2005 7th International Conference on Information Fusion (FUSION)

EKF and Particle Filter Track to Track Fusion : a quantitative comparison from Radar/Lidar obstacle tracks

Christophe Blanc, Laurent Trassoudaine and Jean Gallice
LASMEA
UMR6602 du CNRS
Campus des Cézeaux
63177 Aubière Cedex
FRANCE
Email: cblanc@lasmea.univ-bpclermont.fr

***Abstract*— In road environment, road obstacles tracking is able to extract important information for driving safety. Indeed, kinematic characteristics estimation (relative position, relative speed, ...) provides a clearer road scene comprehension. This estimate is one of the important parameters of driver assistance systems. However, only one sensor generally does not allow to detect quickly (all the potentially dangerous obstacles) in all the directions, under all the atmospheric conditions. Moreover, the degree of obstacle recognition is different according to the sensor used. Multiplication of sensors makes it possible to face these various problems. Amalgamated information will represent entities in further details and with less uncertainty than with only one source. A system of higher level has been thus developed in order to have a robust management of all tracks and measurements coming from various sensors. This system, applied to Radar and Lidar measurements combination, gives important obstacles characteristics present in the front bumper of our experimental vehicle (VELAC : LASMEA's experimental vehicle for driving assistance). This state estimate is based on the use of various Bayesian methods (Extended Kalman Filter and Particle Filter). Here we will use the fusion of two obstacle tracking delivered by two independent systems : track to track fusion. These two systems propose estimates characterizing obstacles positions and relative speeds. Fusion estimation is based on the use of Extended Kalman filter (EKF) or particle filters. A comparison of these two methods is presented in this article.**

I. Introduction

Only one sensor does not allow to have a representation of the environment (obstacles relative position and speed) precise enough under all the conditions (road, light, night, rain, fog...). Sensors multiplication allows tracking performances improvement i.e. estimate improvement and false tracks elimination. Our work is to track obstacles present in front of our experimental vehicle to feed an ACC[1] system. We have sensors able to operate under different atmospheric conditions, with more or less elaborate attributes and with kinematics precision different. In our system, fusion makes it possible to obtain a track which profits from a more important precision on the state. Moreover, it also makes it possible to eliminate false tracks generated by a sensor whose level of recognition is low.

First, our fusion algorithm requires the use of an architecture managing the data to be treated and send to the alarm system (see [1] for a description of this architecture). Moreover, tracks belonging to the same target must be associated. Indeed, it is a crucial level for track to track fusion process. If associations are defective then the fused estimate can potentially become more unsatisfactory than those of the individual tracks. Lastly, after a reliable association, it is necessary to combine tracks to obtain a more reliable environment interpretation.

Many data fusion methods for tracking exist in the literature and a good description of these methods is given in [2]. The most intuitive method is to fuse measurements of various sensors in a central module [3][4]. If estimates delivered by various local tracking are fused, it is decentralized fusion [5][6][1][7]. Comparisons are presented in [8][9]. Another solution consists in feedbacking the fused estimate on mono sensors tracking for adaptation [10]. The diagram of Figure 1 describes this various architectures.

We will use a method based on track to track fusion without feedback from the fused estimate on mono sensor tracking. Indeed, we seek to develop a general data fusion method of tracking which is easily adaptable to various tracking modules. We do not seek to act on these modules. This method could be used to combine Radar tracks with Lidar ones, or any Radar tracks with IR ones, without modifying the respective tracking modules of the sensors. Each sensor will be seen as a black box. Outputs are state estimate of the obstacles and associated measurements.

Fusion module inputs are thus a set of state vectors and measurement vectors corresponding to tracks. A data association module is necessary to gather the tracks emanating from the same target. This procedure of association carried out, we will use measurements of the various sensors to carry out filtering. Tracks association is described in [1]. Then, we consider here that entries of the fusion module are associated tracks. Filtering, heart of the fusion module, will be based

[1] Adaptive Cruise Control

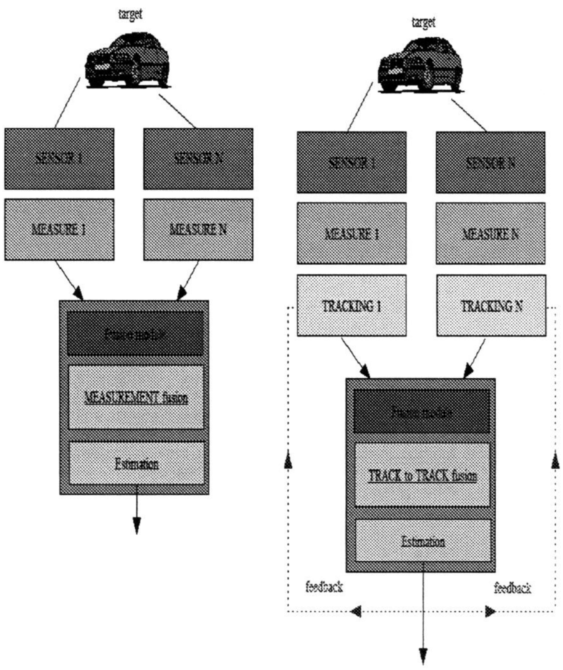


Fig. 1. Fusion architectures

on Kalman filters or particles filters. The validation of fused estimate will be carried out by divergence calculations between the tracks on the sensors levels and the tracks on the fusion module level. We know that tracking is based on the use of disturbed observations and models (state and measurement) for state estimation. Such models have sometimes constraints of non-linearity and noises on states and measurements are sometimes non-gausssian ones. Under these assumptions, the use of the particle filters is perfectly adapted. However, if the noises are supposed to be Gaussian, one will be able to use the Kalman filters if the models are linear and the extended Kalman filters if the models are non linear. Many methods based on such filters exist in the literature [1][11][5][8][12]. Various methods of fusion based on particle filtering are described in [10][13][14][15]. In [13], tracking and fusion modules interact to obtain a more robust face tracking. The fusion module uses the outputs of tracking to build assumptions, and tracking use the output of the fusion module to guide their continuations. In [6], authors compare two fusion architectures to configure their sensors and to estimate the target state. In the first method (known as centralized), the sensors are configured by the adjustment of a foveal gain given by the particles resulting from the resampling. The data fusion is accomplished by using particles weights for each sensor. On the other hand, in the second method (known as decentralized), each sensor uses a particle filter to consider angular measurements and to carry out its own configuration. The centralized method gives better performances for their application. In [14], the authors extend the traditional use of particle filtering, by the estimate of several state processes from achievements of various measurement processes. This algorithm is used to track several targets in a context where measurements are data of direction. In [15], the authors present a multi-target tracking method joint to a sensor management. We present here two estimation methods to compute the fused state. First uses a combination carried out by Kalman filter while the other method uses the particle methods in order to be freed from the linear assumption. Finally, we present some quantitative results coming from the use of two sensors: Radar and Lidar.

II. EKF Track to Track Fusion

Association of two tracks coming from two different sensors must improve knowledge on the track. Thus, we propose to carry out a Kalman filter on the level of fusion module, whose observations are the inputs of local tracking, i.e Z_k^1 and Z_k^2.
Let us consider the target state vector :

$$X^{12} = \begin{pmatrix} x_{12} \\ \dot{x}_{12} \\ y_{12} \\ \dot{y}_{12} \end{pmatrix}$$

where x_{12}, y_{12} are the positions of the track in the reference which is common to both sensors, and $\dot{x}_{12}$, $\dot{y}_{12}$ the track relative speed. Evolution model can be represented in matrix form by :

$$X_{k+1}^{12} = FX_k^{12} + GV_k, V_k \sim \mathrm{N}(0, Q_k) \qquad (1)$$

where F is the transition matrix which modelizes the evolution of X_k, and Q_k the covariance matrix of V_k which represents the acceleration.

$$F = \begin{pmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{pmatrix} G = \begin{pmatrix} T^2/2 & 0 \\ T & 0 \\ 0 & T^2/2 \\ 0 & T \end{pmatrix}$$

$$Q_k = \begin{pmatrix} \sigma_{ax_{12}}^2 & 0 \\ 0 & \sigma_{ay_{12}}^2 \end{pmatrix}$$

The fusion method considered is illustrated on the figure below and was inspired from work of J.B. Gao and C.J. Harris [8].

$\tilde{X}_{k-1/k-1}^{12}$ is the fused state estimate. Its prediction is done by :

$$\hat{X}_{k/k-1}^{12} = F\tilde{X}_{k-1/k-1}^{12}$$

The prediction thus considered is associated with measurements Z_k^1 and Z_k^2 in order to obtain two estimated $\tilde{X}_{k/k}^1$ and $\tilde{X}_{k/k}^2$ by using local Kalman filters. Measurements equations are done by using the model described below :

$$\hat{Z}_{k/k-1}^i = h^i(\hat{X}_{k/k-1}^{12}) + W_k^i, W_k^i \sim \mathrm{N}(0, R_k^i)$$

where $Z_{k/k-1}^i$ is the predicted measurement vector, h^i the function which relates the state to the measurement, and R_k^i

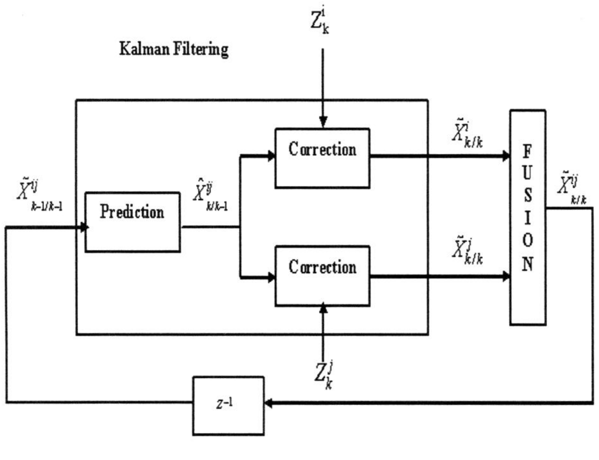


Fig. 2. kalman fusion module

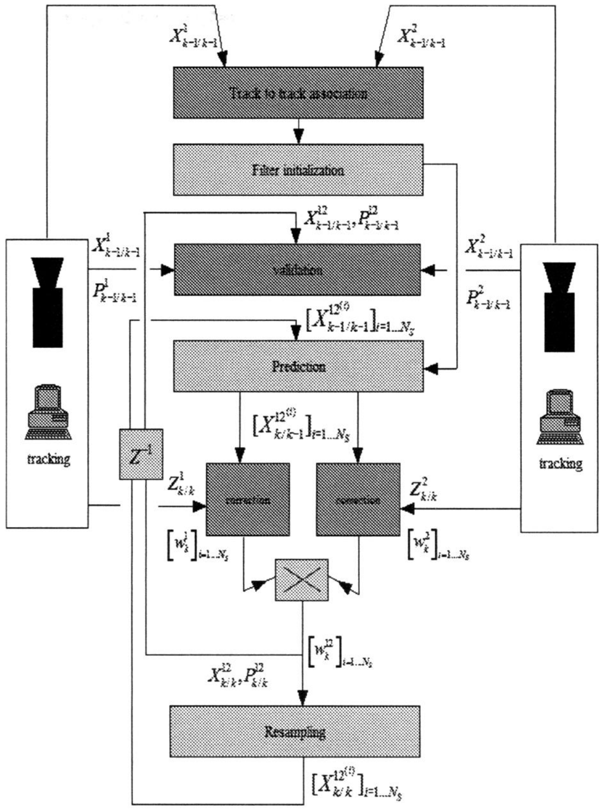


Fig. 3. Particle track to track fusion architecture

the covariance matrix of W_k^i for sensor i. If h^i is not linear an Extended Kalman Filter is used.
For each sensor the estimated state vector is classically computed by :

$$\tilde{X}^i_{k/k} = \hat{X}^i_{k/k-1} + K^i_k[Z^i_k - \hat{Z}^i_{k/k-1}]$$

where K^i_k is the Kalman gain of sensor i filter.

Finally, the new fused state estimation is done by :
$\tilde{X}^{12}_{k/k} = \tilde{X}^1_{k/k} + [\tilde{P}^1_{k/k} - \tilde{P}^{12}_{k/k}][\tilde{P}^1_{k/k} + \tilde{P}^2_{k/k} - \tilde{P}^{12}_{k/k} - \tilde{P}^{21}_{k/k}]^{-1}(\tilde{X}^1_{k/k} - \tilde{X}^2_{k/k})$ where $\tilde{P}^1_{k/k}$ and $\tilde{P}^2_{k/k}$ are the associated covariance matrix of $\tilde{X}^1_{k/k}$ and $\tilde{X}^2_{k/k}$ estimates.
$\tilde{P}^{12}_{k/k} = (\tilde{P}^{21}_{k/k})^t$ is the cross-covariance matrix of $\tilde{X}^1_{k/k}$ and $\tilde{X}^2_{k/k}$. The covariance matrix of the fused estimate is computed by :
$\tilde{P}'^{12}_{k/k} = \tilde{P}^1_{k/k} - [\tilde{P}^1_{k/k} - \tilde{P}^{12}_{k/k}][\tilde{P}^1_{k/k} + \tilde{P}^2_{k/k} - \tilde{P}^{12}_{k/k} - \tilde{P}^{21}_{k/k}]^{-1}[\tilde{P}^1_{k/k} - \tilde{P}^2_{k/k}]^t$

III. Particle Filter Track to Track Fusion

Originally developed in the tracking community [16], particle filtering currently knows a strong development in many research fields (vision, localization, navigation, robotics...), in particular in multi-target tracking.
This filter is a sequential Monte-Carlo method in which particles traverse the state space in an independent way, and interact under the effect of a probability function which automatically concentrates the particles in the state space areas of interest. This method has the advantage it does not require linear or Gaussian assumptions on the model. Moreover, it is very easy to implement, since it is enough to know how to simulate independent various trajectories of the model.
Our fusion method is based on calculation of particle weights from different sensors measurement and on a validation from the fused estimate by the estimates of different tracking modules. It is illustrated figure 3 for a two tracks fusion.

We propose here, to carry out a particle filtering on the fusion module level, whose observations are the inputs of external tracking, i.e $Z^1_{k/k}$ et $Z^2_{k/k}$.
When two tracks are associated, a fused state vector is initialized. We will be able to take the average of the two independent tracking estimates if the respective state vectors of the two sensors are estimated in the same reference. Moreover, we will be able to use part of the state provided by a sensor and/or the other to supplement the fused state vector. From this vector, a set of N_s particles is built. Noise particles are generated ($B_O^{(i)} i = 1, .., N_s$) and applied to the initial vector :

$$X^{12^{(i)}}_{0/0} = X^{12}_{0/0} + B^{(i)}_0, \forall i \in [1...N_S] \quad (2)$$

Then, the model defined in (1) is applied to various particles. It is the particles prediction:

$$X^{12^{(i)}}_{k/k-1} = FX^{12^{(i)}}_{k-1/k-1} + GV^{(i)}_{k-1}$$

Correction is carried out by computing the weights. For each sensor, we calculate N_s weights assigned to the N_s predicted particles.
We have :

$$w_k^{1^{(i)}} = p(Z^1_{k/k}/X^{12^{(i)}}_{k/k-1}) = p(Z^1_{k/k} - h^1_k(X^{12^{(i)}}_{k/k-1}))$$
$$w_k^{2^{(i)}} = p(Z^2_{k/k}/X^{12^{(i)}}_{k/k-1}) = p(Z^2_{k/k} - h^2_k(X^{12^{(i)}}_{k/k-1}))$$

$p(Z^{12}_{k/k}/X^{12(i)}_{k/k-1}) = p(Z^1_{k/k}, Z^2_{k/k}/X^{12(i)}_{k/k-1}) = p(Z^1_{k/k}/X^{12(i)}_{k/k-1}).p(Z^2_{k/k}/X^{12(i)}_{k/k-1})$ thus

$$w_k^{12^{(i)}} = w_k^{1^{(i)}} * w_k^{2^{(i)}}$$

Weights are then normalized :

$$w_k^{\prime 12^{(i)}} = \frac{w_k^{12^{(i)}}}{\sum_{i=1}^{N_S} w_k^{12^{(i)}}}$$

Finally the fused state considered is given by:

$$X^{12}_{k/k} = \sum_{i=1}^{N_S} w_k^{\prime 12^{(i)}} . X^{12^{(i)}}_{k/k-1}$$

and its covariance by :

$$P^{12}_{k/k} = \sum_{i=1}^{N_S} w_k^{\prime 12^{(i)}} (X^{12^{(i)}}_{k/k} - X^{12}_{k/k})(X^{12^{(i)}}_{k/k} - X^{12}_{k/k})^T$$

Particles are resampled and turned over at the stage of prediction if the validation take place. The method of validation is based on a statistical test [1].

IV. Application : Quantitative Comparison from Radar/Lidar Road Obstacles Tracks

A. Radar specifications and tracking results

The key interests to use a Radar in this project are on the one hand the accuracy of the obstacle speed estimate and on the other hand the quality of its informations up to 150 m in spite of difficult weather conditions.
T

1) Treatment of the Radar data: The objective of this step is to determine the distance and the relative speed of the objects (or obstacles) located in the enlightened space by the Radar beam. The reader can refer to [17] for many details on the radar data processing. Every 8 ms the radar delivers a measurement of time, amplitude, range and an index speed for all echoes. The range gate is $\delta R = 22.5$ m and an index speed corresponds to a speed of $\delta_\nu = 0.238$ m/s. In radar measurements, one target can generate several echoes in close range gate as well as neighbor speed samples. A pretreatment is thus necessary in order to gather the echoes emanating from the same target. In a second step, a measurement vector $Z = \begin{pmatrix} r_m \\ V_{r_m} \end{pmatrix}$ and its covariance matrix $C = \begin{pmatrix} \sigma^2_{r_m} & 0 \\ 0 & \sigma^2_{V_{r_m}} \end{pmatrix}$ are associated to each resulting target.

2) Radar tracking results: the Radar tracking is based on Kalman filter and yields to a more accurate range estimate than the gate value (22.5 m)(see figures 4, 5).

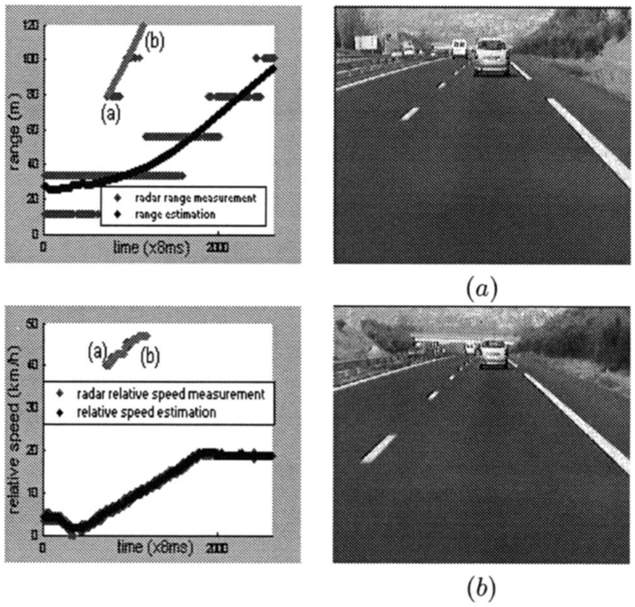


Fig. 4. two obstacles

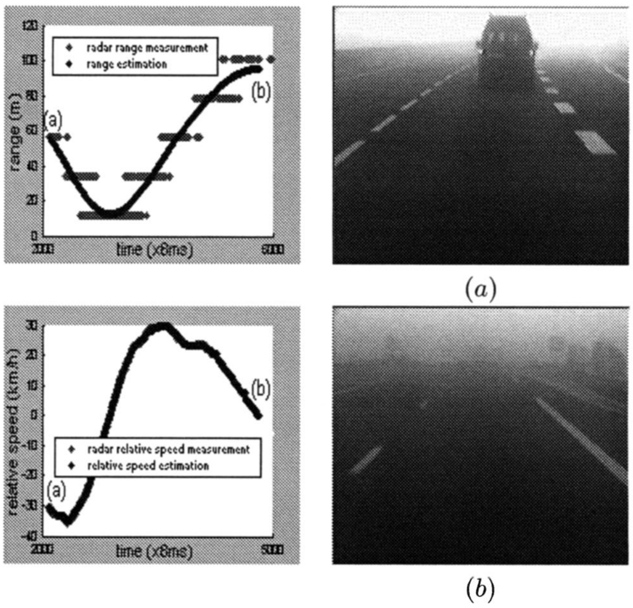


Fig. 5. Obstacle in foggy conditions

B. Lidar specifications and tracking results

The 3D-Laser Mirror Scanner LMS-Z210-60 is a surface imaging system based upon accurate distance measurement by means of electro-optical range measurement and a two axis beam scanning mechanism.
The range finder system is based upon the principle of time-of-flight measurement of short laser pulses in the infrared wavelength region. Many methods for time-of-flight's calculation are described in [18]. The task of the scanner mechanism is to direct the laser beam for range measurement in a accurately defined position. The 3D images are configurable. In our approach 20 lines x 103 pixels images at nearly 2 Hz are

used (see figure 13). After detection of different obstacles, we are able to track them in consecutive frames using a constant velocity Kalman filter and a nearest neighbor standard data association method.
Some results are presented in figures 6, 7, 8. On these figures, relative positions and speed of different tracks are represented with different colors. The figure 9 allows to represent several moments of the sequence. These results show the multi-target tracking system capacity in a motorway context. It is noticed that we are able to detect and track several types of obstacles (cars and trucks). The data association system based on research of nearest neighbor seems sufficient for this system. Indeed, it is noticed here that it does not appear any false track. It is thus not necessary to use methods of type JPDAF or with multiple assumptions. Moreover, the precision of the Lidar measures allows data association to easily integrate observation which corresponds best to the considered track. We will be able, for example, to use the obstacle size as one of the criteria of associations if several measurements fall into the validation window. The advantage of this method is based on the measurements precision delivered by the lidar and on a high detection probability. Moreover, in a road context, the number of tracks to follow in front of our experimental vehicle is weak. That reduces considerably necessary calculations to the data association systems.

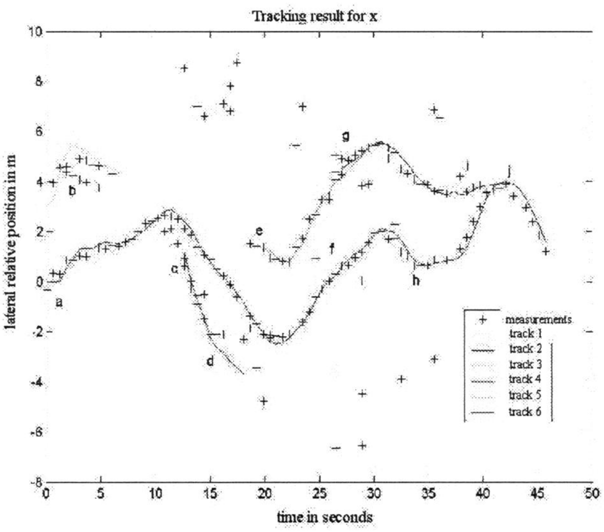


Fig. 6. Lidar tracking results for x in case of multi tracks

C. Track to track fusion results

Given the characteristics of our sensor radar we can affirm that radar data are complementary with all the other data. Indeed, the Radar is insensitive to atmospheric conditions, it is thus judicious even essential to use a sensor of this type for obstacle detection in a road environment.
We present here various results of track to track fusion. Trackings are independent systems obtained by Lidar and Radar. All the presented results are obtained from data gathered under

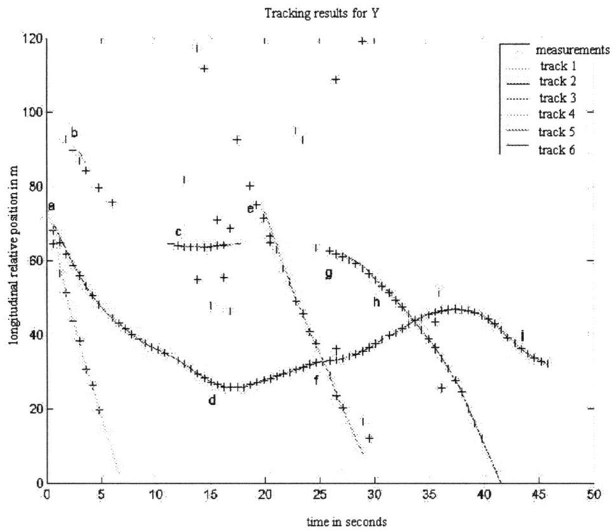


Fig. 7. Lidar tracking results for y in case of multi tracks

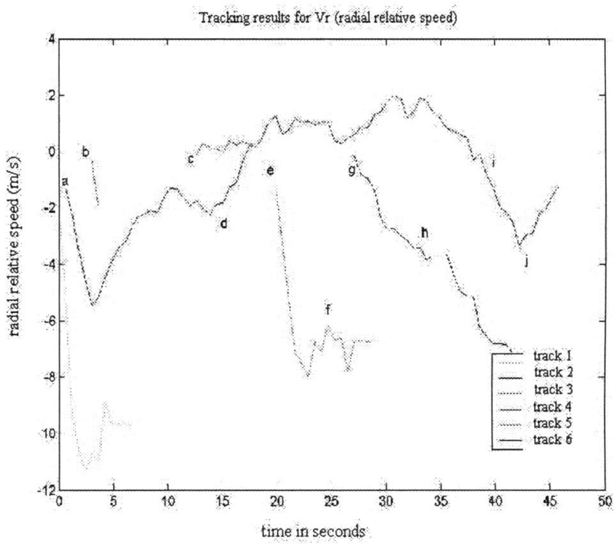


Fig. 8. Lidar tracking results for V_r in case of multi tracks

real conditions of traffic. We will show in this results the fusion contribution for a robust system of obstacles monitoring.
We choose to use the Lidar observations to fuse with the radar data since they are precise in position on the contrary of radar which is precise for relative speed.
For the radar, an observation or a track is characterized by a distance and a radial speed $(r, \dot{r})$ and by the covariance matrix of these measures. For the laser range finder, an observation is an object characterized by a position (x, y) measured in the sensor reference frame and by the covariance on these position. The state vector of the track is extended to relative speed in both directions.
Lidar and radar data are not given in the same reference. We thus choose the reference related to the radar for fusion. The

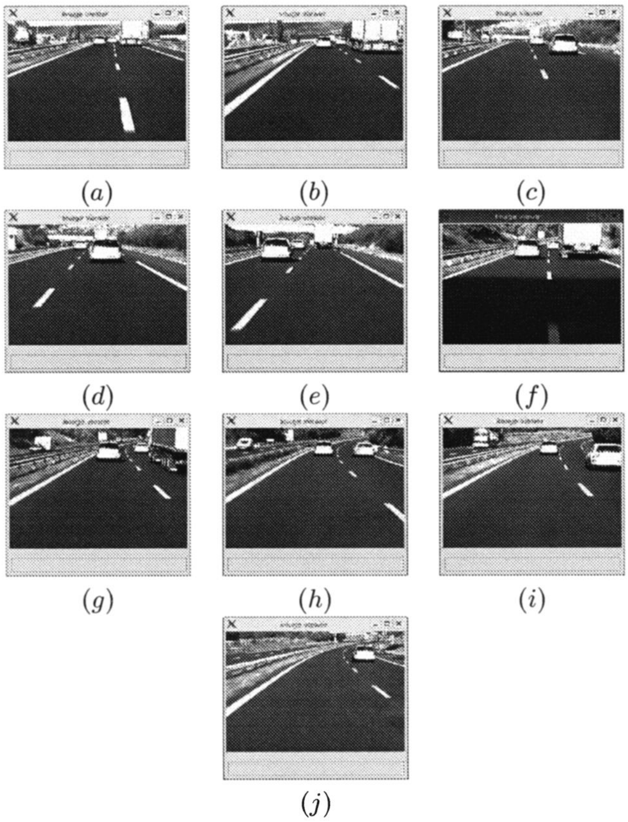


Fig. 9. Times a, b, c, d, e, f, g, h, i and j of the sequence

lidar observations thus undergo a transformation before input in the fusion module. Areas covered by the sensors are also defined in the fusion reference frame (see figure 10). The

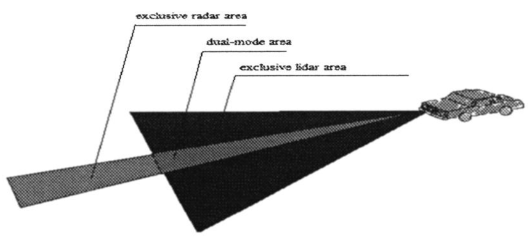


Fig. 10. Sensor areas

exclusive range finder area is the zone covered by the only telemetric sensor. The exclusive Radar area extends beyond the lidar theoretical range (90 meters). Fusion is efficient in the so called dual mode area which is covered by both sensors. The data dating system is based on the use of the bus IEEE 1394 clock, which has the characteristic to give a common base time to all the connected PCs.

1) Multi target tracking: We illustrate results of track to track fusion for road scenes interpretation, and show benefit compared to the results obtained by mono sensor processing. Figure 11 illustrates a simple road situation. We see two vehicles in the image. The tracks produced by sensors are deferred in figures 11.a and 11.b. Relative positions of detected obstacles are symbolized by a green rectangle for lidar obstacles and by red line for radar detection. Fused position estimate is represented by a yellow circle. With only radar data, we can't say if obstacle is on the right or left side of our experimental vehicle. The joint use of lidar data allows this side positioning. The two other figures (12.c 12.d) show the fused position estimation for both filters. For the position estimation, we can see that the choice of the method is not influential. In other road scenarios, radar measurements can generate false tracks [1]. These radar measurements result in general from fixed objects present in road environment.

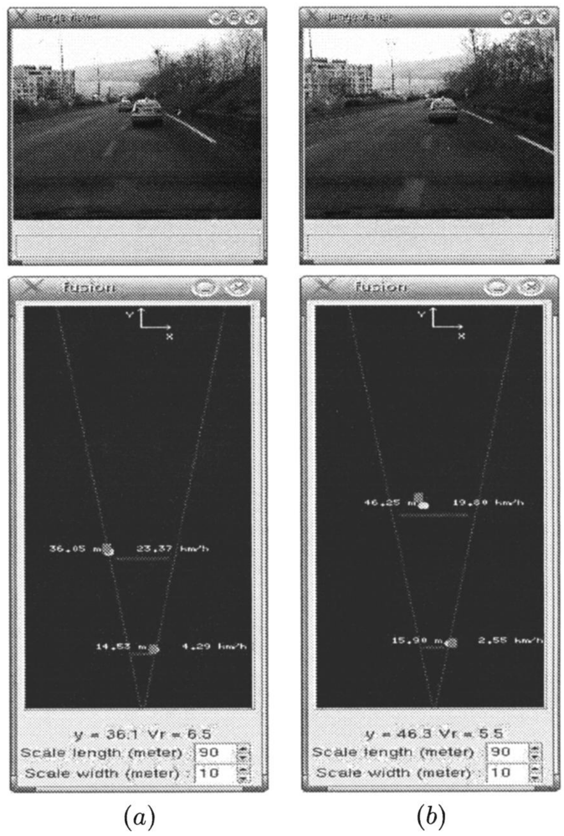
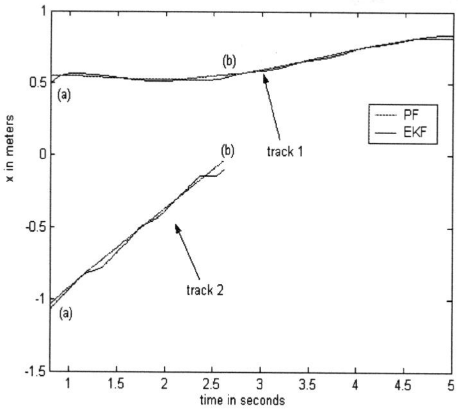


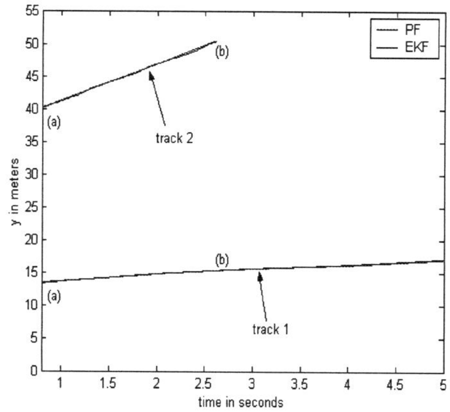
Fig. 11. Simple road obstacle situation

2) False tracks elimination: Generally, a sensor can regard a target as being a dangerous object because of its kinematic coherence. For example, radar which recognition degree is almost zero does not make it possible to distinguish the traffic signs from the fixed vehicles. Figure 13 shows the false radar track elimination. Indeed, we can see that radar detects a traffic sign in the dual mode area, but lidar does not confirm it, so it is discarded, therefore avoiding a false obstacle. Symmetric situations can also happen but they are rare since the degree of vehicle recognition is large for the lidar.

3) Mono target tracking quantitative results: quantitative results are obtained with a ground truth. Velac and only one obstacle are equipped with DGPS. Their location are acquired every second. Obstacle position is sent to Velac by MF communication. In the same time, the obstacle is tracked



(c)



(d)

Fig. 12. Multi target tracking results

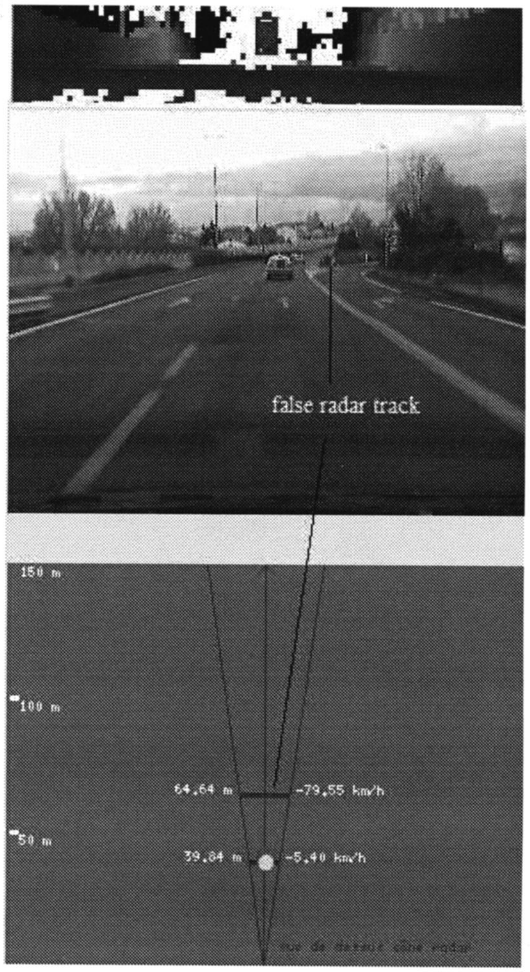


Fig. 13. False track elimination

by both processes (Radar/Lidar). Track to track fusion process and comparison are done offline. The measurement system include a differential GPS Omnistar which delivers trames with format TSIP (Trimble Standard Interface Protocol). This GPS gives, in the best configuration, a position with a ± 40 cm accuracy. It delivers a coefficient, called gdop : the current accuracy is gdop times 40 cm. DGPS errors are shown in figure 14. Moreover, for data communication, a radio modem of Satel receives trames coming from the obstacle. Figure 14 show the results for the range estimation by Extended Kalman filter and particle filter. Moreover, it shows the radar and lidar estimate. We see that radar estimate is less accurate than lidar estimate in this particular scenario. The dgps reference allows to compute root mean square error (rmse) and its standard deviation (std) for both filters . As it is shown on results, performance of range estimation is correct for both filters. Errors and DGPS accuracy have almost the same order. Computation times are shown in figure 15. As expected, the EKF have very small computation time compared with the particle filters. Times are computed on a Pentium 4 M 1.7 Ghz. Only EKF allows real time utilization because of the radar data rate wich is 8 ms.

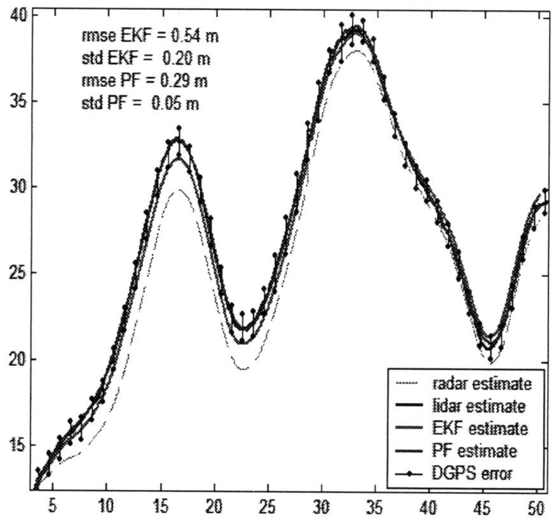


Fig. 14. Mono target tracking results

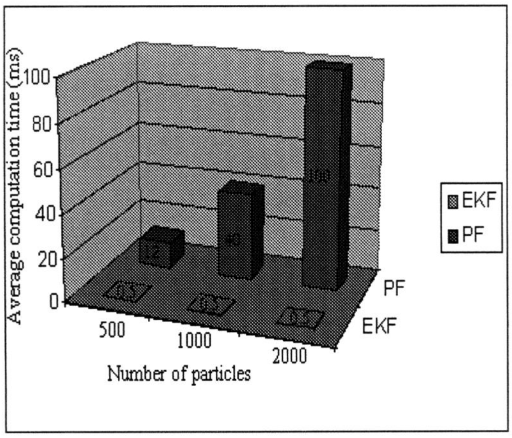


Fig. 15. Performance comparison of EKF and PF : computation time versus number of particles

V. Conclusion

We have shown the interest of the track to track fusion. Indeed, in our case, one goal of fusion was to thwart the error of radar estimation. Moreover, the second goal was to eliminate false tracks. Finally, fusion makes it possible to have a satisfying state sampling.
Concerning the radar, it is not possible to extract from its measurements the lateral characteristics of the detected obstacle. The lidar does not deliver relative velocity measurements but only one estimate every 500 ms. An obstacle, with a relative speed v = 130 km/h, traverses 18 m in 500 ms. It is thus not possible to make robust decisions as for the driver alarm by only using lidar sensor. In addition, the specific sensor tracking system can sometimes deliver false tracks able to generate threats and thus to generate false alarm to alert the driver.
All these remarks pushed us to use a system of fusion making in a robust way and specifies the obstacles of dangerous or not. This system is based on the extraction of maximum possible information on the environment lighted by the sensors under many conditions. Indeed, this process of fusion makes it possible in a first point to eliminate false tracks appearing in the area covered by the two sensors. Moreover, this system allows an estimate of the essential characteristics of an obstacle. This estimate, based on the use of EKF or particle methods, delivers all essential characteristics. The results show that the EKF and the particle filter have similar performances in most situations. Moreover, since the EKF has a much less computational load than the particle filter, it is the best choice for our real time application.

Acknowledgment

This work takes place within the TIMS Research Group Program, granted by the Regional Council of Auvergne, the French Ministry of Research, the CNRS and the Cemagref.

References

[1] C. Blanc, L. Trassoudaine, Y. LeGuilloux, and R. Moreira. Track to track fusion applied to road obstacle detection. In *ISIF 2004(7)*, Stockholm (Sweden), June 28–1st july 2004.
[2] A. Gad, M. Farooq, J. Serdula, and D. Peters. Multitarget tracking in a multisensor multiplatform environment. In *ISIF 2004*, Stockholm (Sweden), June 28–1st july 2004.
[3] S. Hun, Z. Jing, and H. Leung. Optimal fusion algorithm based on multi-sensor tracking. In *ISIF 2004*, Stockholm (Sweden), June 28–1st july 2004.
[4] C. Hue, J.P. LeCadre, and P. Pérez. The (mr)mtpf: particle filters to track multiple targets using multiple receivers. In *ISIF 2001*, Montréal (Canada), 2001.
[5] S. Sun and Z. Deng. Multi-sensor optimal information fusion kalman filter. In *Automatica*, volume 40, pages 1017–1023, june 2004.
[6] S. Coraluppi and C. Carthel. Recursive track fusion for multi-sensor surveillance. In *Information Fusion*, volume 5 issue 1, pages 23–33, march 2004.
[7] Y. Bar-Shalom et T. Kirubarajan X. Lin. Multisensor-multitarget bias estimation for general asynchronous sensors. In *ISIF 2004*, Stockholm (Sweden), June 28–1st july 2004.
[8] J.B. Gao and C.J. Harris. Some remarks on kalman filters for the multisensor fusion. In *Information Fusion*, volume 3, issue 3, pages 191–201, september 2002.
[9] H. Chen, T. Kirubarajan, and Y. Bar-Shalom. Performance limits of track-to-track fusion vs. centralized estimation : theory and application. In *IEEE transactions on Aerospace and Electronics Systems*, volume 39, issue 2, pages 386–400, april 2003.
[10] Y. Xue and D. Morrell. Target tracking and data fusion using multiple adaptive foveal sensors. In *ISIF 2003(6)*, Cairns (Australia), july 2003.
[11] C. Stiller, J. Hipp, C. Rossig, and A. Ewald. Multisensor obstacle detection and tracking. In *Image and Vision Computing*, volume 18, pages 389–396, 2000.
[12] Y. Zhu, Z. You, J. Zhao, K. Zhang, and X. Rong Li. The optimality for the distributed kalman filtering fusion with feedback. In *Automatica*, volume 47, pages 1489–1493, march 2001.
[13] Y. Chen and Y. Rui. Real-time speaker tracking using particle filter sensor fusion. In *Proceedings of the IEEE*, volume 92 issue 3, pages 485–494, march 2004.
[14] C. Hue, J.P. Le Cadre, and P. Pérez. Sequential monte carlo methods for multiple target tracking and data fusion. In *IEEE Transactions on Signal Processing*, volume 50 issue 2, pages 309–325, february 2002.
[15] A. Doucet, B.N. Vo, C. Andrieu, and M. Davy. Particle filtering for multi-target tracking and sensor management. In *ISIF 2002(5)*, Maryland (USA), 2002.
[16] N.J. Gordon, D.J. Salmond, and A.F.M. Smith. Novel approach to nonlinear-non-gaussian bayesian state estimation. *IEE Proc.-F*, vol. 140, no. 2:p. 107–113, 1993.
[17] C. Blanc, R. Aufrère, L. Malaterre, J. Gallice, and J. Alizon. Obstacle detection and tracking by millimeter wave radar. In *IAV 2004*, Lisboa (Portugal), july 2004.
[18] J. Hancock. *Laser intensity-based obstacle detection and tracking*. PhD thesis, The robotics institute, Carnegie Mellon University, Pittsburgh, USA, 1999.