Online Localization and Mapping with Moving Object Tracking in Dynamic Outdoor Environments with new Demonstrator Data

Qadeer Baig, Trung-Dung Vu and Olivier Aycard

Abstract— **In this paper, we present a real-time algorithm for online simultaneous localization and mapping (SLAM) with detection and tracking of moving objects (DATMO) in dynamic outdoor environments from a moving vehicle equipped with laser sensor and odometry. To correct vehicle location from odometry we introduce a new fast implementation of incremental scan matching method that can work reliably in dynamic outdoor environments. After a good vehicle location is estimated, the surrounding map is updated incrementally and moving objects are detected without a priori knowledge of the targets. Detected moving objects are finally tracked using Global Nearest Neighborhood (GNN) method. The experimental results on dataset collected from INTERSAFE-2 demonstrator for typical scenario show the effectiveness of this technique.**

I. INTRODUCTION

Perceiving or understanding the environment surrounding a vehicle is a very important step in driving assistance systems or autonomous vehicles. The task involves both simultaneous localization and mapping (SLAM) and detection and tracking of moving objects (DATMO). While SLAM provides the vehicle with a map of the environment, DATMO allows the vehicle being aware of dynamic entities around, tracking them and predicting their future behaviours. If we are able to accomplish both SLAM and DATMO reliably in real time, we can detect critical situations to warn the driver in advance and this will certainly improve driving safety and can prevent traffic accidents.

Recently, there have been considerable research efforts focusing on SLAM and DATMO [24][14][33][34]. However, for highly dynamic outdoor environments like crowded urban streets, there still remain many open questions. These include, how to represent the vehicle environment and how to differentiate moving objects and stationary objects as well as how to track moving objects over time.

In this context, we design and develop a generic architecture to solve SLAM and DATMO in dynamic outdoor environments. The architecture (Fig. 1) is divided into two main parts: the first part where the vehicle environment is mapped and moving objects are detected; and the second part where previously detected moving objects are verified and tracked.

In the first part of the architecture, to model the environment surrounding the vehicle, we use the Occupancy Grid framework developed by Elfes [10]. Compared with feature-based approaches [16], grid maps can represent any environment and are specially suitable for noisy sensors in

Qadeer Baig, T.D. Vu and O. Aycard are with e-Motion Team & INRIA Rhône-Alpes, Grenoble, France, (e-mails: Qadeer.Baig@inrialpes.fr, Trung-Dung.Vu@inrialpes.fr, Olivier.Aycard@imag.fr)

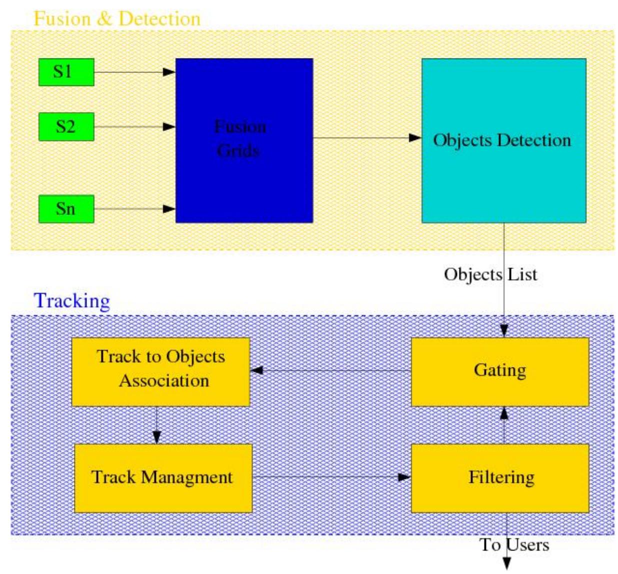

Fig. 1. Architecture of the perception system

outdoor environments where features are hard to define and extract. Grid-based approaches also provide an interesting mechanism to integrate different kinds of sensors in the same framework taking the inherent uncertainty of each sensor reading into account. On the contrary of a feature based environment model, the only requirement for an OG building is a bayesian sensor model for each cell of the grid and each sensor. This sensor model is the description of the probabilistic relation that links sensor measurement to space state, that OG necessitates to make the sensor integration. Fortunately it is possible for a wide class of sensors to factorise this amount of data by taking advantage of the characteristics of the sensor. Regarding telemetric sensors, sensor model for sonar [35] and laser range finders [26] have been defined and used to map the environment. 3D occupancy grids have been built using stereo vision[19] and a set of camera [12].

In the second part, detected moving objects in the vehicle environment are tracked. Since some objects may be occluded or some are false alarms, multi object tracking helps to identify occluded objects, recognize false alarms and reduce miss-detections. In general, the multi objects tracking problem is complex: it includes the definition of filtering methods, but also association methods and maintenance of the list of objects currently present in the environment [3][28]. Regarding tracking techniques, Kalman filters [15] or particle filters [1] are generally used. These filters require the definition of a specific dynamic model of tracked objects. However, defining a suitable motion model is a real difficulty. To deal with this problem, Interacting Multiple Models [18][25] have been successfully applied in several applications. In previous works [7][6], we have developed a fast method to adapt on-line IMM according to trajectories of detected objects and so we obtain a suitable and robust tracker. To deal with the association and maintenance problem, we extend our approach to multiple objects tracking using the Multiple Hypothesis Tracker [4][5].

This architecture has been used in the framework of the European project PReVENT-ProFusion[1]. The goal of this project is to design and develop generic architectures to perform perception tasks (i.e., mapping of the environment, localization of the vehicle in the map, and detection and tracking of moving objects). In this context, our architecture has been integrated and tested on two demonstrators: a Daimler-Mercedes demonstrator and a Volvo Truck demonstrator. The main difference between these 2 demonstrators is the level of abstraction of data provided by the different sensors on each demonstrator: raw data for the Daimler-Mercedes demonstrator (i.e., low level of abstraction) and preprocessed data for the Volvo Truck demonstrator (i.e., high level of abstraction). To achieve this task, we have to design and implement specific parts of the first level of the architecture: specific sensor models and also specific techniques to detect moving objects using the occupancy grid. The second level of the architecture remains the same for each demonstrator.

In [29], a description of the specific part of the first level for the Daimler demonstrator is reported: building specific sensor models and designing specific techniques for detecting moving objects. In [13], we described the specificities of our architecture for the Volvo Truck demonstrator: designing and implementing sensor models for preprocessed data and specific techniques for detection of moving objects. In [8] [31], the second level is detailed. Moreover, results and comparison with other perception system for Pre-Crash applications is described in [23].

Our generic architecture is now used on a Volkswagen demonstrator in the european project INTERSAFE-22 related to safety at intersection. On this demonstrator, we will have to fusion data coming from laser, stereo vision and short range radar. The goal of this paper is to present our preliminary results on this demonstrator and also to discuss the future works on this platform.

The rest of the paper is organized as follows. In the next section, we present the Volkswagen demonstrator. A brief overview of Environment Mapping with Occupancy Grid is given in section III. Detection of moving objects using occupancy grid previously built is detailed in section IV. Experimental results are given in Section V. A detailed discussion about the future work on this demonstrator is reported in section VI. Finally, in Section VII conclusions are presented.

[1] www.prevent-ip.org/profusion

II. INTERSAFE-2 AND VOLKSWAGEN DEMONSTRATOR DESCRIPTION

The European Union funded INTERSAFE-2 project aims to develop and demonstrate a Cooperative Intersection Safety System (CISS) that is able to significantly reduce injury and fatal accidents at intersections. Many teams and automobile makers from around the europe are participating in this project. The novel CISS combines warning and intervention functions demonstrated on three vehicles: two passenger cars and one heavy goods vehicle. These vehicles are equipped with different sensors to observe the environment. On the intersections there are some areas which are not visible for the vehicle so called black sopts. In order to prevent accidensts in these areas infrastructure at the intersections is also equipped with sensors and communication equipment to send the observed information to the vehicle. Another important part is the v2v communication where vehicles communicate with each other to share information at intersections.

Our work pertains to the perception module of this project and specifically relates to the fusion of data from different on-board and infrastructure sensors for Volkswagen demonstrator. We will perform this fusion task at different levels of abstraction for different sensors.

Inorder to perform its tasks for this project Volkswagen demonstrator requires specific environmental sensors such as on-board sensors as well as a communication link between vehicle and infrastructure. Figure 2 illustrates the chosen sensor set and coverage area. The sensor set-up includes sensors which are already in serial cars available, namely front ACC radar and rear-looking radar for lane change support. These sensors are accompanied by a stereo camera system to the front with high field of view of about 60. Stereo vision is a very high potential sensor which has the capability of detecting and measuring objects as well as lane delimiters like curbs, stop lines, pedestrians and other important environment. The main advantage of using stereo instead of monocular vision is the precise distance determination capability. A scanning laser with a field of view of about 160 and dedicated radar sensors directed to +90 and -90 respectively are foreseen for measuring the objects coming from the side. These sensors are able to measure position, velocity and some geometrical parameters of the relevant objects at intersections. The communication link (ad-hoc-network, connection-free, IEEE802.11p standard) offers the possibility to transmit information about objects and their trajectories which are not probably in the line of sight or beyond the maximum range of the on-board sensors. This information enriches the on-board sensor measurements.

III. LOCALIZATION AND MAPPING

For a true autonomous robot it is imperative to solve the localization and mapping problem collectively known as SLAM. But for a working safety navigation application a good global map is not necessary but we need a good local map. So in this work we will not consider the revisiting or loop closing part of the SLAM problem. Here we will use the incremental mapping approach based on a fast scan matching

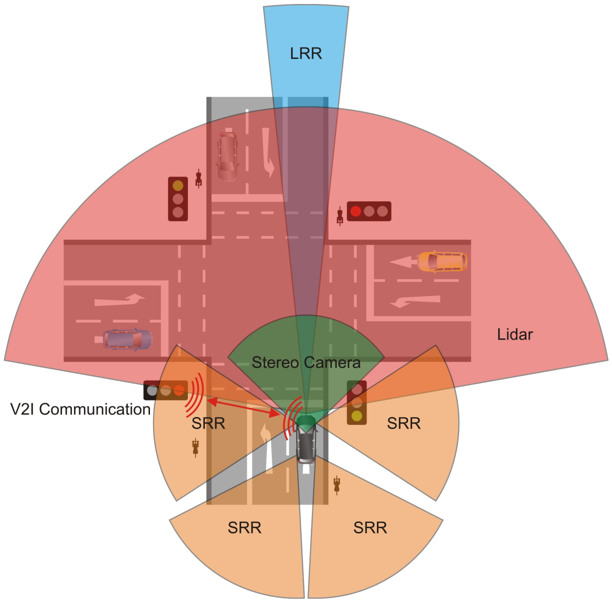

Fig. 2. Sensors installed on the demonstrator vehicle

algorithm presented in [32], and we will verify this technique on our new data set obtained from new demonstrator. This technique involves in building a consistent local vehicle map which is updated incrementally when new data arrive along with good estimates of vehicle locations obtained from the scan matching algorithm. The advantages of this incremental approach are that the computations are very fast and the whole process runs on-line.

A. Notations Used

We denote the discrete time index by the variable t, the laser observation from vehicle at time t by the variable $z_t = \{z_t^1, ..., z_t^K\}$ including K individual measurements corresponding to K laser beams, the vector describing an odometery measurement from time $t-1$ to time t by the variable u_t and it consists of velocity, steering angle and yaw rate, the state vector describing the true location or pose of the vehicle at time t by the variable x_t consisting of xy coordinates and orientation angle. Please note that we use u_t to calculate the corresponding value of x_t.

B. Occupancy Grid Map

Occupancy Grid is a multi-dimensional tessellation of space into cells where each cell stores a probabilistic estimate of its state. In this representation, the vehicle environment is divided into a two-dimensional lattice M of rectangular cells and each cell M_i is associated with an occupancy probability taking a real value in $[0, 1]$ indicating whether a cell is occupied by an obstacle or not. A high value of cell probability indicates the cell is occupied and a low value means the cell is free. Here we make the assumption that occupancy states of individual grid cells are independent. The objective of a mapping algorithm is to estimate the posterior probability of occupancy $P(M \,|\, x_{1:t}, z_{1:t})$ but because of cell independence assumption it is equivalent to estimate $P(M_i \,|\, x_{1:t}, z_{1:t})$ for each cell of grid M_i, given observations $z_{1:t} = \{z_1, ..., z_t\}$ at corresponding known poses $x_{1:t} = \{x_1, ..., x_t\}$.

In the literature, many methods are used for occupancy grid mapping, such as Bayesian [10], Dempster-Shafer [22] and Fuzzy Logic [21]. Here we apply Bayesian Update scheme [28] that provides an elegant recursive formula to update the posterior under log-odds form:

$$\begin{aligned} log\,Odds(M_i \,|\, x_{1:t}, z_{1:t}) &= log\,Odds(M_i \,|\, x_{1:t-1}, z_{1:t-1}) + \\ &+ log\,Odds(M_i \,|\, z_t, x_t) - log\,Odds(M_i) \quad (1) \\ \text{where } Odds(a \,|\, b) &= P(a \,|\, b) \,/\, (1 - P(a \,|\, b)) \end{aligned}$$

In (1), $P(M_i)$ is the prior occupancy probability of the map which is set to 0.5 representing an unknown state. The remaining probability $P(M_i \,|\, x_t, z_t)$, is called the *inverse sensor model*. It specifies the probability that a grid cell M_i is occupied based on a single sensor measurement z_t at location x_t. In our implementation, it is decided by the measurement of the nearest beam to the mass centre of this cell.

It is easy to see that the desired probability of occupancy, $P(M_i \,|\, x_{1:t}, z_{1:t})$, can be recovered from the log-odds representation using the following equation.

$$P(M_i \,|\, x_{1:t}, z_{1:t}) = 1 - \frac{1}{1 + \exp Odds(M_i \,|\, x_{1:t}, z_{1:t})}$$

Moreover, since the updating algorithm is recursive, it allows for incremental map updating when new sensor data arrives.

C. Scan Matching against Occupancy Grid Map

In order to build a consistent map of the environment, a good vehicle localization is required. Because of the inherent error, using only odometery often results in an unsatisfied map (see Fig. 3 left). When features can not be defined and extracted, direct scan matching techniques like ICP [17] can help to correct the odometery error. The problem is that sparse data in outdoor environments and dynamic entities make correspondence finding difficult. One important disadvantage of the direct scan matching methods is that they do not consider the dynamics of the vehicle.

An alternative approach that can overcome these limitations consists in setting up the matching problem as a maximum likelihood problem [27], [14]. In this approach, given an underlying vehicle dynamics constraint, the current scan's position is corrected by comparing with the local grid map constructed from all observations in the past instead of only with one previous scan. Mathematically, we calculate a sequence of poses $\hat{x}_1, \hat{x}_2, ...$ and sequentially updated maps $M^1, M^2, ...$ by maximizing the marginal likelihood of the t-th pose and map relative to the $(t-1)$-th pose and map:

$$\hat{x}_t = \underset{x_t}{\operatorname{argmax}} \{P(z_t \,|\, x_t, M^{t-1}) \,.\, P(x_t \,|\, \hat{x}_{t-1}, u_t)\} \quad (2)$$

In the equation (2), the term $P(z_t \,|\, x_t, M^{t-1})$ is the measurement model which is the probability of the most recent measurement z_t given the pose x_t and the map M^{t-1} constructed so far from observations $z_{1:t-1}$ at corresponding

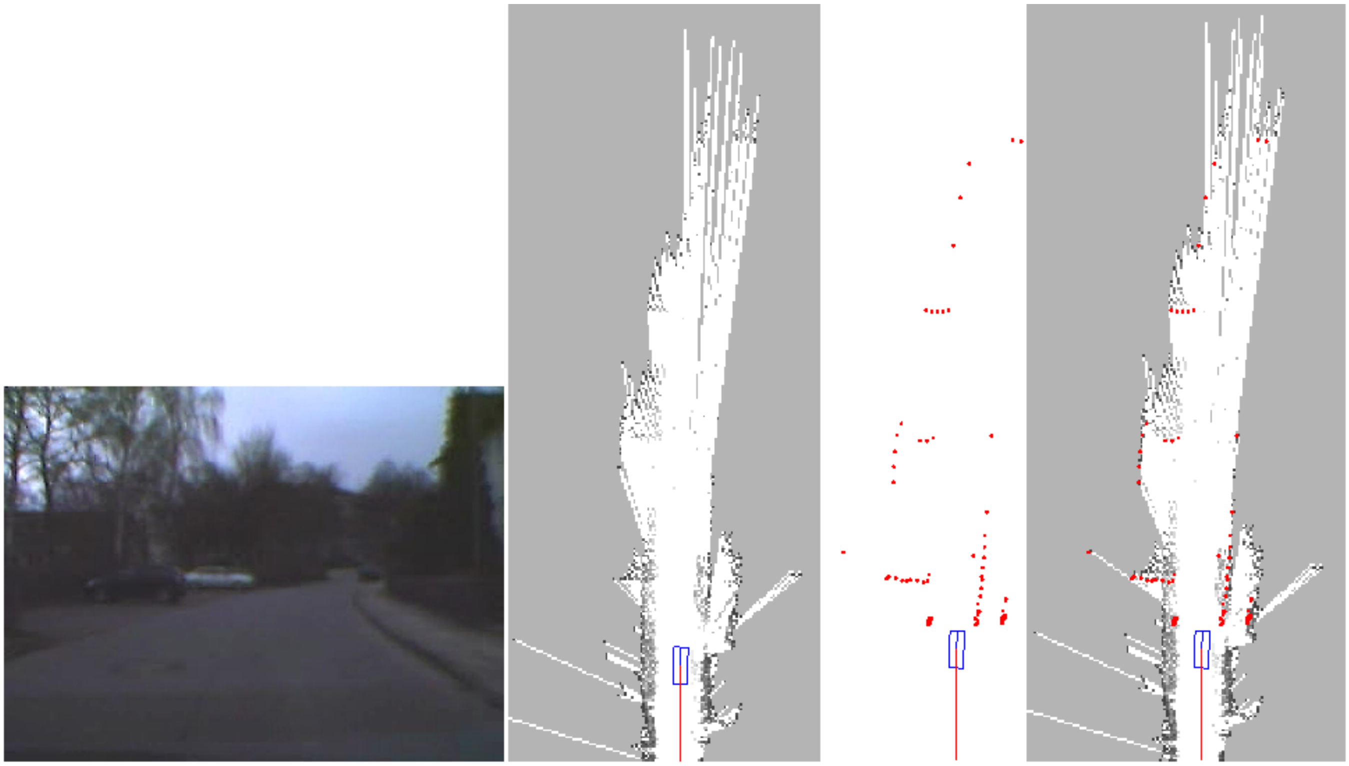

Fig. 4. An example of scan matching. From left to right: reference image; map created so far M^{t-1} and previous vehicle pose x_{t-1}; laser measurement at time t; and matching result is obtained from the consistency of the measurement with the map and the previous vehicle pose.

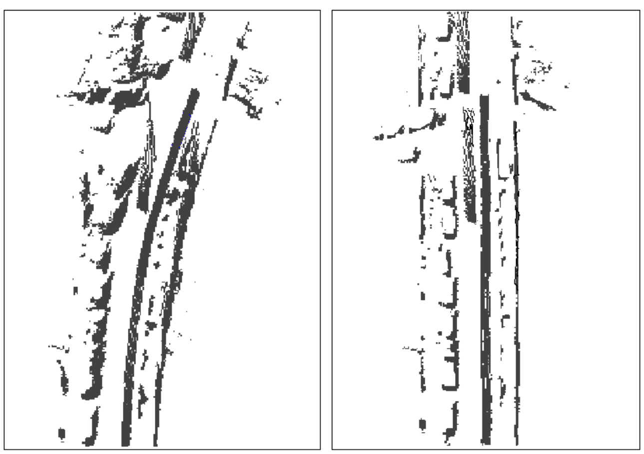

Fig. 3. Hit maps build directly from raw laser data collected from a vehicle moving along a straight street: with vehicle localization using odometery (left); and using results of scan matching (right). Note that the scan matching results are not affected by moving objects in the street.

poses $\hat{x}_{1:t-1}$ that were already estimated in the past. The term $P(x_t \mid \hat{x}_{t-1}, u_t)$ represents motion model which is the probability that the vehicle is at location x_t given that the vehicle was previously at position $\hat{x}_{t-1}$ and executed an action u_t. The resulting pose $\hat{x}_t$ is then used to generate a new map M^t according to (1):

$$M^t = M^{t-1} \cup \{\hat{x}_t, z_t\} \tag{3}$$

Now the question is how to solve the equation (2), but let us first describe the motion model and the measurement model used.

For the motion model, we adopt the probabilistic velocity motion model similar to that of [28]. The vehicle motion u_t is comprised of three components, the translational velocity v_t, steering angle θ_t and the yaw rate ω_t. Fig. 5 depicts the probability of being at location x_t given previous location x_{t-1} and control u_t. This distribution is obtained from the kinematic equations, assuming that vehicle motion is noisy along its rotational and translational components.

For the measurement model $P(z_t \mid x_t, M^{t-1})$, mixture beam-based model is widely used in the literature [11], [14]. However, the model come at the expense of high computation since it requires ray casting operation for each beam. This can be a limitation for real time application if we want to estimate a large amount of measurements at the same time. To avoid ray casting, we propose an alternative model that only considers end-points of the beams. Because it is likely that a beam hits an obstacle at its end-point, we focus only on occupied cells in the grid map. A voting scheme is used to compute the probability of a scan measurement z_t given the vehicle pose x_t and the map M^{t-1} constructed so far. First, from the vehicle location x_t, individual measurement z_t^k is projected into the coordinate space of the map. Call hit_t^k the grid cell that its projected end-point falls into. If this cell is occupied, a sum proportional to the occupancy value of the cell will be voted. Then the final voted score represents the likelihood of the measurement. Let $P(M_i^t)$ denote the posterior probability of occupancy of the grid cell M_i estimated at time t (follows (1)), we can write the measurement model under the sum following:

$$P(z_t \,|\, x_t, M^{t-1}) \propto \sum_{k=1}^{K} \{\, P(M^{t-1}_{hit^k_t}) \,|\, M^{t-1}_{hit^k_t} \text{ is occupied} \,\} \quad (4)$$

The proposed method is just an approximation to the measurement model because it does not take into account visibility constraints, but experimental evidences show that it works well in practice. Furthermore, with a complexity of $O(K)$, the computation can be done rapidly.

It remains to describe how we maximize (2) to find the correct pose $\hat{x}_t$. Hill climbing strategy in [27], [14] can be used but may suffer from a local maximum. Exploiting the fact that the measurement model can be computed very quickly, we perform an extensive search over vehicle pose space. A sampling version of the motion model (Fig. 5 right) is used to generate all possible poses x_t given the previous pose x_{t-1} and the control u_t. The resulting pose will be the pose at which the measurement probability achieves a maximum value. Because of the inherent discretization of the grid, the sampling approach turns out to work very well. In practice, with a grid map resolution of 20 cm, it is enough to generate about four or five hundreds of pose samples to obtain a good estimate of the vehicle pose with the measurement likelihood that is nearly unimproved even with more samples. The total computational time needed for such a single scan matching is about 10 ms on a low-end PC.

An example of scan matching result is shown in Fig. 4. The color of each grid map cell indicates its probability of being occupied: gray=unknown, white=free, black=occupied. The most likely vehicle pose is obtained when the laser scan is aligned with the occupied parts of the map and at the same time the vehicle dynamics constraint is satisfied.

Besides the computational effectiveness, one attraction of our algorithm is that it is not affected by dynamic entities in the environment (see Fig. 3 right). Since we only consider occupied cells, spurious regions in the occupancy grid map that might belong to dynamic objects do not contribute to the sum (4). The voting scheme ensures that measurement likelihood reach a maximum only when the measurement is aligned with the static parts of the environment. To some meaning, measurements from dynamic entities can be considered as outliers. This property is very useful for moving object detection process that will be described in the next section.

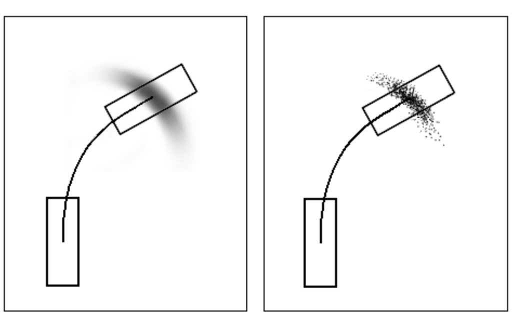

Fig. 5. The probabilistic velocity motion model $P(x_t \,|\, x_{t-1}, u_t)$ of the vehicle (left) and its sampling version (right).

D. Local mapping

Because we do not need to build a global map nor deal with loop closing problem, only one on-line map is maintained at each point in time representing the local environment surrounding of the vehicle. The size of the local map is chosen so that it should not contain loops and the resolution is maintained at a reasonable level. Every time the vehicle arrives near the map boundary, a new grid map is initialized. The pose of the new map is computed according to the vehicle global pose and cells inside the intersection area are copied from the old map.

IV. OBJECT DETECTION AND TRACKING

After a consistent local map of the vehicle is constructed from SLAM, moving objects can be detected when new measurements arrive. The principal idea is based on the inconsistencies between observed free space and occupied space in the local grid map. If an object is detected on a location previously seen as free space, then it is a moving object. If an object is observed on a location previously occupied then it probably is static. If an object appears in a previously not observed location, then we can say nothing about that object.

Another important clue which can help to decide if an object is dynamic or not is evidence about moving objects detected in the past. For example, if there are many moving objects passing through an area then any object that appears in that area should be recognized as a potential moving object. For this reason, apart from the local static map M as constructed by SLAM described in the previous section, a local dynamic grid map D is created to store information about previously detected moving objects. The pose, size and resolution of the dynamic map is the same as those of the static map. Each dynamic grid cell store a value indicating the number of observations that a moving object has been observed at that cell.

From these remarks, our moving object detection process is carried out in two steps as follows. The first step is to detect measurements that might belong to dynamic objects. Here for simplicity, we will temporarily omit the time index. Given a new laser scan z, the corrected vehicle location and the local static map M computed by SLAM and the dynamic map D containing information about previously detected moving objects, state of a single measurement z^k is classified into one of three types following:

$$state(z^k) = \begin{cases} static & : M_{hit^k} = occupied \\ dynamic & : M_{hit^k} = free \text{ or } D_{hit^k} > \alpha \\ undecided & : M_{hit^k} = unknown \end{cases}$$

where hit^k is the coordinate of the grid cell corresponding to the end-point of the beam z^k and α is a predefined threshold.

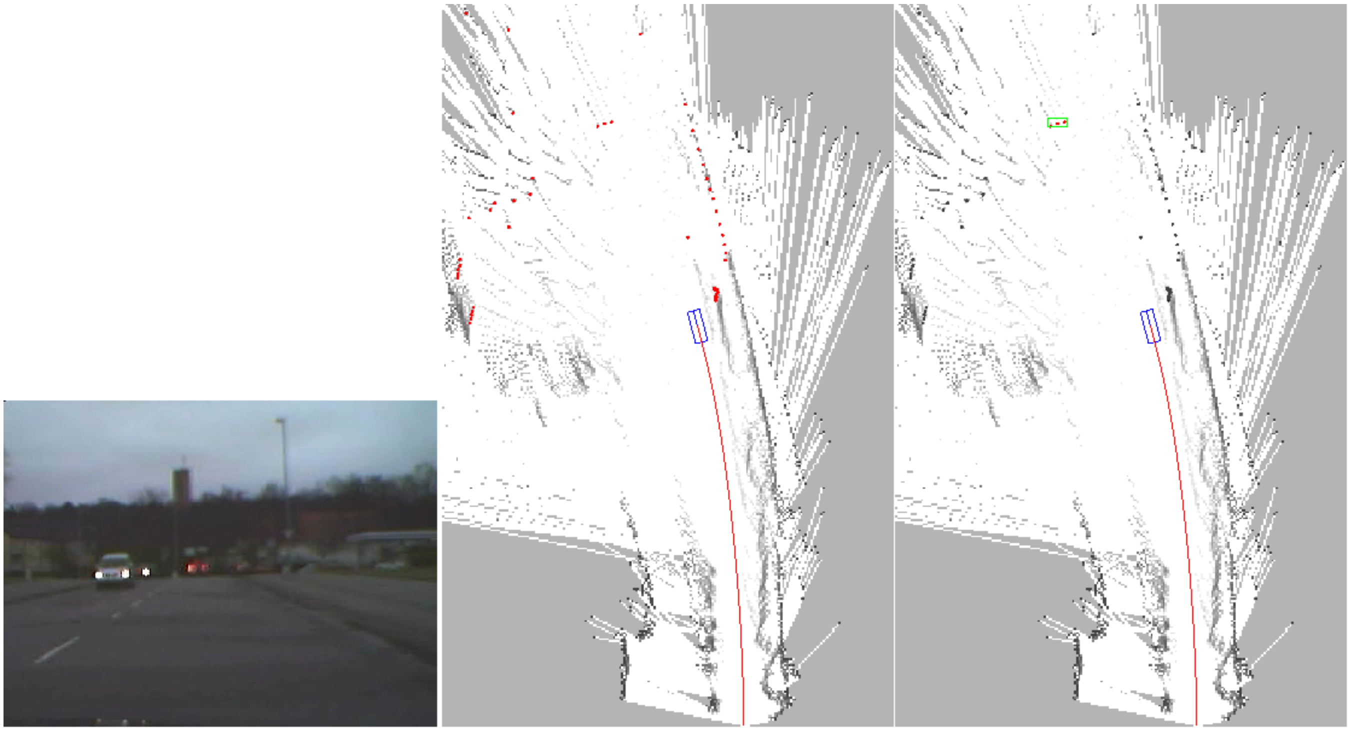

Fig. 6. Moving object detection example.

The second step is after dynamic measurements are determined, moving objects are then identified by clustering end-points of these beams into separate groups, each group represents a single object. Two points are considered as belonging to the same object if the distance between them is less than 0.3 m.

Fig. 6 illustrates the described steps in detecting moving objects. The leftmost image depicts the situation where the vehicle is moving along a street seeing a car moving ahead and a motorbike moving in the opposite direction. The middle image shows the local static map and the vehicle location computed by SLAM and the current laser scan is drawn in red. Measurements which fall into free region in the static map are detected as dynamic and are displayed in the rightmost image. After the clustering step, two moving objects in green boxes are identified and correctly corresponds to the car and the motorbike.

Note that our map updating procedure makes use of results from moving object detection step. Measurements detected as dynamic are not used to update the map in SLAM. For unknown measurements, a priori we will suppose that they are static until latter evidences come. This will help to eliminate spurious objects and result in a better map.

Once we are able to detect moving objects we need to track them in order to estimate their states and predict their behaviour in the future. Tracking multiple moving objects is a classical problem. In the general case this problem is very hard, however it has been shown experimentally that simple methods are good enough to cope with urban scenarios [33]. In our current work, a simple object tracking scheme as described in [2] using Global Nearest Neighbourhood (GNN) and Kalman filter is employed to track detected objects. A replacement using MHT [9] and Adaptive IMM [6] is ongoing.

V. EXPERIMENTAL RESULTS

Our proposed approach for SLAM and DATMO is tested on dataset collected with the INTERSAFE-2 Volkswagen demonstrator car using only laser sensor and odometery. The maximum measurement range of laser sensor is 150 m with a horizontal field of view of $156°$ and a different resolution at different angles with minimum resolution of $1°$ and maximum resolution of $4°$. The vehicle was driven through a scenario which is a typical test case for INTERSAFE-2. The demonstrator vehicle moves straight on a road then it stops to turn left on an intersection, meanwhile two vehicles coming from opposite side cross the demonstrator and a cyclist also moves in front of the vehicle.

In our implementation, the width and height of local grid map are set to 90 m and 108 m respectively, and the grid resolution is set to 30 cm. Every time the vehicle arrives at 40 m from the grid border, a new grid map is created. The local SLAM and DATMO is run for every new laser scan.

The results of local SLAM and DATMO are shown in Fig. 7. The images in the first row represent on-line maps and objects moving in the vicinity of the vehicle are all detected and tracked. The current vehicle location is represented by blue box along with its trajectories after corrected from the odometery. The red points are current laser measurements that are identified as belonging to dynamic objects. The green boxes indicate detected moving objects. The second row has the corresponding images of the scenario.

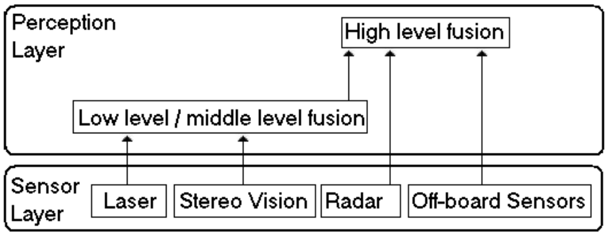

Fig. 8. Sensor Data Fusion Architecture.

In Fig. 7, the first column shows a left turn scenario where the demonstrator car is turning left and a cyclist is going to right in front of the vehicle. This cyclist is detected and tracked. In the second column we have a scenario where a moving vehicle is coming from opposite direction and demonstrator is crossing a vehicle on its right. In both of the cases, precise trajectories of the demonstrator are achieved and local maps around the vehicle are constructed consistently.

VI. DISCUSSION AND FUTURE WORKS

We have presented an approach to accomplish online mapping and moving object tracking simultaneously. Experimental results have shown that our system can successfully perform a real time mapping and moving object tracking from a vehicle at high speeds in different dynamic outdoor scenarios. This is done using a fast scan matching algorithm that allows estimating precise vehicle locations and building a consistent map surrounding of the vehicle. After a consistent local vehicle map is built, moving objects are detected and tracked reliably.

As has been mentioned that this work is related to INTERSAFE-2 project which is about safety on the intersections. The target vehicle as well as infrastructure at the intersections are equipped with many sensors. A key future work is about the fusion of data from these sensors at different levels of abstraction and tracking of moving objects(Fig. 8). We are planning fusion on the following levels:

- Low level fusion with stereo vision. We will use occupancy grid framework for this fusion. Although there are other options available but first we are planning to construct two occupancy grids one for laser and one for stereo camera and use some grid fusion technique to integrate the two grids. As we have constructed occupancy grid for laser another team working on data from stereo vision will construct occupancy grid using elevation maps [20]. One simple fusion technique might be to use $m_i = max(m_i^{laser}, m_i^{cam})$ where m_i^{laser} is a cell of laser occupancy grid and m_i^{cam} is the corresponding cell of stereo camera occupancy gird. Although this a pessimistic map, one cell occupied in one sensor is occupied in final map, but it has advantages on Bayes filter approach when sensors with different characteristics are fused. We will also experiment with other techniques like the ones mentioned in [10].
- Middle level fusion with stereo vision. One other option that we will explore for vision data is the fusion at detection level, object segments detected in vision and laser scans will be fused together to get a more consolidated view.
- High level fusion with radars. At this level objects detected and tracked with laser-vision and radar sensors will be fused together to get more information about the environment.
- High level fusion with infrastructure sensors. Another important aspect of INTERSAFE-2 is that the infrastructure at intersections is also equipped with off board sensors, vehicle will also receive object lists from these off board sensors. So we will also integrate off board sensor information with on-board sensor information to construct a complete view of the environment. These off board information will be especially useful for the areas not visible from the vehicle, so called blind spots, and will reduce the accident in those parts of intersections.

After objects have been detected tracking of the objects is imperative in order to understand and assess the situation around the vehicle. We also plan to use the MHT using Adaptive IMM [30] techniques developed by our group for object tracking.

We also intend to incorporate object models and road models that give a more meaningful representation of detected objects with specific shapes and sizes instead of only sets of contour points as in our current work.

VII. CONCLUSION

We have presented an approach to accomplish on-line mapping and moving object tracking simultaneously. Experimental results have shown that our system can successfully perform a real time mapping and moving object tracking from a vehicle at high speeds in different dynamic outdoor scenarios. This is done based on a fast scan matching algorithm that allows estimating precise vehicle locations and building a consistent map surrounding of the vehicle. After a consistent local vehicle map is build, moving objects are detected and tracked reliably.

Future works include fusion with stereo vision at low and middle levels, with radars and infrastructure sensors at high level.We also intend to incorporate object models and road models that give a more meaningful representation of detected objects with specific shapes and sizes instead of only sets of contour points.

VIII. ACKNOWLEDGEMENTS

The work is supported by the European project INTERSAFE-2[2]

REFERENCES

[1] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filter for online nonlinear/non-gaussian bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2), 2002.

[2] http://www.intersafe-2.eu

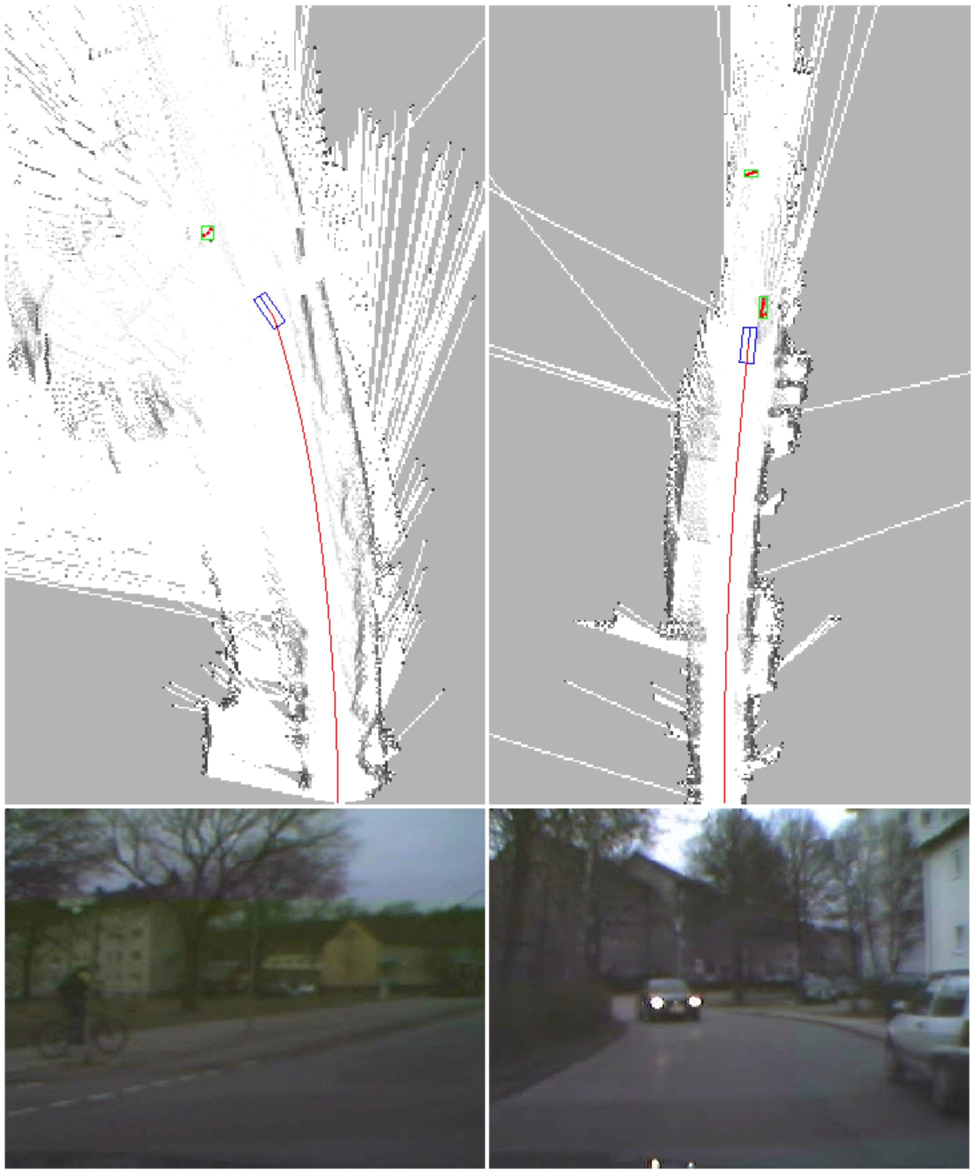

Fig. 7. Experimental results show that the algorithm can successfully perform both SLAM and DATMO in real time for different environments.

[2] O. Aycard, A. Spalanzani, J. Burlet, C. Fulgenzi, T.D. Vu, D. Raulo, and M. Yguel. Pedestrian tracking using offboard cameras. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, October 2006.

[3] Y. Bar-Shalom and T.E. Fortman. *Tracking and Data Association*. Academic Press, 1988.

[4] Y. Bar-Shalom and X. Li. *Multitarget Multisensor Tracking : Principles and Techniques*. YBS Publishing, 1995.

[5] S. S. Blackman. Multiple hypothesis tracking for multiple target tracking. *Aerospace and Electronic Systems Magazine, IEEE*, 19(1):5–18, 2004.

[6] J. Burlet, O. Aycard, A. Spalanzani, and C. Laugier. Adaptive interactive multiple models applied on pedestrian tracking in car parks. In *Proc. of the IEEE-RSJ Int. Conf. on Intelligent Robots and Systems*, Beijing (CN), 2006.

[7] J. Burlet, O. Aycard, A. Spalanzani, and C. Laugier. Pedestrian tracking in car parks: an adaptive interacting multiple model based filtering method. In *Proc. of the IEEE Int. Conf. on Intelligent Transportation Systems*, 2006.

[8] J. Burlet, TD. Vu, and O. Aycard. Grid-based localization and online mapping with moving objects detection and tracking. Technical report, INRIA-UJF, 2007.

[9] I. J. Cox and S. L. Hingorani. An efficient implementation and evaluation of reid's multiple hypothesis tracking algorithm for visual tracking. *Pattern Recognition, 1994. Vol. 1*, 1, 1996.

[10] A. Elfes. *Occupancy grids: a probabilistic framework for robot percpetion and navigation*. PhD thesis, Carnegie Mellon University, 1989.

[11] D. Fox, W. Burgard, and S. Thrun. Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, 11, 1999.

[12] J.S. Franco and E. Boyer. Fusion of multi-view silhouette cues using a space occupancy grid. In *IEEE International Conference on Computer Vision*, 2005.

[13] R. Garcia, O. Aycard, TD. Vu, and M. Ahrholdt. High level sensor data fusion for automotive applications using occupancy grids. In *In IEEE International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2008.

[14] D. Hähnel, D. Schulz, and W. Burgard. Mobile robot mapping in populated environments. *Advanced Robotics*, 17(7):579–598, 2003.

[15] R.E. Kalman. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 35, March 1960.

[16] J. Leonard and H. Durrant-Whyte. Simultaneous map building and localization for an autonomous mobile robot. volume 3, 1991.

[17] F. Lu and E. Milios. Globally consistent range scan alignment for environment mapping. *Autonomous Robots*, 1997.

[18] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan. Interacting multiple model methods in target tracking: a survey. *Aerospace and Electronic Systems, IEEE Transactions on*, 34(1):103–123, 1998.

[19] H.P. Moravec and M.C. Martin. Robot navigation by 3d spatial evidence grids. Technical report, Mobile Robots Laboratory, Robotics Institute, Carnegie Mellon University, 1994.

[20] S. Nedevschi, R. Danescu, T. Marita, F. Oniga, and S. Bota. Stereovision-based sensor for intersection assistance. In *Advanced Microsystems for Automotive Applications 2009*, pages 129–163. Springer Berlin Heidelberg, 2009.

[21] G. Oriolo, G. Ulivi, and Vendittelli M. Fuzzy maps: a new tool for mobile robot perception and planning. *J. of Robotic Systems*, 14(3):179–197, 1997.

[22] D. Pagac, E.M. Nebot, and H. Durrant-Whyte. An evidential approach to map-building for autonomous vehicles. *IEEE Transactions on Robotics and Automation*, 14, 1998.

[23] S. Pietzsch, O. Aycard, J. Burlet, TD. Vu, T. Hackbarth, N. Appenrodt, J. Dickmann, and B. Radig. Results of a precrash application based on laserscanner and short range radars. In *IEEE International Conference on Intelligent Vehicles*, 2008.

[24] E. Prassler, J. Scholz, and P. Fiorini. Navigating a robotic wheelchair in a railway station during rush hour. *Int. Journal on Robotics Research*, 18(7):760–772, 1999.

[25] X. Rong Li and Vesselin P.J. A survey of maneuvering target tracking-part v: Multiple-model methods. *IEEE Transactions on Aerospace and Electronic Systems*, 2003.

[26] S. Thrun. Learning metric-topological maps for indoor mobile robot navigation. *Artificial Intelligence*, 1(99):21–71, 1999.

[27] S. Thrun, W. Burgard, and D. Fox. A real-time algorithm for mobile robot mapping with applications to multi-robot and 3d mapping. In *ICRA*, 2000.

[28] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, September 2005.

[29] TD. Vu, O. Aycard, and N. Appenrodt. Online localization and mapping with moving objects tracking in dynamic outdoor environments. In *IEEE International Conference on Intelligent Vehicles*, 2007.

[30] TD. Vu, J. Burlet, and O. Aycard. Grid-based localization and online mapping with moving objects detection and tracking: new results. In *IEEE International Conference on Intelligent Vehicles*, 2008.

[31] TD. Vu, J. Burlet, and O. Aycard. Mapping of environment, detection and tracking of moving objects using occupancy grids. In *Workshop on Intelligent Transport*, 2008.

[32] Trung-Dung Vu, Olivier Aycard, and Nils Appenrodt. Online Localization and Mapping with Moving Object Tracking in Dynamic Outdoor Environments. In *2007 IEEE Intelligent Vehicles Symposium*, Istanbul Turkey, 2007.

[33] C.-C. Wang. *Simultaneous Localization, Mapping and Moving Object Tracking*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, April 2004.

[34] D.F. Wolf and G.S. Sukhatme. Mobile robot simultaneous localization and mapping in dynamic environments. *Autonomous Robots*, 19, 2005.

[35] B. Yamauchi. A frontier based approach for autonomous exploration. In *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pages 146–151, 1997.