_t , the relative velocity \dot{d}_t , the horizontal angle ψ_t to the obstacle and its change $\dot{\psi}_t$, its vertical angle η_t and the corresponding change $\dot{\eta}_t$, where all values are taken at time t :

$$x_t^T = [d_t \ \dot{d}_t \ \psi_t \ \dot{\psi}_t \ \eta_t \ \dot{\eta}_t]. \quad (7)$$

The a priori state predictions are mapped with the measurements (depth, horizontal and vertical angle to an obstacle) from the segmentation step. We search for the detected object whose position is closest to the a priori estimate from the Kalman filter and whose distance is inside a search area.

Obstacles can disappear in some frames due to the coarseness of the data. This problem is addressed by employing an aging process as described before.

Parameters are updated with the estimated values from the Kalman filter over several frames. The obstacle can thus be recognized if it reappears within an certain time. Otherwise it will be removed. Furthermore, new obstacles are added if not all detected objects of the segmentation could be matched.

As extension a global merging process — as described before — can be added. Adjacent obstacles which move similarly over time are probably part of the same object and can therefore be merged.

Kalman filtering makes the prediction of important state information more robust by employing noise models. It provides the best possible estimation under the assumption of Gaussian noise and a quadratic error function. Regrettably, the initialization of a Kalman filter is difficult and computational cost is not small.

3.3 Results

Since the range sensor is still under development, range image sequences of real traffic scenes are not yet available. Therefore, we use on the one hand simulated range image sequences and on the other hand scaled range images recorded by an ABW range scanner (ABW GmbH, Germany) to develop and test the segmentation and target tracking algorithms.

First, the simulated range images are generated by a graphical simulation software package. An example is illustrated in Fig. 11. From left to right, we see the simulated traffic sequence and the acquired range images. The range data are represented by different greylevels; black represents undefined pixels where no meaningful measurement was obtained. The range sensor has a relatively narrow field of view whereas the driver has a wider field of view. Thus the car driving ahead appears very large in the range images. The scene contains two cars, one driving ahead and the other in the oncoming traffic. During the course of the sequence the oncoming car leaves the sensor's field of view. This 100-frame sequence has a frame rate of 25 images per second, an image size of 16×64 pixels and a field of view of $2.38^\circ \times 10.0^\circ$. In the simulation, the range sensor is installed next to one of the front lights. Secondly, in Fig. 12 an example of a range image sequence of a scaled toy traffic scenario acquired with a structured light sensor is shown. The scene contains one car driving in the oncoming traffic. Several trees are in the background. This sequence has a length of 25 images and a frame rate of 25 images per second. The image size is 65×144 pixels and the field of view is $6.04^\circ \times 14.1^\circ$.

All combinations between the segmentation and tracking approaches are possible. In this paper, however, we focus on the results of two. Results of obstacle detection and tracking for the simulated traffic sequence of Fig. 11 are shown in Fig. 13. In Fig. 13a obstacles are detected using the radial slope approach. Tracking is performed by finding feature correspondences. Both cars (label 1 and 2) are correctly detected as obstacles and tracked successfully along the time axis. In Fig. 13b we separate image features in the segmentation approach and use a Kalman filter for the tracking. The detected obstacles are represented by their bounding boxes, thus the state vector has to be augmented by the

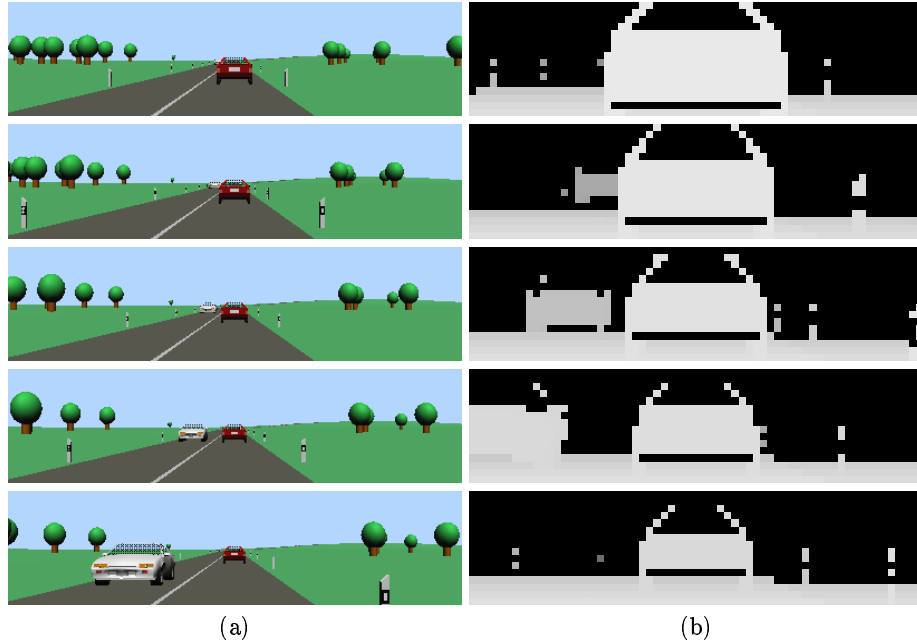


Fig. 11. (a) Simulated traffic sequence and (b) corresponding computed range images, Sequence 22

width and height of each obstacle. The size and location of the bounding boxes are the estimated state values provided by the Kalman filter. The results show only obstacles which are new or tracked within this segmentation step, updated obstacles are not included. As the results show, the separation of image features in combination with a Kalman filter works well. The oncoming bright car (label 2) which is partially occluded is correctly detected and tracked although the position and size change considerably. By comparing the results in Fig. 13 we can observe differences between the two approaches. As the radial slope approach was used in the left column, single pixels (see for example the fifth image from the top) can not be identified as an obstacle. This segmentation scheme needs at least two valid depth measurements in neighboring rows to detect an obstacle. In contrast we see that the separation of image features which is shown in the right column loses image pixels lying below the separation plane.

Examples for tracking a car in a toy traffic scene is illustrated in Fig. 14. The corresponding toy traffic scene is shown in Fig. 12. In Fig. 14a obstacles are once more detected using the radial slope approach and tracked by finding feature correspondences. As can be seen the car (label 1) is successfully detected as obstacle and tracked over the entire range image sequence. The result of separating image features in the segmentation step and use of a Kalman filter for the tracking are shown in Fig. 14b. Even if a tracker loses its obstacle (label 1 \rightarrow 4) the system continues without any difficulties by a new initialization of

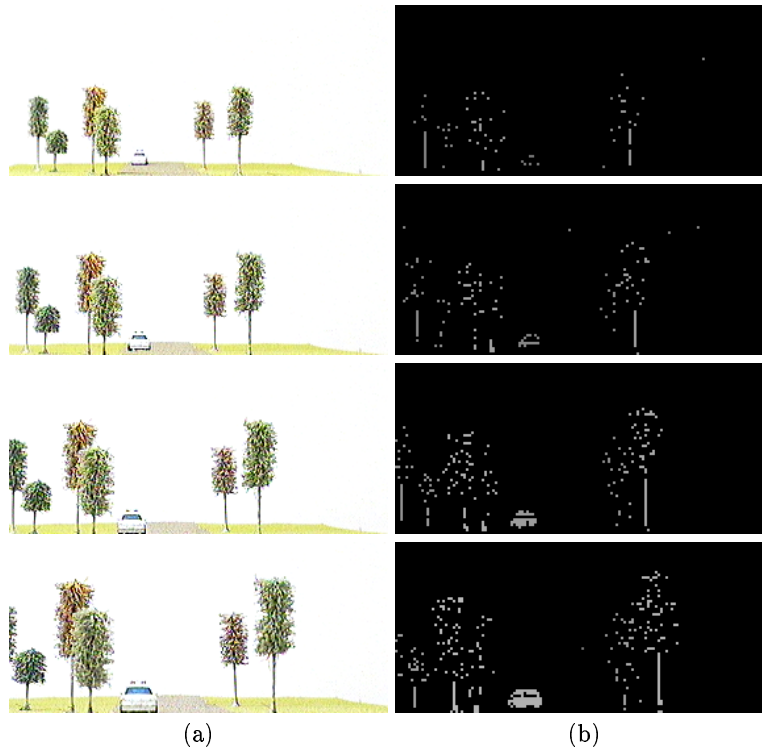


Fig. 12. (a) Sensor’s view at a toy traffic sequence and (b) corresponding range images recorded with a structured light sensor, Sequence 10

the Kalman filter. In this sequence, only obstacles larger than three pixels are shown. Looking at the results in Fig. 14 the dissimilar object sizes are conspicuous. As already explained for the previous result sequence, this is the effect of the different obstacle detection schemes and the various region groupings. Furthermore, differences between the two tracking methods are illustrated. While the matching scheme has problems to track trees in their entire size, Kalman filtering loses the moving car.

4 Conclusion

In this article several approaches for obstacle detection and tracking in low resolution range image sequences were proposed. We focused on an automotive application. The availability of accurate 3D data is found to be advantageous.

Concerning collision avoidance in traffic scenarios, it was shown that robust obstacle detection is possible by checking traversability in the sensor’s field of view. Two approaches were presented based on vertical slope evaluation and computation of a separating plane. Robust long-term tracking of obstacles implies

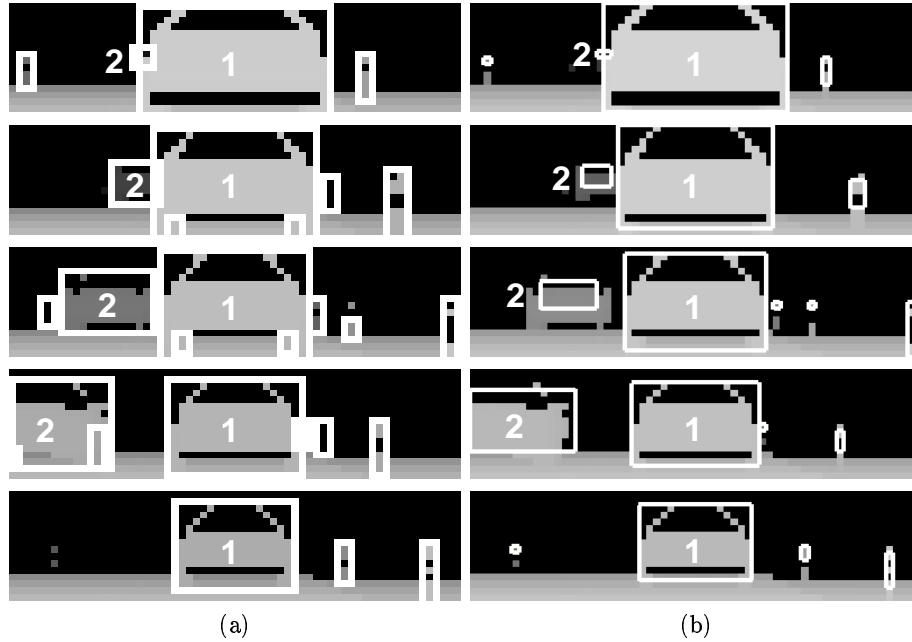


Fig. 13. (a) results of the matching scheme, (b) result of the Kalman filtering

the handling of problems such as occlusion, new appearance and disappearance of scene objects, and furthermore, due to the low resolution of our data, the visibility of narrow obstacles. Our findings reveal that the use of 3D data allows us to treat these problems since information about the geometrical arrangement is available. Thus the search for correspondences can be restricted to a small depth slice and in case of temporal disappearance of obstacles, internal parameters are updated by estimated values. As obstacle detection is performed at every discrete point of time, newly appearing obstacles and even obstacles that were partly occluded previously, are detected and tracked in their entirety. Our investigations focused on two tracking schemes. In a first approach a matching scheme was realized using temporally local and global processes. In a second approach Kalman filtering was employed to track obstacles.

The described methods allow real time processing. This is possible due to the small image size, the minimization of the object management and the restriction of the search space. The issue of how to combine the different approaches in order to improve robustness of an overall system, deserves further study and will be subject of our future work.

Our results imply that range imagery is a key technology for many computer vision applications since many ambiguities of interpretation arising between object boundaries and inhomogeneities of intensity, texture and color can thus be trivially solved.

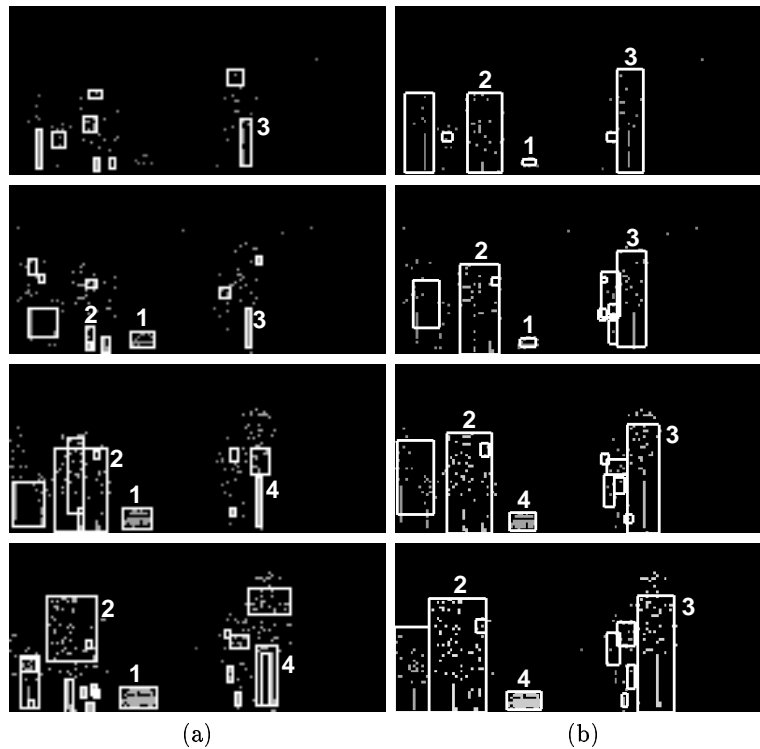


Fig. 14. (a) results of the matching scheme, (b) result of the Kalman filtering

Acknowledgments

This research was partially supported by MINORA, a project of the Swiss priority program OPTIQUE II, funded by the ETH Council.

References

1. M. Asada, M. Kimura, Y. Taniguchi, and Y. Shirai. Dynamic integration of height maps into 3D world representation from range image sequences. *International Journal of Computer Vision*, 9(1):31–53, 1992.
2. J. Bernasch and W. von Seelen. Visual attention control in complex dynamic image sequences. In *9th Israeli Symposium on Artificial Intelligence and Computer Vision*, Ramat Gan, Israel, 28-29 December 1992.
3. M. Bertozzi, A. Broggi, D. Colla, and A. Fascioli. Sensing of automotive environments using stereo vision. In *30th International Symposium on Automotive Technology and Automation (ISATA), Special Session on Machine Vision and Intelligent Vehicles and Autonomous Robots*, pages 187–193, Florence, Italy, 16th-19th June 1997.
4. M. Bertozzi, A. Broggi, G. Conte, and A. Fasciol. Obstacle and lane detection on argo. In *IEEE Conference on Intelligent Transportation Systems*, pages 1010–1015, 1997.

5. M. J. Daily, J. G. Harris, and K. Reiser. Detecting obstacles in range imagery. In *Image Understanding Workshop*, pages 87–97, February 1987.
6. E. D. Dickmanns and B. D. Mysliwetz. Recursive 3-d road and relative ego-state recognition. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 14(2):199–213, 1992.
7. A. Gelb. *Applied Optimal Estimation*. MIT Press, 1996.
8. S. Gil, R. Minanese, and T. Pun. Combining multiple motion estimates for vehicle tracking. In *European Conference on Computer Vision*, volume 2, pages 307–320, 1996.
9. M. S. Grewal and A. P. Andrews. *Kalman Filtering Theory and Practice*. Prentice-Hall, Inc., 1993.
10. Q.-T. Luong, J. Weber, D. Koller, and J. Malik. An integrated stereo-based approach to automatic vehicle guidance. In *5th International Conference on Computer Vision*, pages 52–57, 1995.
11. E. B. Meier and F. Ade. Tracking cars in range image sequences. In *IEEE Conference on Intelligent Transportation Systems (ITS)*, Boston, Massachusetts, 9-12 November 1997.
12. 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*, 1998.
13. J. Roberts and D. Charnley. Attentive visual tracking. In *British Machine Vision Conference*, pages 459–468, 1993.
14. S. R. Ruocco. *Robot Sensors and Transducers*. Wiley, New York, 1990.
15. R. Schneider, G. Wanielik, and J. Wenger. Millimeter-wave imaging of traffic scenarios. In *Proceedings of the 1996 IEEE Intelligent Vehicles Symposium*, pages 327–332, September 19-20 1996.
16. K. Sobottka and H. Bunke. Employing range imagery for vision-based driver assistance. In *SPIE conference: Enhanced and Synthetic Vision*, Orlando, Florida, USA, 13-17 April 1998.
17. K. Sobottka and H. Bunke. Obstacle detection in range image sequences using radial slope. In *3rd IFAC Symposium on Intelligent Autonomous Vehicles*, pages 535–540, Madrid, Spain, 25-27 March 1998.
18. T. Spirig, M. Marle, and P. Seitz. The multi-tap lock-in CCD with offset subtraction. *IEEE Transactions on Electron Devices*, 44(10):1643–1647, 1997.
19. 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, 1995.
20. C. Thorpe, M. Hebert, T. Kanade, and S. Shafer. The new generation system for the CMU Navlab. In I. Masaki, editor, *Vision-based vehicle guidance*, pages 30–82, 1992.
21. S. Tokoro. Automotive application systems of a millimeter-wave radar. In *Proceedings of the 1996 IEEE Intelligent Vehicles Symposium*, pages 260–265, September 19-20 1996.
22. P. A. Veatch and L. S. Davis. Efficient algorithms for obstacle detection using range data. In *Computer Vision, Graphics, and Image Processing 50*, pages 50–74, 1990.
23. 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.