Merging lateral cameras information with proprioceptive sensors in vehicle location gives centimetric precision.

Sio-Song Ieng and Dominique Gruyer
LIVIC - INRETS/LCPC, 13 Route de la Miniere, Batiment 140,
78000 Versailles Satory, France
ieng@inrets.fr, dominique.gruyer@inrets.fr
Paper 459

Abstract

Our work is dealing with lane markings detection and the vehicle location. We will show how computer vision can improve the accuracy of the determination of the vehicle position in a map by GPS and proprioceptive sensors. An efficient method for locating vehicle by cameras, proprioceptive sensors and GPS has been developed and demonstrated in an outdoor experimental track in real time. The system is designed to a well structured road with lane markings. It merges proprioceptive measurement, GPS location and images analysis information with use of a non linear dynamic model(Kalman Filter). The performance of the system is shown in the experimental track with a processing frequency of 15 Hertz and the error of the location is $\pm 5cm$.

1 Introduction

To locate the vehicle is a key level of every advanced driving assistant systems that are designed for many purposes:

- provide users with vehicle geographic position,
- help users with adapted driving instruction,
- provide users with a full or partial autoguidance possibility.

The large amount of sensors or systems that can help in vehicle location encourages researchers to investigate more in such a location systems. The widely used Global Positioning System(GPS) gives location in an absolute coordinate system(Lambert's coordinate system). GPS can be used in several different modes that provide different kind of accuracy. The higher accuracy is reached with Real Time Kinematics mode (RTK). But many conditions can affect the result.(SA effect, high buildings or trees) GPS depends strongly on the external condition. Proprioceptive sensors only measure variable that depends on the vehicle's engine (speed, acceleration, rotation angle). To estimate the position of the vehicle (the position (x,y) and the vehicle course θ, we must integrate measurement at each moment. This implies an important accumulation of error on the result. To estimate locale vehicle position by using exteroceptive sensors, many algorithms were presented by using computer vision [2], [3], [4]. Most of them, are using cameras looking forward. Such a system's performance depends on many kind of perturbations (light, shadows, occlusions). A serious study must be done to design a robust algorithm [8]. To robustify the vehicle location system, Fusion methods are used [5]. Here, we will propose to merge the measurement provided by these systems. The presented system will be able to take into account advantages and drawbacks of each kind of sensor. In the first part, we will describe an algorithm that uses cameras positioned laterally at each side of the vehicle for line detection. In the second part, a non linear Kalman based algorithm that merges information from different sensors (GPS, INS, GPS and Map-Matching, image processing...) is presented. This allows us in the final part to compare the different solutions and show how image analysis algorithms can help and improve the vehicle location with other sensors.

2 The lateral camera system and lane-marking detection

The system needs two cameras positioned in each side of the vehicle looking at the region of the road near the vehicle.(Figure 1 and 4).

Figure 1: The cameras are placed outside of the vehicle.

Both images analysis and the sensor fusion method are implemented in an embedded bi-processor computer as shown in figure 2.

Figure 2: The on board bi-processor computer in the LIVIC vehicle.

2.1 Cameras position and associated coordinates systems.

To determine the local position of the lane marking, we need to introduce first some coordinate systems. For the image description we need an image coordinates system (u, v).(Fig 3) For each camera, we have a coordinate system for the description of the detected lane marking of each side of the vehicle.(Fig 4)

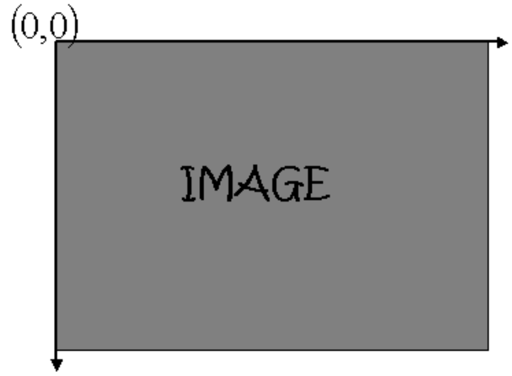

Figure 3: The image coordinate system.

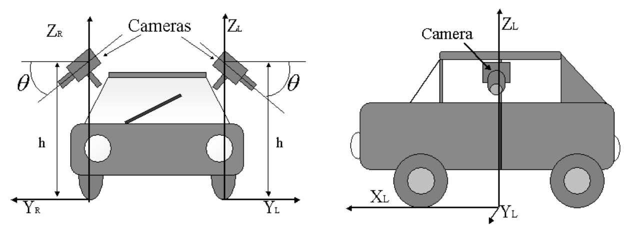

Figure 4: The coordinates systems associated to lateral cameras.

3 The lane-marking detection

3.1 algorithm of features extraction

This extractor uses two main characteristics of the lane marking:

- the high intensity of the pixels belonging to the lane markings,
- the width of the lane markings which is constant.

The main idea of this algorithm is to take into account the lane marking's width. With the perspective, this lane marking's width in the image decreases when the distance between vehicle and the marking increases. Actually, we can show that the markings width decreases linearly to zeros when it reaches the horizon line in the image.(Figure 5)

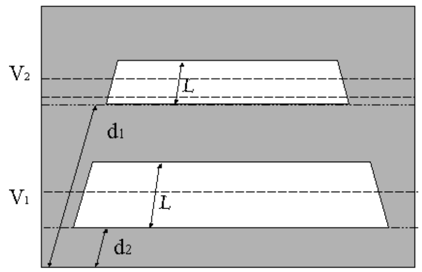

Figure 5: For the same kind of lane markings, the width of their images depend on the distance of the marking to the camera. In the picture, $d1 > d2$ and v_1 and v_2 are the coordinates of the centers of the two marking.

First, the extractor computes intensity gradients of a value higher than S_0 and then searches for a pair of positive and negative gradients within a range $[S_1, S_2]$. The goal is to obtain the maximum number of features really on the lane-markings and at the same time to reduce as much as possible the number of outliers, knowing that in any case the problem of outliers is tackled by the robust fitting algorithm of lane-markings. In order to not miss any feature, even in adverse lighting conditions, we have to set S_0 as small as possible and to analyze the whole image to initialize the detection. It is always possible to limit the analysis in areas of interest thanks to a dynamic shape tracking. Fortunately, the proposed extractor is fast enough to be applied on whole image. For each line image, let u_{init} be the first position for which a gradient is greater than the threshold S_0, $u_{current}$ is the position of the current analyzed pixel. A lane-marking feature is considered to be detected in the image line if $u_{current}$ - u_{init} is within the range $[S_1, S_2]$ where $S_1 = C_1(v - v_h)$ and $S_2 = C_2(v - v_h)$. We can notice that S_1 and S_2 vary when we modify the coordinate v. S_1 and S_2 are very important for removing many of the outliers, and thus C_1 and C_2 have to be chosen carefully. C_1 and C_2 can take into account different kind of errors such as small variations of marking width, errors on camera calibration. Here is the algorithm:

1. Calculate gradient $G(pixInit)$
2. If $G(pixInit) > S_0$ then
   - $pixCourant = pixInit + 1$;
   - $I = Intensity(pixInit) + \frac{G(pixInit)}{2}$;
   - While $Intensity(pixCourant) > I$ and $pixCourant < sizeXImage$ $pixCourant = pixCourant + 1$;
   - If $pixCourant \in [S_1, S_2]$ Then a marker is detected.
   - Return the centre of the marker axis: $\frac{(pixCourant+pixInit)}{2}$
   - $pixInit = pixCourant + 1$, return to 1,
   - else, $pixInit = pixInit + 1$, return to 1,
3. else, $pixInit = pixInit + 1$, return to 1.

3.2 Features extraction result

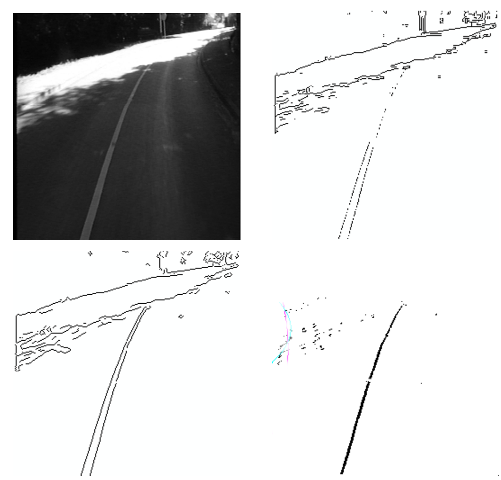

Figure 6: Top left: Original image of a road. Top right: result of Prewitt's filter. Bottom left: result of Canny filter. Bottom right: result with our lane-marking features extractor. It is only with the proposed extractor that the high light area is removed.

The classical edge filters detect lane markings but also every shadow or high light edges. By taking into account the marking's width, only a few outliers due to the high light area are detected. Moreover the lane-marking is completely extracted.(Fig 6)

3.3 The choice of lane-marking model

Many lane markings and road model were proposed. When the camera is looking forward, the road shape is very important, 2D curves or 3D curves are then useful. The drawback of this kind of model is the lack of accuracy on the distance estimation and the system is very sensitive to light variation. But in our case, only a very local part of the lane marking is seen. In this scale, The detected feature contains less noise and the effect of light variation is weak. We can assume that, locally, lane marking is a straight line.
Let us write the equation of a straight line in the moving road coordinate system (Fig 4):

$$y = Sx + D \tag{1}$$

If we set $x = 0$, D represents the distance between vehicle and the lane marking. The slope S represents the tangent of the angle ϕ.(Fig 8) To estimate these two values, ϕ and

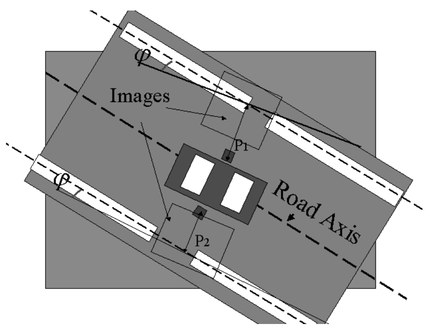

Figure 7: Each time, the system analyses two images of lane marking. phi represents the relative course angle of the vehicle, P_1 and P_2 are distances of the lane markings to the vehicle.

P we can use a hough transform algorithm but we have developed an iterative reweighted least square algorithm presented in [8] that can estimate more complex curves.

3.4 The lane markings position estimation result.

We first, show some results of the ϕ and P estimation with a video sequence. The detection is presented by straight lines.

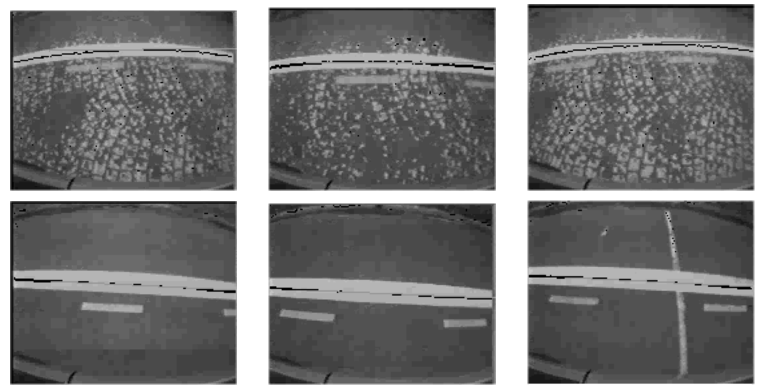

Figure 8: Examples of a video sequence taken during a test of the system. Dark lines represent the center of detected lane marking.

4 Vehicle location system.

4.1 Kalman based algorithm that merges sensors measurements.

The dynamic system that handles sensors measurement is a non linear system. It is used in a extented Kalman filter. Let us first introduce some notations:

- $X_k = (x, y, \theta)$ system's state vector at the moment k,
- $X_{k+1/k}$ is the predicted vector at the moment $k+1$ knowing the estimation at the moment k,
- U_k command vector at the moment k,
- V_k system state noise,
- Y_k measurement vector at the moment k,
- W_k measurement noise at the moment k,
- $f(X_k, U_k)$ represents the non linear evolution of the system,
- $h(X_k)$ represents the estimation process,
- $P_{k+1/k}$ is the covariance matrix on the prediction step for $X_{k+1/k}$,

- $P_{k/k}$ is the covariance matrix on the estimation step,
- R_k is the covariance matrix on the state,
- Q_k is the covariance matrix on the measurement error.

The Kalman filter is defined by the following equations system.

$$\begin{cases} X_{k+1} = f(X_k, U_k) + V_k & \text{prediction equation} \\ Y_k = h(X_k) + W_k & \text{estimation equation} \end{cases} \quad (2)$$

The prediction equation describes the theoretic system model. In our case, the system is a vehicle. The estimation equation depends on the sensors used. This can represent vehicle position or any function depending on it. In the following section, we will give details about sensors and the appropriate functions f and h. Here is the algorithm:

1. initialisation step: the Kalman filter algorithm is a recursive algorithm that needs initialisation. In our tests, we will initialise the algorithm with GPS. Assume here that the initial state is $X_0/0$ and its covariance matrix $P_{0/0}$.
2. prediction step: the prediction for the moment $k+1$ is computed by the prediction equation: $X_{k+1/k} = f(X_{k/k}, U_k)$ The perturbation is taken into account in the calculation of the covariance matrix. To compute this matrix, we must first calculate the f's gradient $\nabla f(X_{k/k}) = F_{k/k}$ because the model is non linear. The covariance matrix is then defined by

$$P_{k+1/k} = F_{k/k} P_{k/k} F_{k/k}^t + R_k \quad (3)$$

3. estimation step: this step will compare the measurement Y_k from any sensor and the prediction $Y_{k+1/k} = h(X_{k+1/k})$. As h may also be non linear, we must compute: $\nabla h(X_{k+1/k}) = H_{k+1/k}$. The error is then calculated $\varepsilon_{k+1} = Y_{k+1} - Y_{k+1/k}$.The correction is then made through the gain matrix:

$$K_{k+1} = P_{k+1/k} H_{k+1/k} [H_{k+1/k} P_{k+1/k} H_{k+1/k}^t + Q_{k+1}]^{-1}. \quad (4)$$

The estimated vector is then

$$X_{k+1/k+1} = X_{k+1/k} + K_{k+1} \varepsilon_{k+1}. \quad (5)$$

We associate to this estimation the covariance matrix [1]

$$P_{k+1/k+1} = (I - K_{k+1} H_{k+1/k}) P_{k+1/k} (I - K_{k+1} H_{k+1/k})^t + K_{k+1} R_{k+1} K_{k+1}^t \quad (6)$$

This algorithm can handle several sensors. The only requirement is to provide measurement and the covariance matrix that gives the quality of the measurement. In the estimation step, a measurement from a sensor will be compared to prediction result, but also, the confidence on this measurement expressed by the covariance matrix is taken into account in equations (4) and (6). Equation (4) can be seen as a weight given to the measurement. When a measurement has a good confidence, the effect of K_{k+1} is important in the update equation (5) This confidence can be represented by ellipsoid (figure 9).

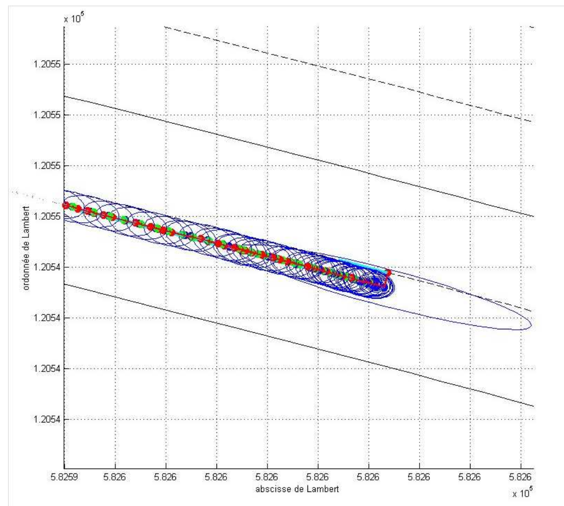

Figure 9: The confidence of the measurement is given by the covariance matrix.

We will see in the following part, how the algorithm handle sensors separately or together. The covariance matrices provided by sensors key

4.2 Using a map as location reference

We need a map of the track as reference of the vehicle location.

4.2.1 Design of the digital Map of the experimental track.

The map of the experimental track is built with topographic measurement. We start with the choice of some referential points at the center of the lane. When the track is locally a straight line, the distance of two consecutive points is twenty meters. And in the curved portion of the track, this distance is five meters. The position of the track's points are saved on a file. The available position the points and lane course are $(x_{track}, y_{track}, \theta_{track})$ (Figure 10).

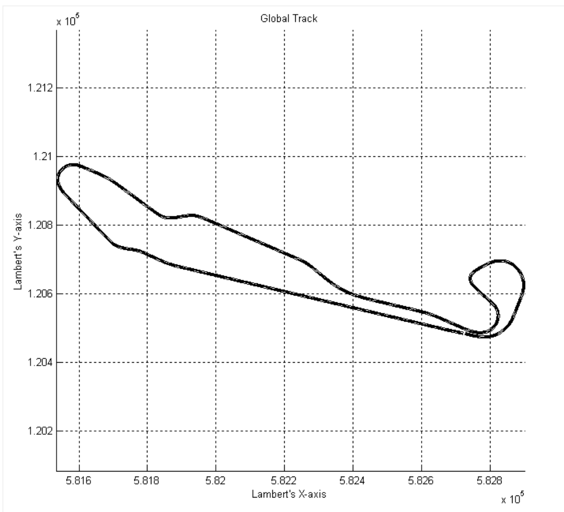

Figure 10: The map of experimental track in the Lambert's coordinate system.

Locally, the track is then approximate by straight segments. Each segment is described by the couple $(\rho_{seg}, \theta_{seg})$ in the straight line equation:

$$xcos(\theta_{seg}) + ysin(\theta_{seg}) - \rho_{seg} = 0 \tag{7}$$

5 Location by proprioceptive sensors in the Kalman prediction step.

Proprioceptive sensors measure the state of the vehicle. This is the reason why the measurement is never affect by external conditions. For that reason, these measurements are used in command vector of the Kalman system U_k. In

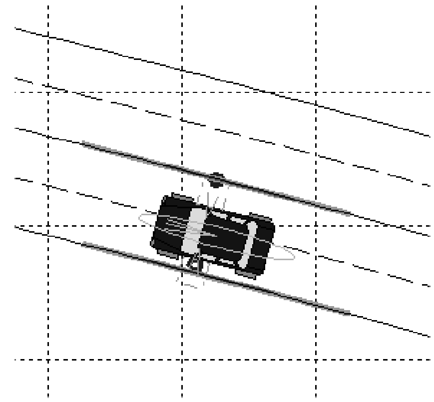

Figure 11: The lane marking is composed of a set of straight segments described by de equation (7).

our vehicle two kind of sensors are available: topometer and inertial sensor (INS). They provide the distance and the rotation speed of the vehicle.

$$U_k = \begin{pmatrix} \Delta d_k = d_{topo,k} - d_{topo,k-1} \\ \Delta\omega_k = r_{INS,k}(t_k - t_{k-1}) \end{pmatrix}$$

We are going to present briefly, the vehicle model used in the system, but this is not our topic. The vehicle is represented by its inertial center. The evolution of the vehicle is described by the equation:

$$X_{k+1} = f(X_k, U_k) = \begin{pmatrix} x_k + \Delta d_k \cos(\theta_k + \frac{\Delta\omega_k}{2}) \\ y_k + \Delta d_k \sin(\theta_k + \frac{\delta\omega_k}{2}) \\ \theta_k + \Delta\omega_k \end{pmatrix} \tag{8}$$

Topometer gives distance every 0.19 meter. We define then the measurement variance as $Var_{Topo} = 0.19^2$. The gyro of the INS measure the rotation speed r with an offset of 10^{-3} and a variance $var = 8.10^{-6}$, angle variation variance $Var_{angle} = var * \Delta t^2$ is then the variance of vehicle angle variation measurement. The covariance matrix of the command is

$$R_{command} = \begin{pmatrix} var_{Topo} & 0 \\ 0 & var_{angle} \end{pmatrix}$$

(Bruit de l'etat pas d'explication)

$$R_{system} = \begin{pmatrix} 0.1^2 & 0 & 0 \\ 0 & 0.1^2 & 0 \\ 0 & 0 & 0.1^3 * \frac{\pi}{180} \end{pmatrix}$$

As the system is non linear, we must calculate $F_k = \Delta_x f(X_k, U_k)$ and $B_k = \Delta_U f(X_k, U_k)$ The prediction covariance matrix is then given by

$$P_{k+1/k} = F_k P F_k^t + B_k R_{command} B_k^t + R_{system}$$

The kalman filtering can be used without estimation step, but the bias is growing with the distance. The test on the track is presented in the experimentation part.

6 Location by GPS in Kalman estimation step

6.1 On the measurements provided by GPS

The measurement is the absolute position of the vehicle (x, y). The precision depends on the mode of the GPS. The receiver switches automatically to the best one when this is available.

- In the RTK mode, the precision reaches several centimeters when the received signal is perfect. But this mode needs another fixed receiver to provide any measurement.

- The DGPS mode provides a less accurate measurement. But it doesn't need a fixed receiver. Only 3 satellites are needed during the initialisation.

The vehicle orientation θ must be calculated by using the previous value or given by proprioceptive sensors. The error of the θ estimation grows during the calculation. We must to take into account this error with this kind of sensors. We must also keep in mind that the GPS measurement is provided with a little delay (near $300ms$).

6.2 GPS measurement integration in the Kalman estimation step.

Before using GPS measurement in our system, we must define the covariance matrices for different modes of GPS.

GPS precision mode	Covariance Matrix Q_{GPS}
31 centimetric	$\begin{pmatrix} 0.04^2 & 0 \\ 0 & 0.04^2 \end{pmatrix}$
32 submetric	$\begin{pmatrix} 5^2 & 0 \\ 0 & 5^2 \end{pmatrix}$
9 metric	$\begin{pmatrix} 5^2 & 0 \\ 0 & 5^2 \end{pmatrix}$
0 decametric	$\begin{pmatrix} 10^2 & 0 \\ 0 & 10^2 \end{pmatrix}$

As GPS doesn't provide the vehicle course θ, only (x, y) is measured here. Then

$$Y_k = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} X_k$$

Let us note

$$H_k = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

We must replace $h_{X_{k+1/k}}$ by H_{k+1_k} in the step of estimation of the Kalman filter.

7 Location by Cameras in Kalman estimation step.

7.1 absolute location with cameras and Map.

Image analysis provide only local measurement. To obtain an absolute vehicle location, we use a simple map-matching technique [6]. The distance D measured by image analysis algorithm is defined in the equation 1. We must calculate the distance of the lane marking to the vehicle gravity center G, D_G thanks to D. First, we can expressed the lane marking's position with D in the system coordinate R presented in : $L = (0, D, 0)^t$ Let us note:

- x_c the position of the camera on the longitudinal axis of the vehicle,

- y_c the position of the camera on the lateral position of the vehicle,

- θ_c the orientation of the camera with respect to the vehicle longitudinal axis.

- $X_c = (x_c, y_c)^t$

(see figure 12)

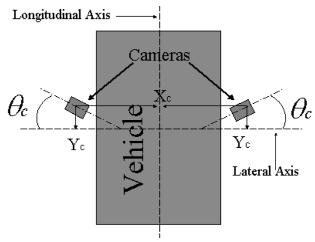

Figure 12: The camera's position is very important in measuring the distance of the vehicle to the lane marking.

We must first calculate the lane marking position on the absolute coordinate system. The position of the camera is then very important (figure 12). The position of the lane marking detected by the camera is calculated by the following formula:

$$X_{lane} = R_{vehicle} R_{camera} L \quad (9)$$

Where $R_{vehicle}$ is the rotation matrix with the angle θ, R_{camera} is the rotation matrix with the angle $theta_c$.
To compare predicted vehicle position and the vehicle position derived from cameras detection, we measure the algebraic distance of X_{lane} to the straight line given by the equation (7). The function h defined in the kalman system is then: $h(X_k) = x_k cos(\theta_{seg}) + y_k sin(\theta_{seg}) - \rho_{seg}$
By linearizing $h(X_k)$ we obtain the matrix H_k. The gain matrix K_k is defined by

$$K_k = PH_k^t (H_k PH_k^t + R_k)^{-1}$$

R_k corresponds to the covariance matrix provided by the lane marking position estimation by the image analysis algorithm [8].

8 On merging Sensors measurement and experimentation.

In this section, we are going to compare several combinations of different sensors. We will show through this comparison, how efficient result we can obtain when we merge different kind of sensors together. We propose to combine:

- topometer and inertial sensor. The Kalman filter is only using its prediction mode in this case,
- topometer, inertial sensor and GPS. The prediction will be done thanks to topometer and inertial measurement and the GPS information is used in estimation step,
- finally, we add to this last system, our lane marking detection by image analysis.

The experimentation is done on the track we have presented. We have tested on line our system. The measurements are saved at the same time.

8.1 topometer and inertial sensor.

The accuracy of the vehicle position is calculated during the test on the track, it is shown in the table below.

error of the vehicle position	distance
$0.5m$	$31m$
$1.0m$	$97m$
$2.m$	$183m$

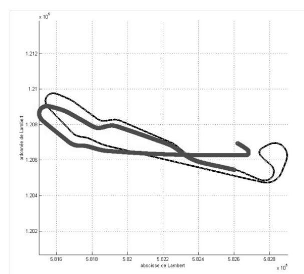

Figure 13: The accuracy of the prediction decreases with the distance. The vehicle's path does not match the experimental track.

We can plot the vehicle position estimated by the kalman prediction without any estimation step. This result show proprioceptive sensors can not be used alone (figure 13).

8.2 combination of topometer, inertial and GPS

The fusion of sensors allows us to use prediction and estimation steps of Kalman filter. The result depends on the

GPS mode. The drawback is the very frequent change of the GPS mode due to the loss of the GPS signal quality when the vehicle is near an obstacle. The figure (14)

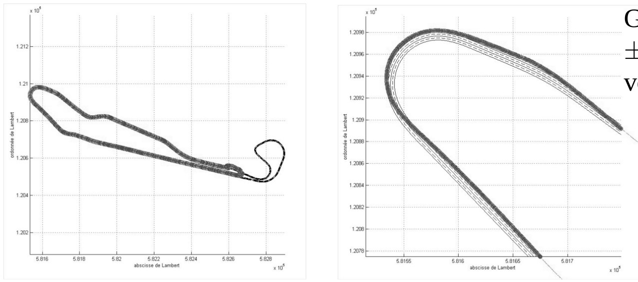

Figure 14: The vehicle's path seems now to be accurate. But if we zoom the image, we notice some estimation error. The vehicle's position is not on the track.

shows a better result when we use GPS with proprioceptive sensors. But this accuracy is not enough if we want to get a very accuracy vehicle position estimation.

8.3 combination of topometer, inertial, GPS and cameras.

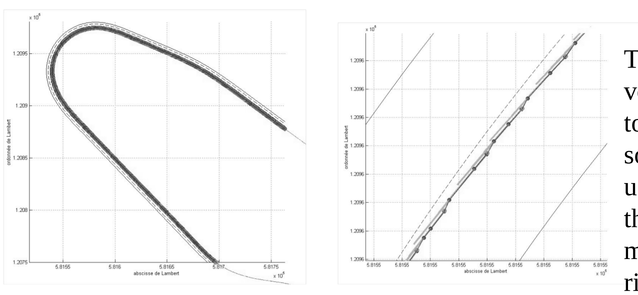

Figure 15: By using cameras information, the vehicle position estimation has a higher accuracy. The vehicle's path fits the track. In the second image, we notice that the inertial center of th vehicle are positioned nearly at the center of the road.

We can obtain an accurate vehicle's position estimation, if the vehicle lateral distances to the lane markings of both side of the vehicle are available. This information is only provided by cameras. By comparing figures 14 and 15, we notice the vehicle is positioned at the center of the lane when cameras detection is available. Without cameras, the vehicle seems to be out of the lane. When the GPS mode is RTK, the location is very accurate (error of $\pm 0.05m$), this accuracy allows us to detect some manoeuver of the driver (figure16).

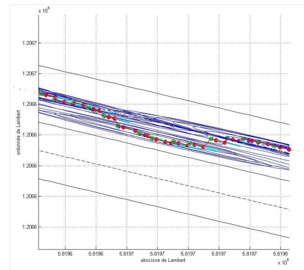

Figure 16: Thanks to the lateral position of the vehicle on the road, driver's manoeuver is detected.

9 Conclusion

The algorithm presented in this paper provides us with a very accurate vehicle location on the road. This allows us to show how powerful is a system that uses different sensors. Here, proprioceptive sensors, GPS,and cameras are used. The results obtained by our experimentation proves that computer vision has a key part in such a system. The main result is certainly the importance of the fusion algorithm that allows us to merge different kind of measurements.

Acknowledgments

This work is partly funded by Region de l'ile de France. The authors are grateful to LIVIC technical team for its help in the experiments and Francois Bulteau for his help for the experimental result.

References

[1] M. Athans, R.P Wishner and A. Bertoli - *Suboptimal State Estimation for Continuous Time Non-Linear Systems from Discrete Noisy Measurements.* - IEEE Trans. Automatica Control, volume AC-13 pages 504-514, october 1968.

[2] E.D. Dickmanns and A. Zapp - *A curvature-based scheme for improving road vehicle guidance by computer vision.* - In Proceedings of SPIE Conference on Mobile Robots. S. 161-16, volume 727, 1986.

[3] D. De Menton, -*A Zero-bank Algorithm for Inverse Perspective of a Road from a Single Image*, In Proceedings IEEE International Conference on Robotics and Automation, 1987, pp 1444-1449.

[4] E.D. Dickmanns and B.D. Mysliwetz -*Recursive 3D Road and Relative Ego-State Recognition*, In IEEE Transactions on Pattern Analysis and Machine Intelligence, 1992, vol. 14, pp 199-213.

[5] M.E.El Najjar and P. Bonnifait. -*A Road Reduction Method Using Multi-Criteria Fusion.* - IEEE Intelligent Vehicles Symposium (IV2002) Versailles, June 2002.

[6] D. Bernstein and A. Kornhauser - *An introduction to Map Matching for Personal Navigation Assistants* - Research report New Jersey TIDE Center, August 1996.

[7] Tarel, J.-P. and Guichard, F. - *Combined Dynamic Tracking and Recognition of Curves with Application to Road Detection.* - In Proceedings IEEE International Conference on Image Processing (ICIP'2000), September 10-13, 2000, volume I, pp 216-219.

[8] Tarel, J.-P. Ieng, S.-S and Charbonnier, P.- *Using Robust Estimation Algorithms for Tracking Explicit Curves.* - In Proceedings European Conference on Computer Vision (ECCV2002) May 2002, Part I, pp 492-507.

[9] Ieng, S.-S and Tarel, J.-P. -*On the Design of a Single Lane-Markings Detector Regardless the On-board Camera's Position.* - To be published in Proceedings Intelligent Vehicles Symposium (IV2003) June 2003.