Object Perception for Intelligent Vehicle Applications: A Multi-Sensor Fusion Approach

Trung-Dung Vu, Olivier Aycard, Fabio Tango

Object Perception for Intelligent Vehicle Applications: A Multi-Sensor Fusion Approach

Trung-Dung Vu, Olivier Aycard and Fabio Tango

***Abstract*— The paper addresses the problem of object perception for intelligent vehicle applications with main tasks of detection, tracking and classification of obstacles where multiple sensors (i.e.: lidar, camera and radar) are used. New algorithms for raw sensor data processing and sensor data fusion are introduced making the most information from all sensors in order to provide a more reliable and accurate information about objects in the vehicle environment. The proposed object perception module is implemented and tested on a demonstrator car in real-life traffics and evaluation results are presented.**

I. INTRODUCTION

In this paper we will describe our advanced research work on the object perception problem which is carried out within the framework of the European project *interactIVe*[1] (2009-2013). The project has aimed at pushing the safety of road transport towards the goal of accident-free traffic by developing advanced driver assistance systems (ADAS) for safer and more efficient driving. In the project, a common perception platform was designed and developed which allows an easy and flexible adaptation for a variety of applications with different demonstrator cars on which different sets of sensors are equipped. The designed perception platform is comprised of different modules dealing with tasks at different levels, such as sensor refinement, object perception and situation understanding, where the research on novel algorithms of sensor data processing as well as sensor data fusion are emphasized in order to provide a more reliable and accurate information about the vehicle environment.

Our contributions to the object perception task presented in this paper is addressed in the frontal object perception (FOP) module which is developed as part of the common perception platform. The FOP module is designed to take input from different sensors (i.e.: lidar, camera, radar) and perform tasks of detection, tracking and classification of obstacles appear in front of the vehicle. While the object detection provides knowledge about the presence of obstacles including static and dynamic ones, the object tracking allows the prediction of future behavior of moving objects which is a very important information for safety applications in highly dynamic environments. Besides, the object classification provides further information about different type of obstacles on the road, such as vulnerable users (e.g.: pedestrians) and other vehicles which helps the target application to decide suitable actions in case of confronting a developing dangerous situation.

T-D. Vu is with INRIA Rhône-Alpes, Grenoble, France (e-mail: Trung-Dung.Vu@inria.fr)

O. Aycard is with Laboratoire d'Informatique de Grenoble (LIG), Université Joseph Fourier (UJF), France (e-mail: Olivier.Aycard@imag.fr)

F. Tango is with Centro Ricerche Fiat (CRF), Turino, Italy (e-mail: Fabio.Tango@crf.it)

[1]http://www.http://interactive-ip.eu/

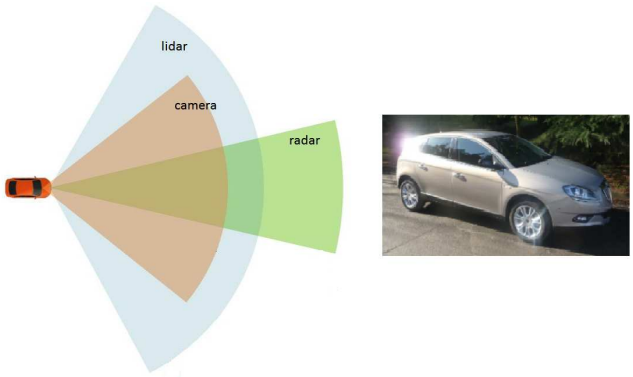

Fig. 1. Sensor setup on the CRF demonstrator.

A. Related works

In the literature, the object perception problem with its main tasks of detection, tracking and classification of objects have been active research topics. Due to the limited characteristic of individual perception sensors, single-sensor approaches have been revealed its flaws. For instance, the radar provides good information for the object detection and tracking but it provides no information for the classification. Additionally, the radar has difficulty to detect non-rigid objects like pedestrians. State-of-the-art vision system [6] provides a very interesting way for the detection and tracking of a specific class of object (ex: pedestrian). However the image processing is usually time-consuming which makes the vision-based systems not suitable for real-time applications especially when more than one object class are considered. Using the lidar with very reliable source of possible detections, a very good system for the tracking of generic objects can be obtained [15]. However the lidar only sees visible part of the object, the object classification with lidar data is not easily determined and the object tracking can be severely affected as indicated in [9]. Using a model-based approach [9], the tracking with lidar data can be improved, but unfortunately this method is limited to the detection and tracking of vehicles only.

To overcome these limitations, in this paper we address the object perception problem by a multi-sensor based approach where the detection, tracking and classification of objects are solved simultaneously and different object classes (i.e.: pedestrians, bikes/motorbikes, cars, trucks) are taken

into account. Firstly new algorithms for raw sensor data processing (e.g.: lidar, camera) are introduced for a fast and robust extraction of objects of interest. Then a fusion process at object-level is employed to combine the most information from all sensors. While lidar data allows for a better estimation of object's geometry and a better tracking performance; camera images allow for a better object class information. Final outcome will be a better and more reliable representation of detected objects in the surroundings. To demonstrate that our proposed method is able to meet critical requirements for automotive applications, we also present experiments and evaluation results.

B. Experimental platform

All the experiment and test results reported in this paper are performed with the CRF demonstrator car on which the FOP module is implemented and integrated. The car is equipped with a 2D lidar, a radar and a mono camera with the configuration and sensor coverage is shown in Fig. 1.

C. Paper outline

The rest of the paper is organized as follows. In the next section, we describe in detail our implementation of the FOP module together with the lidar and camera data processing as well as the fusion process. Section III presents test results with qualitative and quantitative performance evaluation of the FOP module in different scenarios. Section IV will concludes the paper and future works are given.

II. FRONTAL OBJECT PERCEPTION

The architecture of the FOP module is depicted in Fig. 2. The FOP module takes inputs from camera, radar, lidar sensors. In addition, the ego-vehicle dynamics information is provided by another module in the perception platform, named Vehicle State Filter (VSF). The FOP module delivers as output a list of objects (tracks) together with object classification information. While the lidar and camera provide raw data at low-level in terms of points and images, respectively, the radar provides high-level data in terms of detected targets. Additionally, the FOP module takes vehicle dynamics information from another module in the common perception platform.

At the beginning, raw data from the lidar and the camera are processed and objects are extracted separately before being incorporated with the radar data all together at a fusion stage which is done at object-level to decide the final output. Furthermore, for the target application, we would like to pay more attention to several classes of road-users (i.e.: pedestrians, bikes/motorbikes, cars, trucks), for each object extracted from camera or lidar, a likelihood that object belonging to one of these classes is also estimated. The object class at the final output is also decided at the fusion stage.

In the following, we will detail the data processing at each stage of the FOP module.

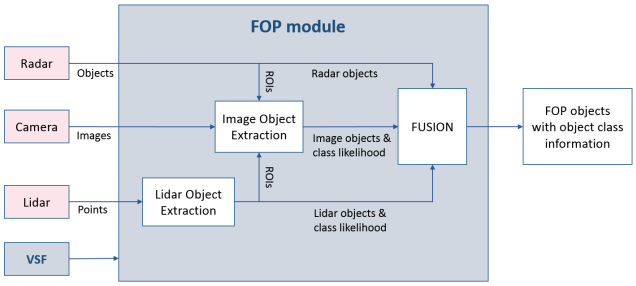

Fig. 2. The Frontal Object Perception architecture.

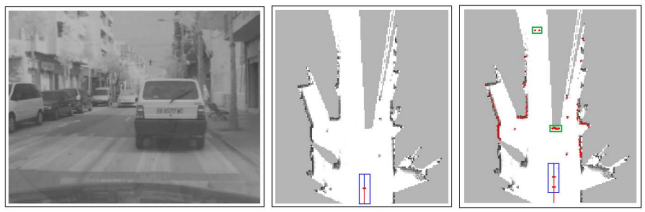

Fig. 3. Grid-based fusion of raw lidar data. From left to right: a) reference situation; b) occupancy grid is built by fusion of all data received; c) new scan received (in red); d) static and moving entities can be identified based on the grid map: green boxes represent moving objects.

A. Lidar Object Extraction

The input to this stage is a list of lidar points which are processed to deliver as output a list of lidar objects including static and dynamic objects.

Static objects: To identify static objects from moving ones, we employ a grid-based fusion approach which was developed in our previous work [14]. A occupancy grid is used to represent a static map of the local environment which is constructed incrementally. In this representation, the environment is divided into a two-dimensional lattice of rectangular cells and each cell is associated with a measure indicating the probability that the cell is occupied by an obstacle or not. A high value of occupancy grid indicates the cell is occupied and a low value means the cell is free. An example of the occupancy grid is illustrated in Fig. 3, where color of each pixel indicates the occupancy of the cell: black: occupied, white: free, grey: unexplored cell that has no information yet. Based on the constructed grid, when a new lidar data measurement is received, a static object and dynamic object can be detected if it appears at occupied or object-free regions, respectively. Since a static object can be of various size and shape, it is then extracted in a form of contour points or a bounding box of measurements depending on the target application. And as remarked in our previous work, one obvious advantage of using grid-map representation compared with other approaches using feature-based [5] that the noise and sparseness of raw lidar data can be inherently handled and at this low-level fusion, no data association is required.

Dynamic objects: For dynamic objects, after being detected from the grid map, we would like to track them in order to estimate their dynamics and can predict their future behaviors. A conventional detection-before-tracking

approach [3] can be applied here using classic data association algorithms like the Joint Probability Data Association (JPDA) [10] or Mutiple Hypothesis Tracking (MHT) [2]. However, these approaches face two well-known problems as described in [13]. Firstly, due to the inherent discreteness of the grid and threshold functions, moving object detection at one time instant usually results in ambiguities with missed/false detections and objects can be split into several segments that make data association for tracking sometimes very difficult. Secondly, due to the fact that the lidar sensor only sees part of object, object extraction in this way does not always reflect the true geometry of the object which severely affects the accuracy of the tracking result.

To overcome these drawbacks, we have an important remark that the number of classes of moving object of interest is quite limited and fortunately they can be represented by simple geometric models, for example: rectangle for vehicles and bicycles, small circle for pedestrians. In this work, we introduce a new algorithm using a model-based approach which formulates the detection and tracking of moving objects as a batch optimization problem using a temporal sliding window over a fixed number of data frames. Dynamic measurement detection at a single frame based on the grid map mentioned previously is now used as a coarse detection that provides bottom-up evidences about potential moving objects. Since these evidences are actually visible parts of the objects, they are used to generate hypotheses of the actual object using all possible object models. Object hypotheses generated from all frames are then put into a top-down formulation (a global view) taking into account all object motion models and sensor models. This leads to an optimization problem where we search for a set of trajectories of moving objects explaining the best of the measured data. The optimal solution is found by a very efficient sampling technique that can meet the real-time requirement (in several tens of milliseconds). More detail description about the algorithm can be found in our published work [13].

Our new approach for tracking dynamic objects has many advantages. Firstly the detection and tracking of objects are solved simultaneously taking into account data from several frames which significantly reduces the ambiguities that might be caused by the detection at a single frame. Secondly, using our model-based approach, object geometry are estimated more accurately which also helps to improve the overall tracking results. Thirdly object class information is naturally given by the chosen object model.

B. Image Object Extraction

For the target applications, we would like to pay more attention about several object classes, namely pedestrians, cars and trucks. While information about the class of objects extracted from the lidar can be estimated based on its estimated geometry. Camera images with rich appearance information can help to provide more accurate about the type of objects.

Fig. 4. Different detection windows with informative blocks are selected for each class of object, from left to right: pedestrian, car and truck (for visibility purpose, only some of them are displayed). Histograms of gradients are computed over these sparse blocks and concatenated to form SHOG features.

To identify these objects of interest from the camera images, we follow most popular approaches using a sliding-window paradigm where a detection window is tried at different positions and scales. For each window, visual features are extracted and a classifier (usually pre-trained off-line) is applied to decide if an object of interest is contained inside. In general, the choice of image representation and classification methods decides the performance of the whole system.

Image representation: we based our approach on the work of Dalal and Triggs [4] on the histograms of oriented gradients (HOG) which has recently become a state-of-the-art feature in computer vision domain for object detection tasks. In their original idea, a detection window is divided into a dense grid of cells and histograms of gradients are computed over all overlapping square blocks of four cells adjacent. From experiments, we found out that for a given class of object (e.g.: pedestrian, vehicle), a block is not necessarily square and by only using a few of the most informative blocks we could represent the object image to obtain similar performance with a benefit of much less computational effort. The resulting feature vector is quite compact with its dimension of about a few hundred compared with about several thousands in the original method. Fig. 4 illustrates some of our blocks selected to extract features for different object classes: pedestrian, car, truck. It turns out that these selective blocks correspond to meaningful regions of the object image (for example: head, shoulder, legs for the pedestrian class). We call our feature selection method SHOG which stands for Sparse HOG feature. In our implementation, we used 6 histogram bins for all object classes, 9 blocks for pedestrian, 7 blocks for car and 7 blocks for truck. To accelerate the SHOG feature computation process, we employed the idea of using integral image introduced by Viola [12]. We compute and store an integral image for each bin of the HOG (resulting in 6 images in our case) and use them to compute efficiently the HOG for any rectangular image region which requires only 4*6 image access operations.

Image classifier: Given computed features, the choice of classifiers has a substantial impact on the resulting speed and quality. To achieve a suitable trade-off, we chose the discrete Adaboost method [7], a boosting-based learning algorithm. The idea of a boosting-based classifier is to

Fig. 5. Examples of successful classification-based object detection of pedestrians and cars from images.

combine many weak classifiers to form a powerful one where weak classifiers are only required to perform better than chance hence they can be very simple and fast to compute. For each object class of interest (e.g.: pedestrian, car, truck), a binary classifier is pre-trained to identify object (positive) and non-object (negative) images. For the off-line training stage, positive images are collected from public datasets [1] or manually labeled containing objects of different viewpoints, for example: pedestrian (frontal, profile), car (frontal, rear side), truck (frontal, rear side). They are all scaled to have sampling images of the same size for each object class: pedestrian: 32x80 pixels, car: 60x48 pixels, truck: 60x60 pixels. Negative samples are generated randomly from images which do not contain an object of interest. SHOG features are computed for all samples which are then used for training classifiers.

The training process starts where each training sample is initially assigned the same weight and iterates for a fixed number of times. On each round, a weak classifier is trained on the weighted training data and its weighted training is recomputed. The weights are increased for training samples being misclassified so that the weak classifier is forced to focus on these hard samples in the next step. The final classifier is the sign of weighted sum over individual learned weak classifiers. In our implementation, decision trees are used as weak classifiers in this boosting scheme.

Image object detection: Final classifiers for each object class (i.e.: pedestrian, car, truck) obtained after the off-line training are used for the online object detection stage in the sliding-window scheme. Detection time is affected by both phases of feature extraction and classification. Thanks to the use of the integral image, the feature extraction step is fast only taking about 10ms or less. Likewise, the classification time is very fast taking only about 2ms per 100 samples. For an input image of size 752x250 pixels, there are about several thousand windows to check and the whole detection time is about 70ms for each object class. Fig. 5 shows examples of pedestrian and car detection results (green and red boxes respectively) before merging into the final objects.

Speed-ups: Although the image object detection process is quite fast, we still need a lot of speed-ups since the total time allowed for both FOP and MOC modules is only 75ms. Instead of searching for the whole input image, we make use of information about targets detected by radar sensor and lidar processing module described above to focus on some regions of interest (ROIs) in the image. Thanks to the sensor calibration parameters, we can compute the homograph to transform coordinates of radar and lidar targets onto the image to calculate ROIs. In this way, the number of sliding windows per image can be then reduced to several hundreds that makes the whole image detection process in only about 20-30ms.

Image object classification: In our image-based object detection process with the sliding-window scheme, the likelihood of object class can be naturally estimated based on the number of detection around object location. Basically, the greater the number of positive windows (containing object of interest), the greater is the probability that the object belongs to that class. False alarms are often returned with very few positive responses.

C. Object Fusion

At this stage, a unified fusion process takes place to fuse all information from list of objects detected by different sensors (i.e.: lidar, camera, radar) in order to decide the final FOP output. Since sensors have different fields of view (Fig. 1), the fusion is performed only in the overlapping region in the common coordinate system. Moreover, different sensors have different characteristics, the fusion aims to make use of the complementary information of these sensors to improve the overall object detection and classification provided by individual sensor. Additionally, conflict evidences can be used to reduce the number of false positives and missed detection/classification. Our fusion approach is based on the Dempster-Shafer (DS) theory [11]. It takes, as sources of evidences, individual lists of objects provided by all sensors. For each object, its complete state includes information about its location, shape, size and velocity together with individual object classification. Using the DS theory we are able to represent evidences about these object features coming from different sensor detectors, and their classification likelihood into a common representation. The proposed fusion process relies on two main parts: the instantaneous fusion, obtained from the combination of evidence provided by individual sensor per object at current time; and the dynamic fusion, which combines evidence from previous times with the instantaneous fusion result. The mechanism used to combine the sources of evidence is a proposed rule of combination based on the one presented in [16]. This mechanism allows us to give more support to common hypothesis and use complementary evidence by managing situations with different levels of conflict without getting counter-intuitive results. These situations usually appear when sensors with low reliability are used, their evidence is noisy or contradictory and when the demonstrator is placed in cluttered scenarios. Given that the performance of the individual object detectors varies according to the type of sensor and their specifications, we included two uncertainty factors into the rule of combination: sensor reliability and sensor precision to certain properties of the returned objects. The final state (location, shape, size, velocity and classification information) for each object is selected as the hypothesis with a highest evidence value after the dynamic fusion is performed. By this way final outcome comprises the most of sensor capabilities to detect specific features of the object. For example, a

camera sensor provides a better approximation of a vehicle width, radar can give a direct measurement of relative speed and lidar sensor can give a more precise moving direction for moving object and gives more accurate measures of object's geometry and size when it is available. Cluttered urban areas are a common scenario where image-based classifiers capabilities help to classify a pedestrian/group of pedestrians correctly where usually lidar is not able to. The output of this stage is a list of FOP objects with all information about the object's properties: location, geometry, dynamics plus the classification information from the fusion process. For more details about this fusion process, the reader can refer to our published work [8].

III. TESTING AND EVALUATION

In the project *interactIVe*, the FOP module is integrated and tested on the CRF demonstrator within the common perception platform, called the Reference Perpetion Platform (RPP), together other perception modules where critical requirements have to be met.

A. Computational time

In the RPP, some modules have dependencies with others and they are designed to run at different stages (levels) which is assured by a common scheduler. The integrated FOP module is triggered to run from level 2 to level 5 of the RPP with a total time allowed of 75ms (per 100ms of one RPP cycle) which is still a challenge for the whole sensor data processing. From the statistics which measure the running time of each RPP module, the average and maximum computing time of the FOP module is about 40ms and 65ms, respectively which fulfills the timming requirement of the designed platform.

B. Qualitative performance

For the qualitative assessment, we would like to verify general functionality of the whole module (i.e.: object detection, tracking and classification). Additionally, we are interested in assessing the advantages of the fusion process.

In the following, we will show some results obtained from different scenarios. Output provided from the FOP module is checked with the camera video to see if all the functions are working as expected. Note that, the FOP output is displayed in both the camera view and the birds-eye view.

Fig. 6 shows two scenarios on the test track. In the first situation, the ego-vehicle is approaching a stationary car. In the camera view, we can see that the target vehicle is well detected and correctly classified. Although it is seen by all sensors: radar (red circles), lidar (green dots) and camera (yellow boxes), only the camera can provide information about object class. The lidar only sees the rear part of the car giving no clue about the type of object. In the second situation, the ego-vehicle is following a moving car. Again this target is seen by all sensors and is correctly classified as a car. In this situation, when the target car moves, the lidar is able to estimate the target model which supports the correct classification. The accuracy of the lidar tracking algorithm is verified by comparing the lidar-based estimated speed with the speed provided by the radar sensor and the speed of the ego-vehicle. However, while the radar only provides Doppler velocity of the target and no information about target moving direction, thanks to the lidar tracking module, the car moving direction and its geometry are both well estimated. For the assessment of dangerous situations, this information of moving target is very important.

Fig. 7 shows examples of detecting pedestrians on the test track. In the first situation, two pedestrians are crossing each other in the frontal area and in the second situation two pedestrians are moving, closely, towards the ego-vehicle. In both cases, we observe that radar detection of pedestrians is not fully reliable in particular for distances above 30m. On the other hand they are well detected and tracked by the lidar. However, only the camera is able to provide good class information of objects. Two pedestrians in the first test are well recognized and the final target in the second test is correctly classified as a group of pedestrians thanks to the image classification module.

Fig. 8 shows output examples of the FOP module in two real-life scenarios: one on a highway and one on an urban road. Although these scenarios contain lots of traffic, all vehicles moving in two directions are well detected, tracked and correctly classified as cars and trucks. In these examples, static objects (e.g.: barriers) are also reported in the birds-eye view. Moving objects are distinguished by attached velocities and their moving directions are well estimated thanks to the lidar tracking module. Note that, in the object-level fusion stage, the radar Doppler velocity information helps to improve the target speed estimated by the lidar after its moving direction is known. However the radar only covers a small frontal area (FOV of $15°$) compared with the lidar area (FOV of $110°$).

We can see that in all tests performed, from specific test scenarios to real-life traffic scenarios, the FOP module with all functions of detection, tracking and classification has been working well as expected. And it is very interesting to see that data fusion process help to make use of the best characteristics of different sensors into the final perception output. The state of object at output contains lots of information: location, geometry, object class, speed, moving direction (for moving ones) that cannot be provided by only one individual sensor.

C. Quantitative performance

Since there is no ground-truth data available at the testing moment, we have evaluated the performance of FOP-MOC module manually and we focus on the detection and classification functions since they are more critical to the target application. The evaluation procedure will be conducted as follows. We choose some typical scenarios from the available dataset and perform a frame-by-frame evaluation. For each data frame, we label objects of interest (e.g.: car, truck, pedestrian) identifiable by human eyes from the camera video. For each object, we will count for how many frames it is correctly detected and classified. The number of wrong-

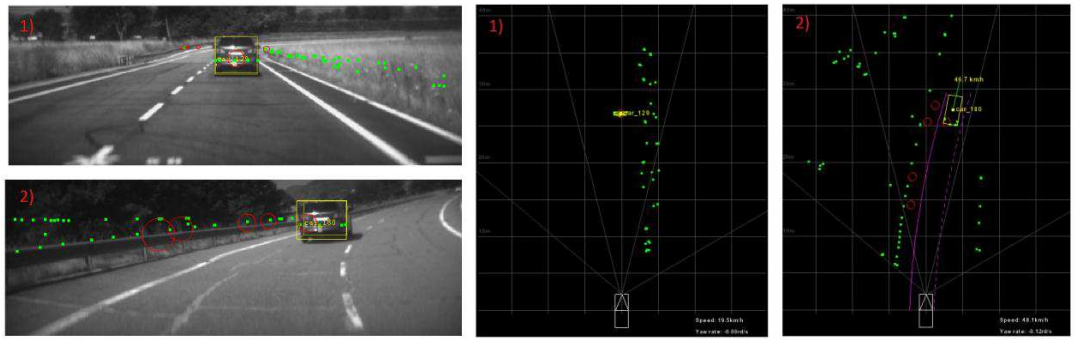

Fig. 6. Detection, tracking and classification of stationary and moving vehicles on the test track.

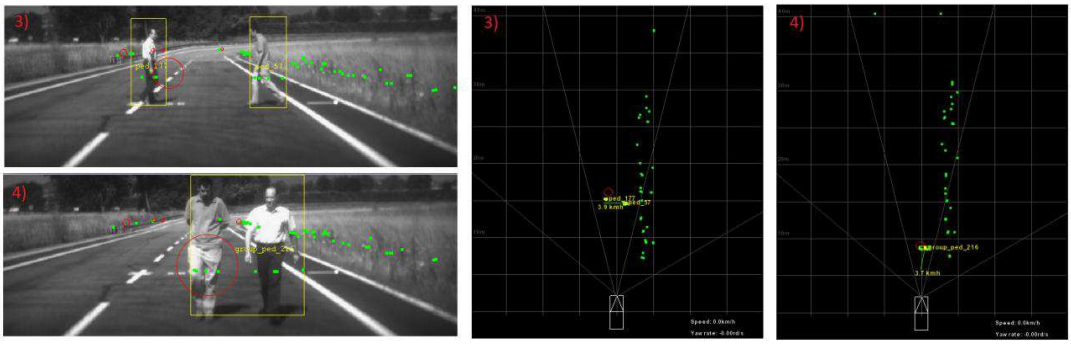

Fig. 7. Detection, tracking and classification of pedestrian/group of pedestrians on the test track.

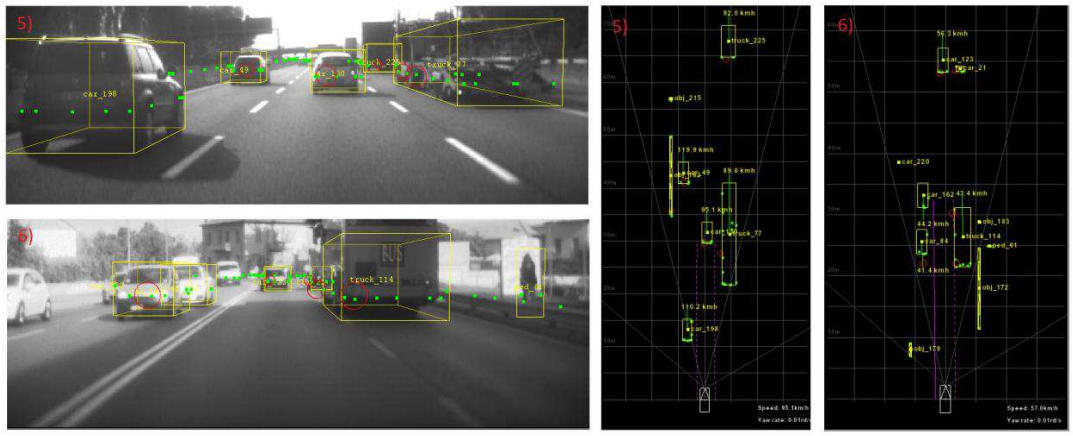

Fig. 8. Examples of successful detection, tracking and classification of pedestrians and cars in real-life scenarios.

Scenarios	Total objects			Correct Detection			False Detection			Correct Classification			False Classification		
	ped	car	truck	ped	car	truck	ped	car	truck	ped	car	truck	ped	car	truck
Motorway	0	682	216	0	655	201	0	20	0	0	630	175	0	2	0
				n/a	**96,0%**	**93,1%**	n/a	**2,9%**	**0%**	n/a	**92,4%**	**81,0%**	n/a	**0,3%**	**0,0%**
Urban	33	525	87	27	495	72	4	0	0	26	483	63	5	4	5
				81,8%	**94,3%**	**82,8%**	**12,1%**	**0,0%**	**0,0%**	**78,8%**	**92,0%**	**72,4%**	**15,2%**	**0,8%**	**5,7%**
Test track	248	301	0	247	300	0	0	1	0	240	300	0	0	0	0
				99,6%	**100%**	n/a	**0,0%**	**0,3%**	n/a	**96,8%**	**100%**	n/a	**0,0%**	**0,0%**	n/a

Fig. 9. Quantitative evaluation of the FOP module.

detections and wrong-classifications (false positives) are also counted.

Fig. 9 summarizes the results collected after testing the FOP module with data from different scenarios. The testing scenarios are grouped into three categories: motorway, urban road and test track. Bikes/motorbikes rarely appear in any of the available test data, so this object category is omitted in the table. We can see that in all tests performed, for all considered objects of interest, high detection and classification rates are achieved with relatively low false positives. In the test track scenarios where only one car or a few pedestrians are present, the detection and classification rate of pedestrians and cars is nearly perfect (97% and 100% respectively). In the motorway scenarios, the detection rate of vehicles is also very good: car (96%), truck (93%) where the missed detections are due mainly to inherent noisy and clutter data (for example: lidar hitting ground cant see object). The large size of the truck explains the truck detection is not as good as the car detection since it is sometimes confused with the barrier. The false detection (false positives) of cars (3%) is due mainly to the reflection which creates ghost objects. However, the false positives are very low for the vehicle detection and classification thanks to the fusion process from different sensors. In the urban scenarios, the vehicle detection and classification is still high (83% and 94% respectively). However the pedestrian detection goes down to 82% with a false positive for the detection of 12%. This is mainly due to the fact that in urban roads, there are lots of traffic posts that are easily detected and misclassified as pedestrians.

We can see that from the initial quantitative evaluation process, the FOP module is shown to perform well providing quite reliable object perception outputs in terms of detection, tracking and classification while maintaining the tight computational time requirement of the Reference Perception Platform. A more complete quantitative evaluation, in particular the tracking evaluation, will be part of our future work when the ground-truth data for the testing scenarios is available.

IV. CONCLUSIONS

We have presented our solution for an advanced object perception task using different sensors (i.e.: lidar, camera and radar) through the integrated FOP module which is developed within the *interactIVe* project. Firstly, new algorithms for raw sensor data processing (i.e.: lidar, camera) are introduced which help to obtain better and more reliable results. Secondly, a unified high-level fusion is described to make use of the best information from individual sensors to the final output. Finally, promising results obtained through the initial test and the evaluation process has confirmed the efficiency and the applicability of our perception module for real-time automotive applications. Future works focusing on the quantitative evaluation dedicated to the tracking result assessment are foreseen.

V. ACKNOWLEDGMENTS

The work is supported by the European project *interactIVe* which is part of the FP7-ICT program for Safety and Energy Efficiency in Mobility. The authors would like to thank all partners within the project for their fruitful cooperation and valuable contribution.

REFERENCES

[1] Computer vision datasets. http://www.cvpapers.com/datasets.html.
[2] S. S. Blackman. Multiple hypothesis tracking for multiple target tracking. *IEEE Aerospace and Electronic Systems Magazine*, 19(1):5–18, Jan 2004.
[3] Samuel Blackman and Robert Popoli. *Design and Analysis of Modern Tracking Systems*. Artech House, Norwood, MA, 1999.
[4] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *CVPR*, pages 886–893, 2005.
[5] Tobias Einsele and Georg Farber. Real-time self-localization in unknown indoor environments using a panorama laser range finder. In *IEEE/RSJ International Workshop on Robots ans Systems, IROS 97*, pages 697–703. IEEE Press, 1997.
[6] A. Ess, B. Leibe, K. Schindler, , and L. van Gool. A mobile vision system for robust multi-person tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08)*. IEEE Press, June 2008.
[7] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. Additive logistic regression: a statistical view of boosting. *Annals of Statistics*, 28:2000, 1998.
[8] Ricardo Omar Chavez Garcia, Trung-Dung Vu, Olivier Aycard, and Fabio Tango. Fusion framework for moving-object classification. In *FUSION*, pages 1159–1166, 2013.
[9] Anya Petrovskaya and Sebastian Thrun. Model based vehicle tracking for autonomous driving in urban environments. In *Proceedings of Robotics: Science and Systems IV (RSS)*, Zurich, Switzerland, June 2008.
[10] D. Schulz, W. Burgard, D. Fox, and A.B. Cremers. Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2001.
[11] Glenn Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.
[12] Paul Viola and Michael Jones. Robust real-time object detection. *International Journal of Computer Vision*, 2001.
[13] Trung-Dung Vu and Olivier Aycard. Lased-based detection and tracking moving object using data-driven markov chain monte carlo. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, May 2009.
[14] Trung-Dung Vu, Olivier Aycard, and Nils Appenrodt. Online localization and mapping with moving object tracking in dynamic outdoor environment. In *Proceedings of the IEEE Intelligent Vehicle Symposium*, Istanbul, Turkey, June 2007.
[15] Chieh-Chih Wang, Charles Thorpe, Sebastian Thrun, Martial Hebert, and Hugh Durrant-Whyte. Simultaneous localization, mapping and moving object tracking. *The International Journal of Robotics Research*, 26(9):889–916, September 2007.
[16] Ronald R. Yager. On the Dempster-Shafer framework and new combination rules. *Inf. Sci.*, 41(2):93–137, March 1987.