Fusion of Telemetric and Visual Data from Road Scenes with a Lexus Experimental Platform

Igor Paromtchik, Mathias Perrollaz, Christian Laugier

Fusion of Telemetric and Visual Data from Road Scenes with a Lexus Experimental Platform

Igor E. Paromtchik, Mathias Perrollaz, Christian Laugier

***Abstract*— Fusion of telemetric and visual data from traffic scenes helps exploit synergies between different on-board sensors, which monitor the environment around the ego-vehicle. This paper outlines our approach to sensor data fusion, detection and tracking of objects in a dynamic environment. The approach uses a Bayesian Occupancy Filter to obtain a spatio-temporal grid representation of the traffic scene. We have implemented the approach on our experimental platform on a Lexus car. The data is obtained in traffic scenes typical of urban driving, with multiple road participants. The data fusion results in a model of the dynamic environment of the ego-vehicle. The model serves for the subsequent analysis and interpretation of the traffic scene to enable collision risk estimation for improving the safety of driving.**

I. INTRODUCTION

Sensor fusion has been used successfully in automotive applications [1], [2], [3], [4]. This paper focuses on data fusion from telemetric sensors (lidars) and stereo-vision by means of the Bayesian Occupancy Filter (BOF) [5], [6]. The environment is represented by a grid [7], [8], and the BOF provides to assign probabilities of *cell occupancy* and *cell velocity* for each cell in the grid. The preprocessing of stereo images results in a disparity map. The probabilistic models of a lidar and a stereo camera are used.

The data fusion is performed in the BOF with the probabilistic grids computed from the real data from the lidars and stereo-vision. The clustering and tracking algorithm identifies individual objects in the scene in front of the ego-vehicle [9]. The data fusion, detection and tracking are required for estimation and prediction of collision risk for the ego-vehicle [10] and are integrated in our conceptual framework for analysis of dynamic scenes [11].

II. BAYESIAN SENSOR FUSION

The Bayesian Occupancy Filter (BOF) is used for data fusion from the lidars and stereo-vision. The BOF operates with a four-dimensional grid representing the environment. Each cell of the grid contains a probability distribution of the cell occupancy and a probability distribution of the cell velocity. The probabilistic models of a lidar and a stereo camera are developed, in order to compute occupancy grids, which are used as observations for the BOF.

A. Sensor Models

The lidar model is beam-based [8]. It includes four layers of beams and assumes each beam to be independent. We build a probabilistic model for each beam layer independently. The stereo camera is assumed in a “rectified” geometrical configuration, that allows us to compute a disparity map, which is equivalent to a partial three-dimensional representation of the scene. The disparity map computation is based on the double correlation method [12], which has two major advantages: a better matching over the road surface and an instant separation between “road” and “obstacle” pixels, without using any arbitrary threshold. The computation of the occupancy grid is directly performed in the disparity space associated with the disparity map, thus, preserving the intrinsic precision of the stereo camera.

The partially occluded areas of the scene are monitored by means of our visibility estimation approach. Consider a cell c in the u-disparity plane. Let $P(C_c)$ denote the confidence of c being occupied, $P(V_c)$ be the probability of c being visible, and $P(R_c)$ be the confidence of c containing the road surface. The occupancy probability of cell c is

$$\begin{aligned} P(O_c) = \quad & [\, P(V_c) \cdot P(C_c) \cdot (1 - P_{fp}) \\ & + P(V_c) \cdot (1 - P(C_c)) \cdot P_{fn} \\ & + (1 - P(V_c)) \cdot 0.5\,] \cdot (1 - P(R_c)), \end{aligned} \tag{1}$$

where P_{fp} and P_{fn} are the false positive and false negative probabilities of the stereo matching algorithm. Then, the u-disparity occupancy grid is transformed into a Cartesian grid for its use in the BOF. This probabilistic model of the stereo camera is described in detail in [13].

B. Fusion and Filtering

At each time step, the probabilities of cell occupancy and cell velocity are estimated by means of Bayesian inference with the BOF [5], [6]. This is a recursive algorithm containing two steps: prediction and estimation (correction). The prediction computes the *a priori* distribution, and the estimation uses the prediction result and the current observations from the sensors to compute the *a posteriori* distribution.

Let $Z_i = Z_i^t$ denote an observation from a sensor i at time t, and $\mathbf{Z} = [Z_1 \cdots Z_S]$ be a set of observations from S sensors. Let $P(O_c\, A_c)$ denote the *a priori* probability for a cell c, where $P(O_c)$ is the occupancy probability and $P(A_c)$ is the antecedent (velocity) probability. In this context, the prediction step propagates the probability distributions of cell occupancy and cell velocity of each cell and obtains the prediction $P(O_c\, A_c)$. Let $P(O_c\, A_c \,|\, \mathbf{Z})$ denote the *a posteriori* probability obtained according to the observations. In the estimation (correction) step, $P(O_c\, A_c \,|\, \mathbf{Z})$ is updated by taking into account the observations $\mathbf{Z}$ yielded by the S sensors.

The authors are with the National Institute for Computer Science and Control, INRIA Grenoble Rhône-Alpes, 38334 Saint Ismier Cedex, France igor.paromtchik@inrialpes.fr

At the input of the filter, the occupancy grids provided by the sensors are merged according to the following equation:

$$P(\mathbf{Z}\,|\,O_c\,A_c) = \prod_{i=1}^{S} P(Z_i\,|\,O_c\,A_c), \tag{2}$$

and the *a posteriori* probability estimate is obtained as

$$P(O_c\,A_c\,|\,\mathbf{Z}) = \frac{P(O_c\,A_c)\cdot P(\mathbf{Z}\,|\,O_c\,A_c)}{P(\mathbf{Z})}, \tag{3}$$

where $P(\mathbf{Z})$ is a uniform probability distribution. The probability of cell occupancy $P(O_c\,|\,\mathbf{Z})$ and the probability of cell velocity $P(A_c\,|\,\mathbf{Z})$ are computed by marginalization and are used for the next prediction step. Note that the prediction step assumes a constant velocity of objects, and an internal parameter of the BOF serves to take into account the corresponding prediction error, when a constant velocity assumption does not hold.

III. FAST CLUSTERING AND TRACKING

Our Fast Clustering and Tracking (FCT) algorithm serves to retrieve an object level representation from the estimated grids and to track the objects' trajectories [9]. It operates at an object representation level and contains three modules: a clustering module, a data association module, and a tracking and tracks management module.

The clustering module combines the probabilities of the cell occupancy/velocity estimated by the BOF with the prediction for each object being tracked by the tracker, i.e. a region of interest (ROI). We then try to extract a cluster in each ROI and associate it with the corresponding object. There could be a variety of cluster extracting algorithms, however, we have found that a simple neighborhood-based algorithm provides satisfactory results: the eight-neighbor cells are connected according to an occupancy threshold and the velocity distribution is employed to distinguish the objects that are close to each other but move at different velocities. The output of this module leads to three possible cases, as shown in Fig. 1: (i) no object is observed in the ROI, (ii) unambiguous observation with one and only one cluster extracted and implicitly associated with the given object, and (iii) ambiguous observation, where the extracted cluster is associated with multiple objects.

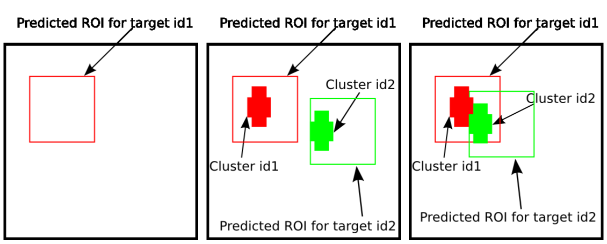

Fig. 1. The possible cases of clustering result: no object observed, unambiguos observation, and ambiguous observation

The data association module aims to solve the problem of ambiguous observation (multiple tracked objects, overlapped ROIs) in the clustering module. Assume there are N objects associated with a single cluster, where N is a number we know exactly. The cause of the ambiguity is twofold: (i) numerous objects are very close to each other and the observed cluster is the union of observations generated by N different objects, and (ii) N different objects correspond to a single real object and the observations must be merged into one.

We employ a re-clustering strategy to deal with the first situation and a cluster merging strategy for the second one. The re-clustering aims to divide the cluster into N sub-clusters and associate them with the N objects, respectively. Because the number N is known from the prediction step, a K-means algorithm is applied [14].

The cluster merging is based on a probabilistic approach. Whenever an ambiguous association F_{ij} between two tracks T_i and T_j is observed, a random variable S_{ij} is updated to indicate the probability of T_i and T_j being parts of a single object. The probability values $P(F_{ij} \mid S_{ij})$ and $P(F_{ij} \mid \neg S_{ij})$ are the algorithm parameters which are constant with regard to i and j. Similarly, the probability $P(S_{ij}\,|\neg F_{ij})$ is updated when no ambiguity between T_i and T_j is observed. Then, by thresholding the probability $P\left(S_{ij}\right)$, the decision of merging the tracks T_i and T_j can be made by calculating the Mahalanobis distance between them. Now we arrive at a set of clusters which are associated with the objects being tracked without ambiguity. Then, the tracking and tracks management module uses a general tracks management algorithm to create and delete the tracks, and use a Kalman filter to update their states [15].

IV. EXPERIMENTAL RESULTS

A. *The Lexus Platform*

Our experimental plaform is built on a Lexus LS600h car, shown in Fig. 2. The car is equipped with a TYZX stereo camera [16] situated behind the windshield, two IBEO Lux lidars [17] placed inside the frontal bumper, and an Xsens IMU combined with GPS [18]. The on-board DELL computer with an NVidia graphics processing unit (GPU) is used for collecting and processing of the sensor data and the risk assessment. The visual and telemetric data are used concurrently for a preliminary qualitative evaluation.

Fig. 2. Our experimental platform on a Lexus car, with a TYZX stereo camera behind the windshield and two IBEO Lux lidars inside the frontal bumper

The TYZX stereo camera has a baseline of 22 cm, a resolution of 512x320 pixels, and a focal length of 410 pixels. The IBEO Lux lidar provides four layers of up to 200 impacts at a sampling period of 20 ms. The maximum lidar detection range is about 200 m, the angular range is 100°, and the angular resolution is 0.5°. We use two lidars to monitor the area in front of the car. The observed region is 40 m in length and 40 m in width, a maximum height is 2 m, and the cell size of the grid is 0.2x0.2 m.

The user interface is based on the Qt library and it provides access to several parameters of the system, e.g. filtering, disparity computation, BOF. The Hugr middleware [19] allows recording and synchronizing of the data from different sensors as well as replay capability.

Note that the data fusion with the BOF requires calibration of the extrinsic parameters of the sensors in the common coordinate system. Thanks to the BOF and a grid resolution with a cell size of 0.2x0.2 m, a slight calibration error has little impact on the final grid after data fusion. The following parameters are set for the occupancy grid computation from stereo-vision: $P_{fp} = 0.01$ and $P_{fn} = 0.05$.

B. Occupancy Grids and Sensor Data Fusion

We discuss our concept on an example of the data obtained with our Lexus platform on urban roads with multiple traffic participants. Fig. 3-a shows an image of such a traffic scene, when approaching a crossroad. The BOF is used to merge the data from the on-board sensors, which monitor the environment: two lidars (Fig. 3-b and Fig. 3-c) and the stereo camera (Fig. 3-d). This results in a grid representation of the local environment in front of the car. The grid is shown in Fig. 3-e, where the black color indicates the occupied areas, the white color corresponds to the unoccupied space, and different levels of the grey intensity represent the occupancy probability of other areas. The occupancy grid in the u-disparity plane, corresponding to the data in Fig. 3-d is shown in Fig. 3-f. The yellow rectangles in Fig. 3-a show the objects, which are correctly detected and tracked: a bus, a bicycle, cars, and the infrastructure elements.

One of the advantages of using the BOF for a grid representation in comparison with the static grid-based approaches is the estimation of velocities of cells in the BOF. Since the velocity estimation is taken into account in the clustering stage, it results in distinguishing between two objects, which move close to each other at different velocities, e.g. a bicycle and a car in the left half of Fig. 4-a are separated correctly into two different clusters.

A limitation of our current implementation is concerned with a constant velocity assumption, that does not hold during a sharp turn. This assumption can lead to over-segmentation of objects, e.g. the cells corresponding to the front of the car in Fig. 4 have an estimated velocity which differs from that of the rear of the car. Nevertheless, a solution is to increase the frequency of data processing, e.g. by means of implementing the BOF in hardware as a system-on-chip (SOC), or to estimate the motion of the ego-vehicle by means of its proprioceptive sensors.

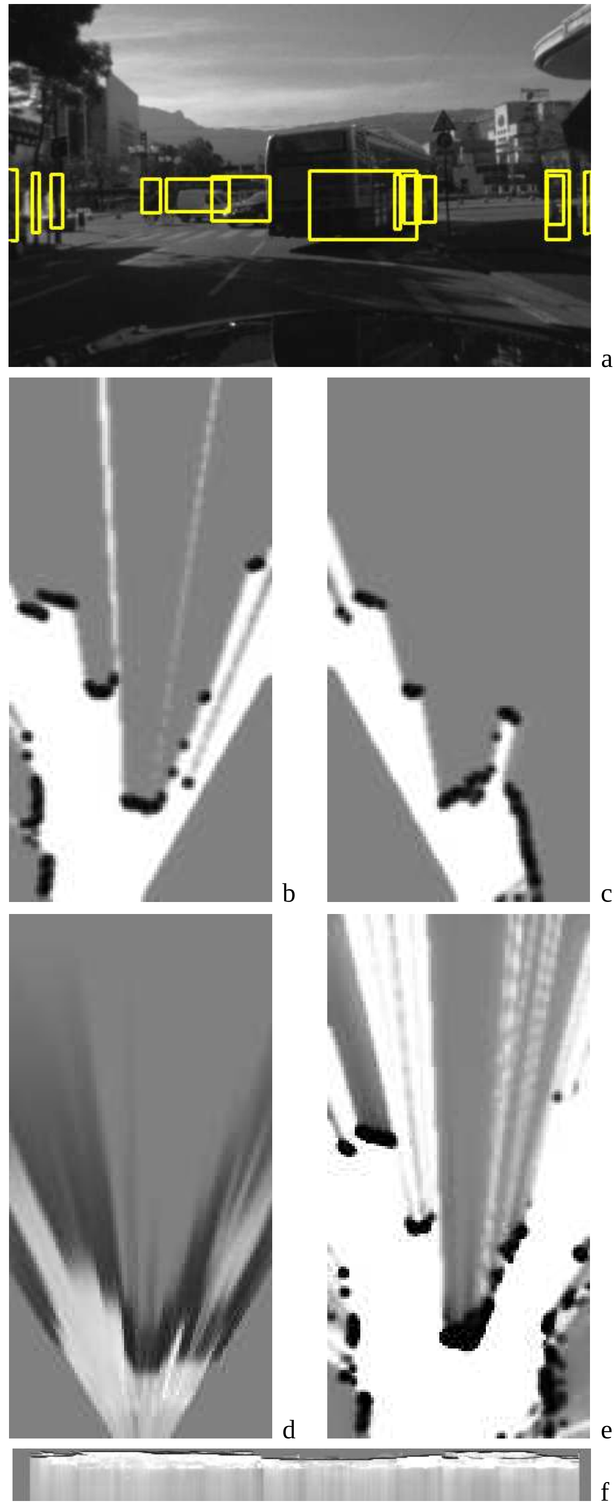

Fig. 3. Approaching a crossroad: (a) a traffic scene image, where the rectangles indicate the detected and tracked objects, (b) a grid representation from the left lidar (lower scanning layer), (c) a grid representation from the right lidar (lower scanning layer), (d) a grid representation from stereo-vision, (e) a grid representation after data fusion, (f) an occupancy grid in the u-disparity plane

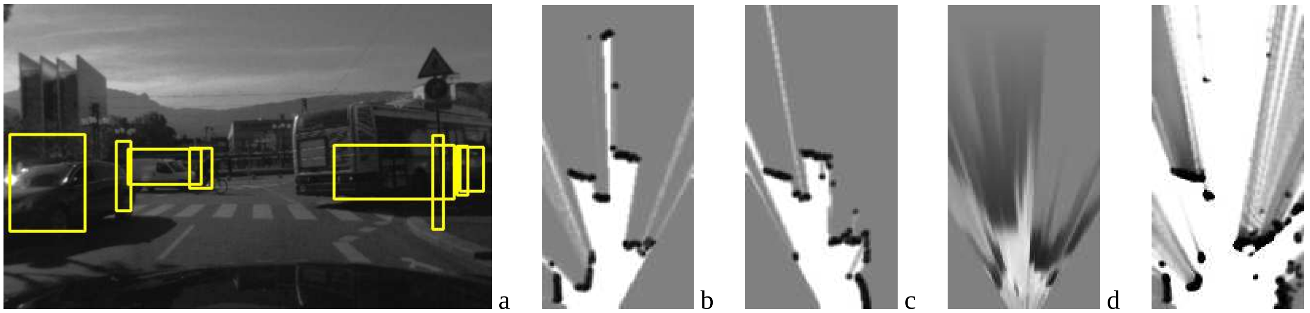

Fig. 4. Entering a crossroad: (a) a traffic scene image, where the rectangles indicate the detected and tracked objects, (b) a grid representation from a left lidar, (c) a grid representation from a right lidar, (d) a grid representation from stereo-vision, (e) a grid representation after data fusion

Fig. 5 gives an example of telemetric data obtained with the two on-board lidars, where the laser impacts are plotted onto the camera images (red dots correspond to the left lidar, and the green dots correspond to the right one). There are four scanning layers in the vertical direction for each lidar. The laser impacts with the road are filtered out thanks to the fusion of the multiple layers, as seen in Fig. 6-e. The lidars have overlapping viewfields, that provides to detect correctly the distant objects, e.g. two pedestrians in Fig. 6.

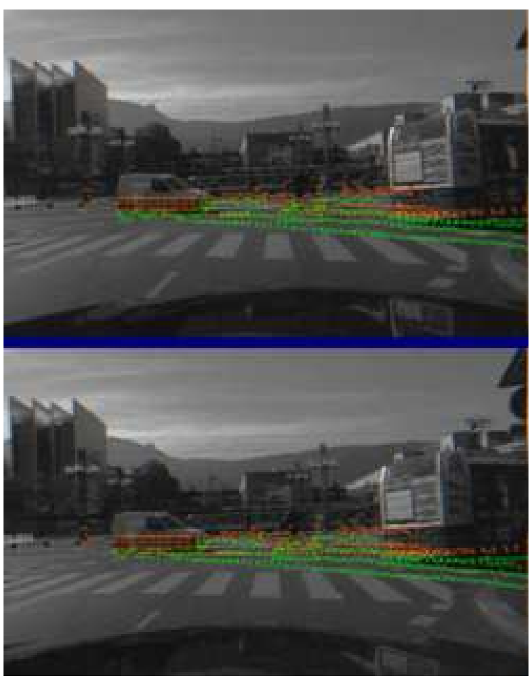

Fig. 5. An example of the multi-layer telemetric data represented by laser impacts (colored dots) from the two on-board lidars

Note that the height of the rectangles in Fig. 3-a and Fig. 4-a is set empirically to 1.8 m for the visualization purpose. The constant height can become a problem to visualize tall objects, e.g. a bus in the scene, or in the case of small objects. The width of rectangles equals twice the lateral standard deviation σ_{xx} of the objects positions obtained from the FCT algorithm. This provides a correct visualization of the width of frontal objects, while it is not currently adapted to visualize non-frontal objects, e.g. the bus in Fig. 7-a. A motorcycle and a bicycle behind the bus are correctly detected and separated because of the velocity estimation, as seen in Fig. 7-e.

Various objects are present in the traffic scene in Fig. 8, where the bus is detected and is separated into two objects because the lidars' data is affected by laser impacts with the rear wheels of the bus, and the stereo-vision does not provide sufficient accuracy at such a large distance. Note that the accuracy of lidars remains constant over the distance, while the accuracy of stereo-vision becomes poor at long range (i.e. telemetric data is given more confidence relative to the visual information in this case). One can observe that two pedestrians, crossing the street in Fig. 8, are detected as a single object because they walk together at the same speed.

Fig. 9 shows another advantage of data fusion, that is due to a broad viewfield provided by the two lidars. While the truck in the right side of the scene is hardly visible for the stereo camera, it is still detected from the lidars data, as seen in the grid representation after data fusion in Fig. 9-e.

The above results also show that the effect of stereo-vision is significantly lower than that of the lidars on the resulting occupancy grid. This is due to a perception range constraint because of a small baseline of the stereo camera. Nevertheless, the stereo-vision remains valuable because of its potential for objects recognition, classification, and visual tracking. The accuracy of stereo-vision is sufficiently high at distances upto 10 m to enable detection of objects. Additionally, stereo-vision is an inexpensive alternative to multi-layer lidars for production cars.

C. Computation time

Two critical stages of the sensor fusion have been implemented on GPU: the BOF and the stereo image processing, including matching and occupancy grid computation. In comparison to the high computational cost of the BOF, the cost of the FCT algorithm can be neglected [6], [9]. The BOF being designed to be highly parallelizable, it runs on GPU NVidia GeForce GTX 480 in 20 ms, without specific optimization. The complete processing chain for a lidar (including the BOF and the FCT algorithm) is capable of running at 20 Hz. The implementation of our stereo image processing on the GPU allows us to perform the matching process in 6 ms and the occupancy grid computation in 0.1 ms.

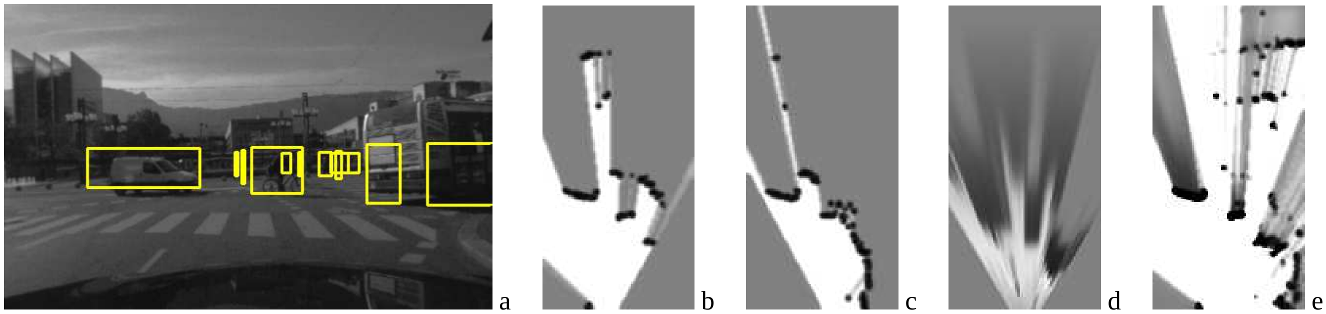

Fig. 6. Advancing at a crossroad: (a) a traffic scene image, where the rectangles indicate the detected and tracked objects, (b) a grid representation from a left lidar (lower scanning layer), (c) a grid representation from a right lidar (lower scanning layer), (d) a grid representation from stereo-vision, (e) a grid representation after data fusion

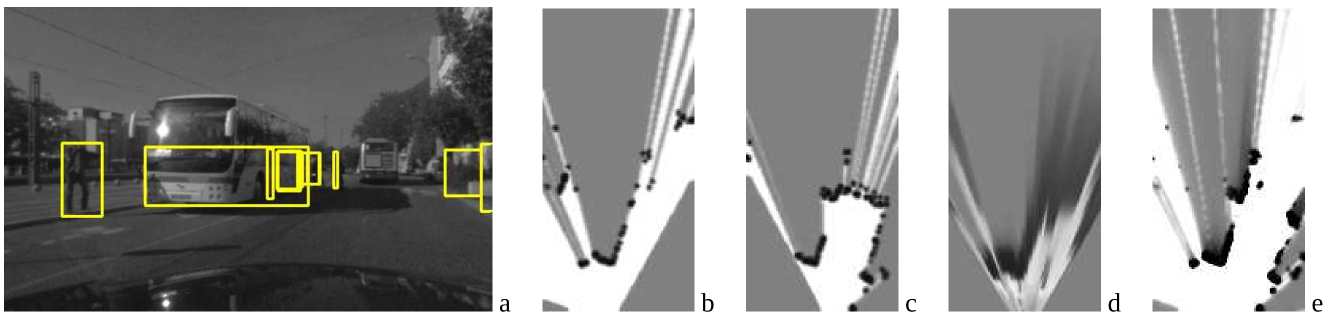

Fig. 7. Leaving a crossroad: (a) a traffic scene image, where the rectangles indicate the detected and tracked objects, (b) a grid representation from a left lidar (lower scanning layer), (c) a grid representation from a right lidar (lower scanning layer), (d) a grid representation from stereo-vision, (e) a grid representation after data fusion

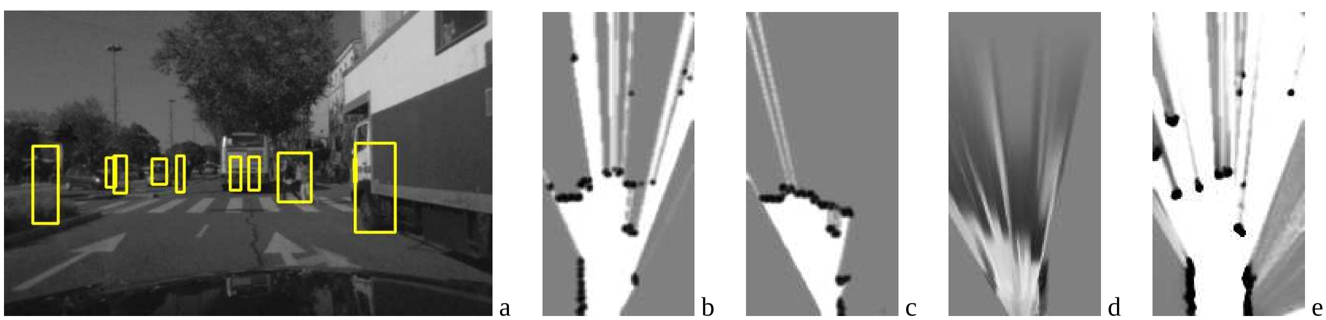

Fig. 8. Moving on a straight road: (a) a traffic scene image, where the rectangles indicate the detected and tracked objects, (b) a grid representation from a left lidar (lower scanning layer), (c) a grid representation from a right lidar (lower scanning layer), (d) a grid representation from stereo-vision, (e) a grid representation after data fusion

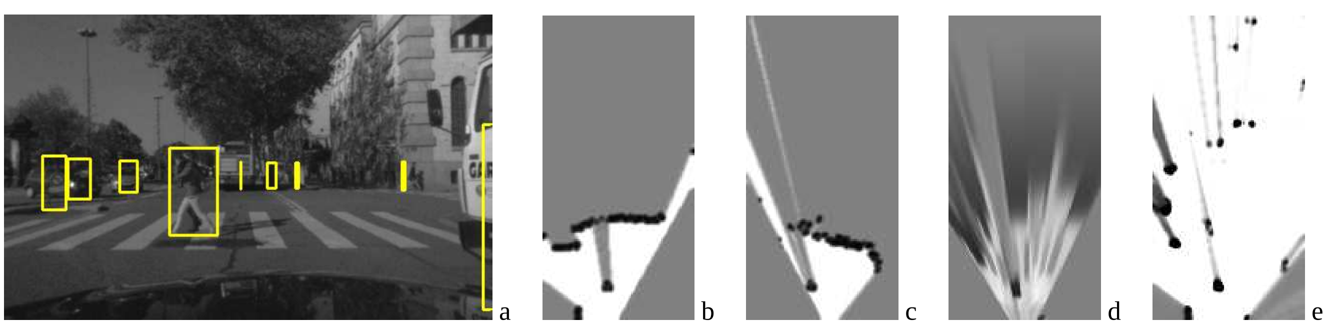

Fig. 9. Waiting at a pedestrian crossing: (a) a traffic scene image, where the rectangles indicate the detected and tracked objects, (b) a grid representation from a left lidar (lower scanning layer), (c) a grid representation from a right lidar (lower scanning layer), (d) a grid representation from stereo-vision, (e) a grid representation after data fusion

V. Conclusion

We discussed our approach to sensor fusion of telemetric and visual data with the BOF for a grid representation of the traffic environment for the ego-vehicle. The approach was implemented and tested on our experimental platform on a Lexus car. The experiments were conducted in scenarios typical of urban driving, with multiple road participants. The examples of data fusion were discussed to explain the advantages and indicate potential pitfalls. The experimental results proved the feasibility and relevance of our approach. The probabilistic approach to sensor fusion and environment modeling is part of our conceptual framework, which serves to estimate and predict collision risks for the ego-vehicle. The experimental platform will be used to create a database to allow for benchmarking, quantitative evaluation and comparison of alternative approaches.

VI. Acknowledgments

We thank Toyota Motor Europe for their continuous support of our experimental work on the Lexus car. Our thanks are given to Nicolas Turro, Jean-François Cuniberto, Amaury Nègre and Mao Yong (INRIA) for their technical assistance, as well as to Kamel Mekhnacha (ProBayes) for the ProBT© library and valuable discussions.

References

[1] M. Munz, M. Mählisch, K. Dietmayer. "Generic Centralized Multi Sensor Data Fusion based on Probabilistic Sensor and Environment Models for Driving Assistance Systems", *IEEE Intelligent Transportation Systems Magazine*, Vol. 2(1), 2010.

[2] F. Fayad, V. Cherfaoui. "Object-level Fusion and Confidence Management in a Multi-sensor Pedestrian Tracking System", *Proc. of the IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, Seoul, Korea, 2008.

[3] M. Mählish, R. Hering, W. Ritter, K. Dietmayer. "Heterogenous Fusion of Video, LIDAR, and ESP Data for Automotive ACC Vehicle Tracking", *Proc. of the IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems*, Heidelberg, Germany, 2006.

[4] T. Gindele, S. Brechtel, J. Schröder, R. Dillmann. "Bayesian Occupancy Grid Filter for Dynamic Environments Using Prior Map Knowledge", *Proc. of the IEEE Intelligent Vehicles Symp.*, Xi'an, China, 2009.

[5] C. Coué, C. Pradalier, C. Laugier, T. Fraichard, P. Bessière. "Bayesian Occupancy Filtering for Multitarget Tracking: An Automotive Application", *Int. J. Robotics Research*, No. 1, 2006.

[6] M. K. Tay, K. Mekhnacha, C. Chen, M. Yguel, C. Laugier. "An Efficient Formulation of the Bayesian Occupation Filter for Target Tracking in Dynamic Environments", *Int. J. Autonomous Vehicles*, 6(1-2):155-171, 2008.

[7] H.P. Moravec. "Sensor Fusion in Certainty Grids for Mobile Robots", *AI Magazine*, 9(2), 1988.

[8] S. Thrun, W. Burgard, D. Fox. "Probabilistic robotics", *MIT Press*, 2005.

[9] K. Mekhnacha, Y. Mao, D. Raulo, C. Laugier. "Bayesian Occupancy Filter based "Fast Clustering-Tracking" Algorithm", *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Nice, France, 2008.

[10] C. Tay. "Analysis of Dynamics Scenes: Application to Driving Assistance", *PhD Thesis*, INRIA, France, 2009.

[11] I. E. Paromtchik, C. Laugier, M. Perrollaz, M. Yong, A. Nègre, C. Tay. "The ArosDyn Project: Robust Analysis of Dynamic Scenes", *Proc. of the Int. Conf. on Automation, Robotics, and Computer Vision*, Singapore, 2010.

[12] M. Perrollaz, R. Labayrade, R. Gallen, D. Aubert. "A Three Resolution Framework for Reliable Road Obstacle Detection Using Stereovision", *Proc. of the IAPR MVA Conf.*, 2007.

[13] M. Perrollaz, J.-D. Yoder, C. Laugier. "Using Obstacle and Road Pixels in the Disparity Space Computation of Stereo-vision based Occupancy Grids", *Proc. of the IEEE Int. Conf. on Intelligent Transportation Systems*, Madeira, Portugal, 2010.

[14] C. M. Bishop. "Pattern Recognition and Machine Learning", *Springer*, 2006.

[15] G. Welch, G. Bishop. "An Introduction to the Kalman Filter", `http://www.cs.unc.edu/~welch/kalman/kalmanIntro.html`

[16] TYZX, `http://www.tyzx.com/products/cameras.html`

[17] IBEO Lux Manual, `http://www.ibeo-as.com/english/products_ibeolux.asp`

[18] Xsens MTi-G Manual, `http://www.xsens.com/en/general/mti-g`

[19] CyCab Toolkit, `http://cycabtk.gforge.inria.fr`