ROAD GEOMETRY PREDICTION FOR MULTISENSOR SYSTEMS USED IN DRIVER ASSISTANCE APPLICATIONS

Mihai NICULESCU[1]

În transportul rutier, Sistemele Inteligente de Transport includ aplicaţii atât la nivelul adminstratorilor de reţea de drumuri, precum şi sisteme la bord de asistenţă a conducătorilor auto. Pentru acestea din urmă, ca să fie eficiente şi sigure, este esenţial să se asigure o reprezentare corectă a mediului din jurul vehiculului şi să fie întotdeauna conştiente de relaţia cu mediul în ceea ce priveşte poziţia, viteza şi direcţia de deplasare. Această lucrare va prezenta o soluţie pentru procesarea numerică a semnalelor pentru sisteme de asistenţă a conducătorului auto folosind fuziunea datelor provenite de la hărţi digitale şi a datelor de la senzori GPS, video şi radar.

In road transport, ITS systems include applications both at road network operators' level as well as on-board driver assistance systems. For the latter, in order to be efficient and reliable, it is crucial to be capable of providing an accurate representation of the environment around the vehicle and to be always aware of the vehicle relation with that environment in terms of position, speed and heading. This paper will present a solution for digital signal processing in a driver assistance system based on fusion of digital map data and data taken from GPS, video and radar sensors.

Keywords: multi-sensor fusion, driver assistance, digital maps, road geometry, intelligent transport systems

1. Introduction

At present, transport is one of the most important aspects of daily life. People always need transport services, be it personal mobility or supply of goods and services. As a direct consequence of this, all modes of transport have grown dramatically and become increasingly complex technically and logistically. Thus, communications and information technology are pervasive in all systems in transport.

EU's major concerns are related to the reduction of accidents, congestion and pollution. In this respect, several policy and technological measures are

[1] PhD student, Department of Railway Vehicles, University POLITEHNICA of Bucharest, Romania, e-mail: mniculesc@rdskink.ro

promoted, the most famous of these being Intelligent Transport Systems (ITS). ITS are defined as applications of electronics, telecommunications and information technology for safe, efficient and "clean" transport process management. ITS are used in all transport modes with a common core of mature technologies (identification and location of vehicles, voice and data communication with the vehicle, monitoring traffic situations, etc.) but also with particular elements of each mode.

In particular for road transport, some of the most common concerns today are related to the development of systems and applications to assist drivers both to increase comfort as well as for identifying dangerous situations and making decisions in these cases. Most of these applications are made using several different sensors mounted on board and use fusion algorithms to combine information from them. Multisensor systems are based on the synergy effect – that means having better and more accurate results when fusing information gathered from multiple sensors, as compared to using a single sensor.

Some benefits of using a multisensor system are:

- high level of detection, identification and tracking;
- better assessment of the situation;
- increased robustness;
- increased spatial and temporal coverage;
- lower response times.

In this article, we propose a method of information processing for a multisensor system that predicts the road geometry using information taken from a radar sensor, a camera and a digital map / GPS sensor. This system can be used in driver support applications such as ACC - Adaptive Cruise Control, Lane Change Assist or Collision Avoidance. At present these are used just for informational support however in the immediate future they will take a much more active role and will intervene actively on the vehicle which requires more stringent requirements on accuracy and quality of information used as input for decision making. These requirements can be met only by using multisensor fusion.

There are some other solutions proposed in the current literature. A very similar application is described in [1]. The objective is to estimate the road geometry using clothoids. The sensors used are GPS and a video camera. The algorithm we propose improves this especially by adding another sensor, the radar information which estimates the road geometry based on the paths of the vehicles in front of the ego vehicle. Another example of a similar system can be found in [3]. Here the sensors used are GPS, video camera information, vehicle dynamics and LIDAR. Klotz *et all* [5] describe a similar system intended only for the determination of the ego vehicle position. Also the fusion algorithm is simpler then what we propose as it uses only GPS and ABS derived information. In [7] a system that estimates the road geometry is described. However the sensors used are only GPS and a video camera for image processing which makes the algorithm less robust than the one we propose as it also uses radar information processing.

2. The general structure of a multisensor system

The generic name multisensor system actually refers to a system that receives information from several sensors, usually different, and then fuses this information into a unified model of the observed environment. Fig. 2.1 shows an example of a general architecture. This can be divided into two parts: physical and communication architecture.

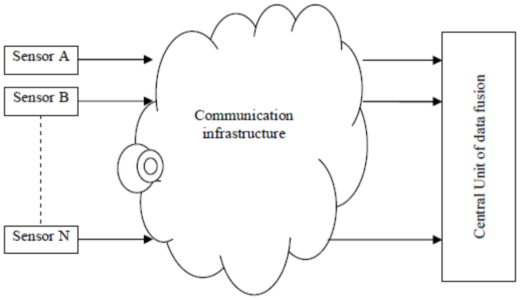

Fig.2.1. The general architecture of a multisensor system. Source: author

2.1. Physical architecture

Physical architecture refers to the way in which fusion is performed, after the data are available in the Central Unit of data fusion. There are three major solutions proposed in the literature for the physical architecture [1].

Fig. 2.2 presents a model in which unification is done at the primary sensory data. Then features are extracted and afterwards the decision is made on the elements making up the studied environment. This type of architecture can be used only when the sensors are of the same type, for example a number of radars or cameras. Fig. 2.3 shows the second architecture model in which the first step is feature extraction for each sensor and then the unification of general characteristics is made.

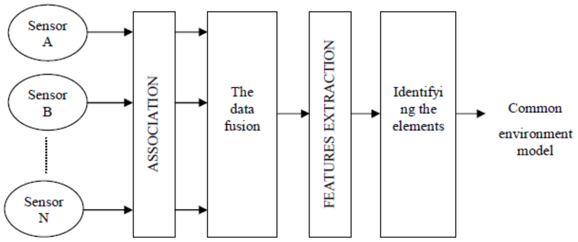

Fig. 2.2. Fusion of the primary sensory data. Adapted from Dave L. Hall and James Llinas,

“Introduction to Multisensor Data Fusion”, Proc. of IEEE, Vol. 85, No. 1, pp. 6 – 23, 1997

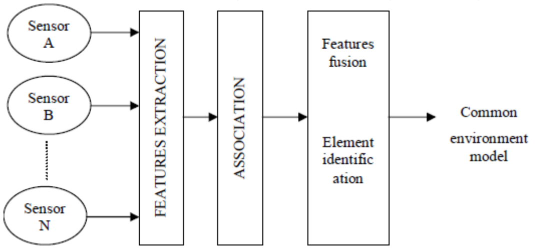

Fig. 2.3. Fusion of features. Adapted from Dave L. Hall and James Llinas, “Introduction to Multisensor Data Fusion”, Proc. of IEEE, Vol. 85, No. 1, pp. 6 – 23, 1997

The third model of physical architecture is illustrated in Fig. 2.4. Notice that each sensor data is processed at a high level representation of the environment. Unification is then performed at the level of these individual representations to obtain a common model.

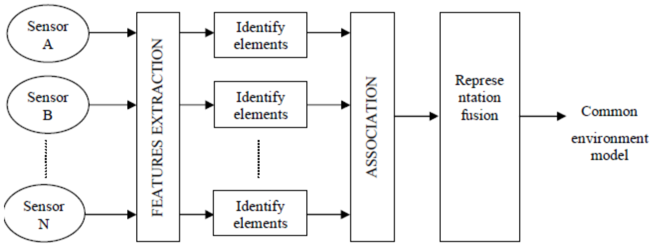

Fig. 2.4. High-level fusion. Adapted from Dave L. Hall and James Llinas, “Introduction to Multisensor Data Fusion”, Proc. of IEEE, Vol. 85, No. 1, pp. 6 – 23, 1997

3. Digital signal processing in multisensor systems

Multisensor systems acquire and process a significant amount of information. In all systems used for the transport field this information is available in digital format which is processed using standard algorithms to ensure optimal use of hardware and software. One of the best known algorithms is the Kalman filter. In the following we will present the elements of the Kalman filter that were used for the proposed solution. It works iteratively and builds a description of the environment based on vectors of properties. Thus a known mathematical model of the environment is needed and it will be constantly updated based on observations made.

Perception itself is merely a method by which an agent (an entity acting) can identify a specific behaviour [2]. An intelligent agent must understand the environment around him to plan and execute an action. Understanding the environment is based on a description which is built by fusing the "perceptions" from various sensing organs (or separate methods of interpretation) at discrete times. Perception can be defined as: the process by which an internal representation of the external environment is built and updated. The external environment is that part of the universe that an agent can perceive with its sensors at a given time. The perception by an agent may be called "dynamic environment modelling" which implies that the internal representation is constantly changing based on information received. A general representation of the dynamic modelling process is given in Fig. 3.1. It can be seen that individual observations are transformed into a common coordinate space and vocabulary. Further observations are merged into one internal representation in a cyclic process comprising three stages: prediction, detection and update.

Prediction: during the prediction phase the future state of the external environment is estimated based on the current state of the internal model.

Identification: during this phase observations are compared with predictions. Thus, both must be represented by a common vocabulary and the same coordinate space.

Update: during this phase information obtained from observations are integrated in the estimated state of the internal model to create an updated description of the external environment. Also, at this stage “old” data is deleted: those that "are not interesting" for the system or those that have proved transitory or wrong.

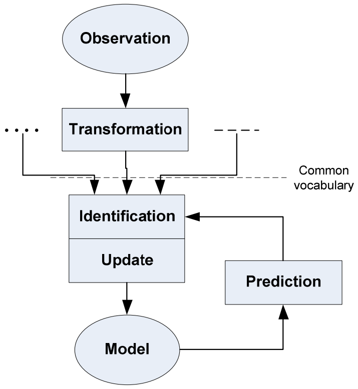

Fig. 3.1. Dynamic environment modelling [2]

3.1. Vectors of properties

A dynamic model of the environment *M (t)* can be built from a list of primitives *P$_i$ (t)* describing the state at a time *t*.

$$M(t) = \{P_1(t), P_2(t), \ldots, P_m(t)\} \tag{3.1}$$

Each primitive *P$_i$(t)* describes a portion of the local environment as an intersection of estimated properties $\widehat{X}(t)$ together with a unique identifier *ID* and a confidence factor *CF (t)*.

$$P(t) \equiv \{ID, \widehat{X}(t), CF(t)\} \tag{3.2}$$

The identifier acts as a primitive label and allows reuse whenever needed.

The confidence factor *CF (t)* allows the system to control the size of the model. Thus, new observed segments have a low confidence growing after each successive observation. However, if the segment is not present in the next observation after the initial detection, it will be regarded as noise and deleted. On the other hand, if the confidence is high enough, the segment will be kept a certain number of cycles, even if it does not appear in direct observations.

The vector $\widehat{X}(t)$ consists of a set of *n* properties estimating the local state of a part of the environment:

$$\widehat{X}(t) \equiv \{\widehat{x}_1(t), \widehat{x}_2(t), \ldots, \widehat{x}_n(t)\} \tag{3.3}$$

The real state of the environment *X (t)* is estimated by *Y (t)* following a process of observation noted ${}^{Y}\mathrm{H}_X$ affected by noise *Z (t)*:

$$Y(t) =^{Y} \mathrm{H}_X X(t) + Z(t) \tag{3.4}$$

The vector $\widehat{X}(t)$ is built in each cycle by combining an estimated vector *Y* (t)* with *Y (t)* resulting from observations.

The dynamic modelling process is conditioned on calculating the uncertainty of vectors $\widehat{X}(t)$ and *Y(t)*. This uncertainty can be modelled mathematically as a covariance matrix between $\widehat{X}(t)$ and *X (t)*:

$$\widehat{C}(t) \equiv E\left\{\left[X(t) - \widehat{X}(t)\right]\left[X(t) - \widehat{X}(t)\right]^T\right\} \tag{3.5}$$

3.2. Prediction equations of the Kalman filter

The prediction phase provides a projection for the vector $\widehat{X}(t)$ towards the estimated value $X^*(t + \Delta t)$ and for the uncertainty $\widehat{C}(t)$ towards the estimated value $C^*(t + \Delta t)$. This requires the determination of the derivatives of the elements of $\widehat{X}(t)$ and the covariance matrix between the elements and their derivatives. In terms of notation, derivatives may be added as elements of $\widehat{X}(t)$.

Let $\widehat{x}(t)$ be an element of $\widehat{X}(t)$ having variance $\widehat{\sigma}_x^2$. Then, its derivative is:

$$\widehat{x}'(t) = \frac{\partial \widehat{x}(t)}{\partial t} \tag{3.6}$$

Determination of first order derivative for *X (t)* can be done by grouping the Taylor series expansion higher order terms into an unknown random vector *V(t)* estimated by $\widehat{V}(t)$. Using these notations, we define the first prediction equation:

$$x^*(t + \Delta t) = \widehat{x}(t) + \frac{\partial \widehat{x}(t)}{\partial t}\Delta t + \widehat{v}(t) \tag{3.7}$$

Let $\widehat{Q}_x(t)$ be the estimation of uncertainty for vector $X^*(t+\Delta t)$ calculated by determining the covariance matrix between each element $\hat{x}(t)$ and its derivative. Then the second prediction equation is:

$$C_x^*(t+\Delta t) = \varphi\widehat{C}_x(t)\varphi^T + \widehat{Q}_x(t) \tag{3.8}$$

3.3. Identification - Mahalanobis distance

Let us consider the estimated model $M^*(t)$ having primitives $P_n^*(t)$ with vectors $X^*(t)$ and an observed model *O(t)* having primitives $P_m(t)$ with vectors *Y(t)*. The identification phase determines the most likely association of observed and predicted primitives based on the similarity between the predicted and observed properties. The mathematical measure for such similarity is to determine the difference of the properties, normalized by their covariance. This distance, normalized by covariance, is a quadratic form known as the squared Mahalanobis distance. Thus, noting ${}^Y\mathrm{H}_X$ a matrix that transforms the coordinate space of the estimated model to the observed model coordinate space, we can write:

$$Y_n^* = {}^Y\mathrm{H}_X X_n^* \tag{3.9}$$

In this case the covariance is:

$$C_{yn}^* = {}^Y\mathrm{H}_X C_{xn}^* \, {}^Y\mathrm{H}_X^{\;T} \tag{3.10}$$

Denoting by C_{ym} the covariance of observed vectors Y_m, the squared Mahalanobis distance D_{nm}^2 for the case where a single scalar property is compared is defined as follows:

$$D_{nm}^2 = \frac{1}{2} \cdot \frac{\left(y_n^* - y_m\right)^2}{\sigma_{y_n}^{*\,2} + \sigma_{y_m}^2} \tag{3.11}$$

The identification phase matching involves minimizing the normalized distance between predicted and observed properties or verifying that the distance falls within a certain number of standard deviations.

3.4. Update - update equations of Kalman filter

Let us consider the set of estimated properties $\widehat{X}_n(t)$ comprising a set of estimated properties $Y_n^*(t)$ and a set of observed properties $Y_m(t)$. One of the most important elements of the Kalman filter is a weighting matrix known as the Kalman Gain and defined as:

$$K(t) = C_x^*(t)^Y H_X^{\;T}\left[C_y^*(t) + C_y(t)\right]^{-1} \tag{3.12}$$

The Kalman gain provides a relative waiting between the prediction and observation, based on their relative uncertainties. The Kalman gain permits us to update the estimated set of properties and their derivatives from the difference between the predicted and observed properties:

$$\hat{X}(t) = X^*(t) + K(t)\left[Y(t) - Y^*(t)\right] \qquad (3.13)$$

The precision of the estimate is determined by:

$$\hat{C}(t) = C^*(t) - K(t)^Y H_X C^*(t) \qquad (3.14)$$

4. The proposed solution

Information from digital maps can be easily integrated with usual algorithms that determine the trajectory and with trajectory level fusion. However, firstly a conversion from the WGS 84 coordinate system used in digital maps to Cartesian space is needed.

The overall architecture of the proposed system is shown in Fig. 4.1. It consists of a radar module, GPS module and a video module. Information received from the modules is fused at the level of the parameters describing the geometry of the road, resulting in a fused representation of it.

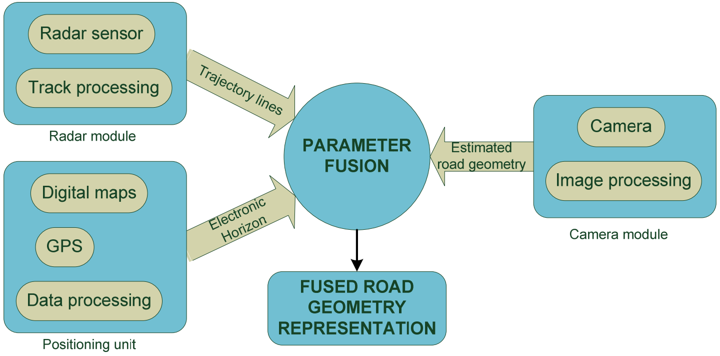

Fig. 4.1. System architecture. Source: author

4.1. Geometric modelling of the road

In civil engineering handbooks, the road can be modeled as a sequence of successive clothoid curves [3]. A clothoid is defined as a curve with curvature linearly depending on its arc length l:

$$c(l) = c_0 + c_1 l \qquad (4.1)$$

where c_0 is the initial curvature and c_1 is the curvature's rate with respect to its length.

Road borders are assumed to be sufficiently described by a clothoid, which in turn can be approximated by a 3rd order polynomial. Thus, considering the X axis parallel to the direction of the vehicle and perpendicular to the Y axis, we can write:

$$y(x) = y_0 + \tan(h) \cdot x + \frac{c_0}{2} x^2 + \frac{c_1}{6} x^3 \qquad (4.2)$$

with y_0 being the offset distance from the ego-vehicle's position and tan(h) being the tangent of the heading angle in the beginning of the road segment.

4.2. RADAR data modelling

Data from radar sensors is used to calculate trajectories based on the trajectories of the vehicles in front of the ego-vehicle. The algorithm is based on the assumption that all vehicles are moving within traffic lanes and that the trajectory of the vehicle in front is a possible trajectory for the ego-vehicle.

Some checks are needed for the algorithm to work properly. Thus, there must be first a selection mechanism to eliminate certain paths - eg changing lane for vehicles. Also, another aspect to take into account is lateral movement of vehicles detected.

The position information provided by the radar sensor is processed so as to finally obtain a relation to determine the clothoid parameters h, c_0 and c_1.

4.3. Modelling digital map data

Navigation equipment is used to extract information about road segments that the vehicle might travel. The resulting information is called "electronic horizon" and includes all possible routes of the ego-vehicle.

The algorithm first transforms the geodetic coordinates of the points in Cartesian coordinates. The next step is to group points in road segments. Then, for each segment a Kalman filter is used to determine the parameters h, c_0 and c_1 of the clothoid. The end result is a list of segments each with its associated parameters that are then used in the fusion algorithm.

4.4. Modelling data from video camera

By processing the images obtained by the camera information on the parameters that characterize the geometry of the road can be extracted [5]. Thus, specialized software is used to identify lane markings and also ego-vehicle position within the lane.

First, based on a pre-existing model that describes the layout of the lane markings, the digital image is processed using contour detection and segmentation techniques to identify the areas where the markings are. Then measurements are made only for the areas identified. Finally the parameters describing the clothoid are calculated.

4.5. Fusion algorithm

The final geometry is calculated by fusing the parameters of the clothoid model from each individual source of information. The variance of each clothoid parameter is taken into consideration in order to assign the greater weight to the source with the lower estimation error.

The road geometry is described by equation (4.2) where each parameter p representing the tan(h), c_0 and c_1 are calculated with:

$$p = \frac{w_c \cdot p_c + w_r \cdot p_r + w_h \cdot p_h}{w_c + w_r + w_h} \tag{4.3}$$

where the indices c, r and h denote the camera, radar and digital maps respectively both for the weights w and the parameters p.

5. Testing the proposed solution

The proposed solution was tested by implementing the fusion algorithm within Matlab programming environment using real input data for the GPS module and simulated data for radar and video modules [6].

The algorithm was applied to several sections of road selected to include both curved and linear profiles. For each section, 12 triplets of GPS coordinates (latitude, longitude and height) were used, stored in sequential order of their acquisition in the direction of travel of the ego-vehicle. Sections were subdivided into partially overlapping segments consisting of three successive points. Thus, the origin of the first segment, is the first point; the origin of the second is point 2 and so on. We chose this method for two main reasons:

- creating as many segments as possible given the acquired coordinates
- better distribution for areas where the available points are not evenly distributed along the road.

For each segment the clothoid parameters describing the road geometry were determined according to the relation (4.2). Parameters obtained were fused, according to relation (4.3), with the simulated parameters of the radar and video modules. For each segment the clothoid curve described by the parameters calculated was plotted, yielding the final section by adding graphic representations of each segment. This representation was compared with the actual profile of the road section as represented on the digital map.

To check the calculated parameters, the road geometry representation from the digital map was considered as a reference. Therefore, we determined the absolute difference between the calculated parameters using the fusion algorithm and the same parameters calculated using as input the selected successive GPS coordinates from the digital map corresponding to the road segment analyzed, according to the relation:

$$dif_p = \left| \frac{p_f - p_h}{p_h} \right| * 100 \tag{5.1}$$

where:

dif_p is the absolute difference in percent

p_f is the parameter calculated by fusion

p_h is the parameter calculated using the coordinates from the digital map

5.1. Results

Below, we present the results for two scenarios: a linear road profile and a curved one.

The route chosen for the first scenario is represented in Fig. 5.1. The route consisting of the 12 points identified by the GPS receiver is represented with yellow colour while with red colour the real geometry of the road is platted. From it 12 points marked with p01 ... p05 and p1 ... p7 were extracted

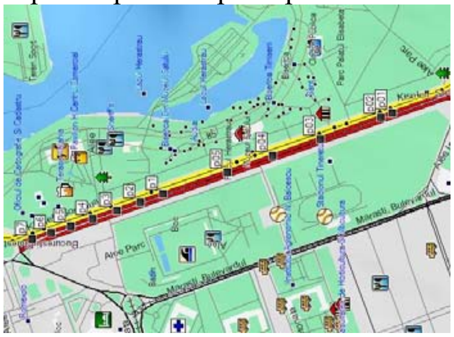

Fig. 5.1. The route chosen for the first simulation scenario. Source: author

In Fig. 5.2 road geometry determined by fusion is represented and in Fig. 5.3 road geometry for the same section is shown but determined using GPS coordinates selected from the map. For both x and y axes conventional values are used related to each other by the clothoid equation.

In both cases one can see large estimation errors when the points that make up a certain segment are far apart and unevenly distributed.

A third graphical representation is shown in Fig. 5.4. It describes the absolute differences for each parameter tan, c_0 and c_1 calculated for each segment separately. For the fused parameters, the weight of the GPS sensor was three times lower than the one of the radar sensor and two times lower than the one of the video sensor because it was noticed that there were significant differences between the Cartesian coordinates derived from the coordinates acquired by GPS and those derived from the coordinates on the map. The algorithm is designed to prefer the acquired coordinates but if there are significant differences compared to the most likely and near map track, its coordinates will be considered but with a lower weight for the GPS sensor as it is considered that the GPS signal is weak and therefore positioning errors are large. Moreover, if a likely and near track cannot be found on the map, which is common with poor quality maps, the weight of the GPS sensor will be higher than in the previous situation but still lower compared with other sensors.

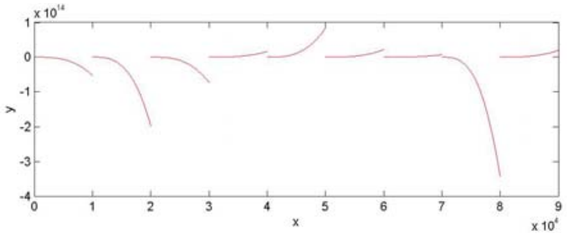

Fig. 5.2. Scenario 1. Section geometry resulting from fusion. Source: author

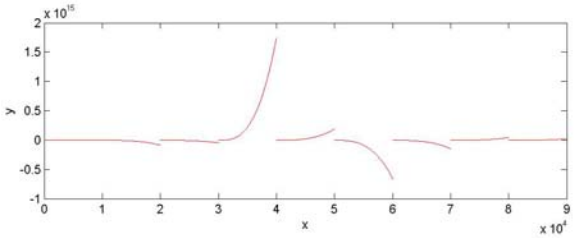

Fig. 5.3. Scenario 1. Section geometry resulting from the map points. Source: author

By comparison, it can be seen in Fig. 5.5 that the differences are greater when the weights are reversed, ie GPS sensor has the largest share. Analyzing each segment individually, it is found that the percentage error is very high for certain segments (eg 1 or 8) where there are significant positioning differences between the points selected from the map and the points acquired by GPS.

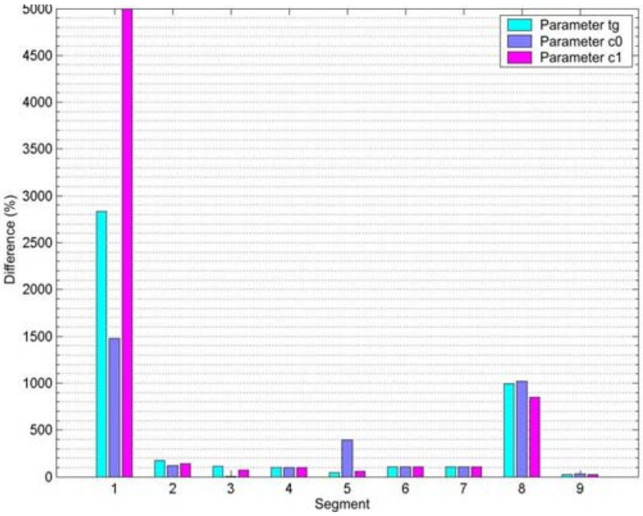

Fig. 5.4. Absolute differences for each parameter of the clothoid when the weight of the GPS sensor is lesser than the others. Source: author

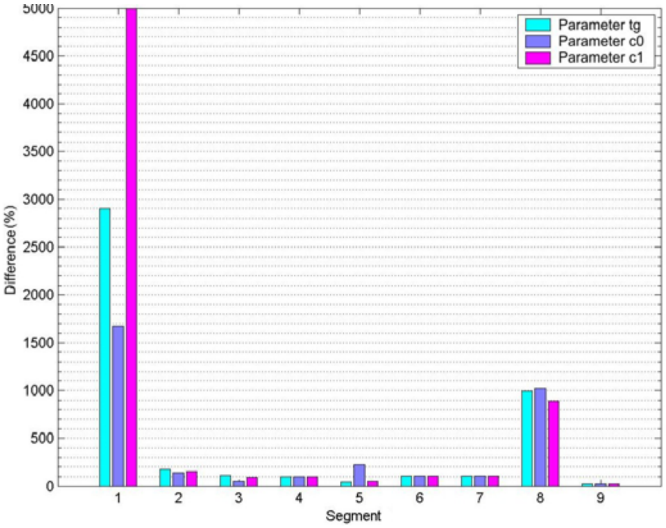

Fig. 5.5. Absolute differences for each parameter of the clothoid when the weight of the GPS sensor is higher than the others. Source: author

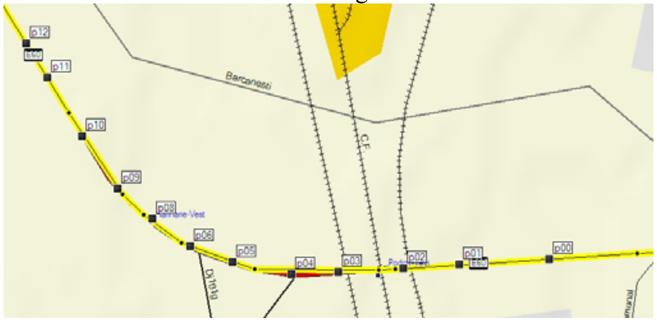

Fig. 5.6. The route chosen for the second simulation scenario. Source: author

The route chosen for the second scenario is represented in Fig. 5.6. The route consisting of the 12 points identified by the GPS receiver is represented with yellow colour and with red colour the real geometry of the road is represented. From it 12 points were extracted marked with p_{00} ... p_{11}.

In Fig. 5.7 road geometry determined by fusion is represented while in Fig. 5.8 road geometry for the same section is shown but determined using GPS coordinates selected from the map. For both *x* and *y* axes, conventional values are used related to each other by the clothoid equation. This time, given the observations mentioned in the previous scenario, the points from the map were chosen more evenly spaced and it can be seen that the geometric representation of the section is much closer to the real situation.

A third graphical representation is shown in Fig. 5.9. It describes the absolute differences for each parameter tan, c0 and c1 calculated for each segment separately. For the fused parameters, the weight of the GPS sensor was the height because this time it was noticed that there were effectively no differences between the Cartesian coordinates derived from the coordinates acquired by GPS and those derived from the coordinates on the map. By comparison, it can be seen in Fig. 5.10 that the differences are greater when the weights are reversed, ie GPS sensor has the lowest share.

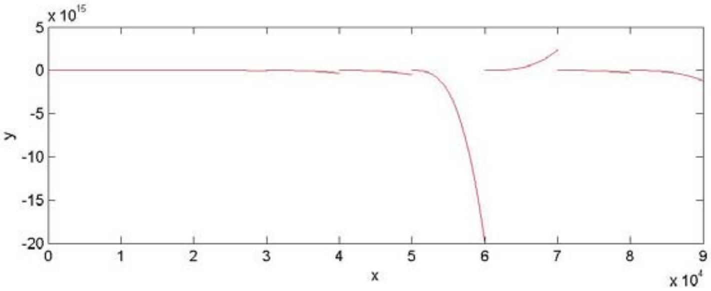

Fig. 5.7. Scenario 2. Section geometry resulting from fusion. Source: author

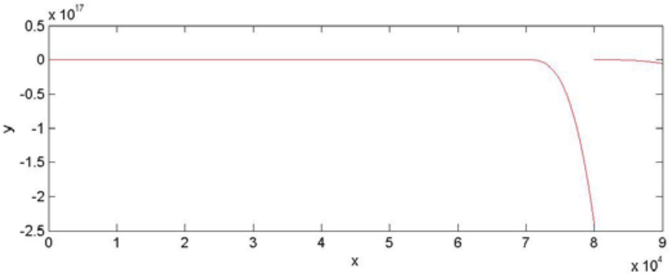

Fig. 5.8. Scenario 2. Section geometry resulting from the map points. Source: author

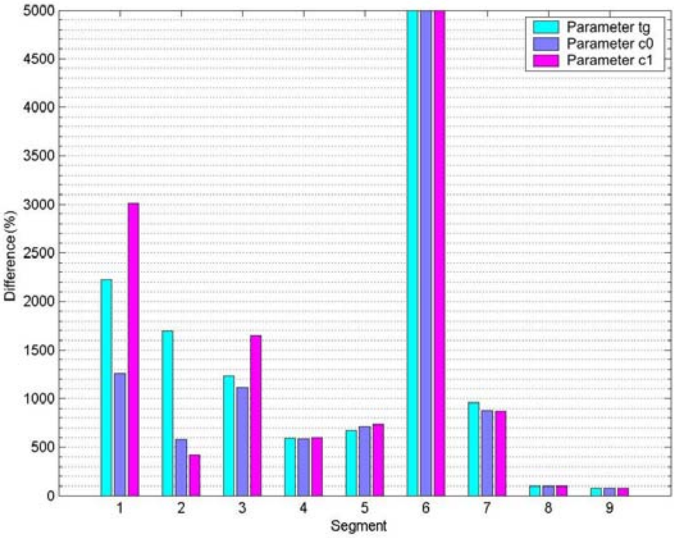

Fig. 5.9. Absolute differences for each parameter of the clothoid when the weight of the GPS sensor is higher than the others. Source: author

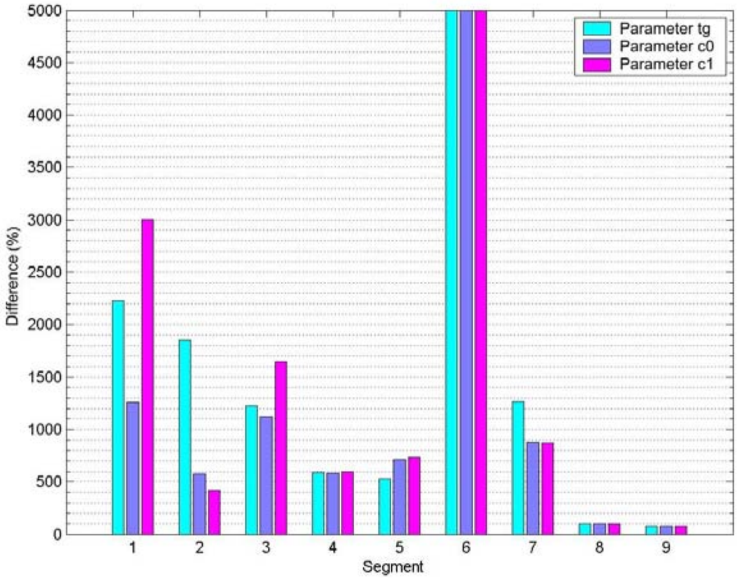

Fig. 5.10. Absolute differences for each parameter of the clothoid when the weight of the GPS sensor is lesser than the others. Source: author

6. Conclusions

In this article we presented a method of information processing and fusion for multisensor systems used for road geometry prediction in driver assistance applications.

Accurate representation of the geometry of the road is essential not only to estimate the ego-vehicle position and the degree of risk traffic manoeuvres involve but also to identify objects that are on the ego-vehicle's trajectory or are about to intersect with it. All this information is used for various applications such as ACC - Adaptive Cruise Control, Lane Assist or Collision Avoidance. At present these are used just for informational support; however in the immediate future they will take a much more active role and will intervene actively on the vehicle which requires more stringent requirements on accuracy and quality of information used as input for decision making. These requirements can be met only by using multisensor fusion.

In Chapter 5 we presented the tests performed to verify and validate the proposed method. Test procedure is based on software developed using the Matlab programming environment.

Results are described in two situations: a linear road profile and a curved one. Also, another difference between the two is that the first case the acquired GPS points have larger errors. This is the typical scenario for an urban area where the GPS signal is weak, even nonexistent for shorter or longer periods.

Results from the first scenario show that using points acquired by the GPS receiver produces erroneous geometry estimates, just because points are unevenly distributed and largely spaced apart. It was also shown that this shortcoming can be partially offset by the use of selected points from the digital map from the section which is most likely to correspond with the acquire points. But there are situations in which this section cannot be determined because the map is not detailed enough in that area. However, the algorithm compensates for these shortcomings by reducing the corresponding weight of GPS module compared to the other sensors.

Results from the second scenario support the conclusions obtained. Since the points acquired by GPS are positioned correctly, the road geometry is well estimated whether acquired or map points are used. Also, it is confirmed that the weight must be related to the more reliable sensor. In other words, this time a higher weight associated with the GPS sensor determines a correct representation resulting from the fusion.

It can be concluded that the tests validate the proposed algorithm; so, using digital maps is an effective solution for the fusion algorithm.

BIBLIOGRAPHY

[1] *Gackstatter, C., Heinemann, P., Thomas, S., Rosenhahn, B., Klinker, G.* – Fusion of Clothoid Segments for a More Accurate and Updated Prediction of the Road Geometry, 13th International IEEE Annual Conference on Intelligent Transportation Systems, Portugal, 2010

[2] *Crowley, J. L., Demazeau, Y.* – Principles and Techniques for Sensor Data Fusion, http://www-prima.inrialpes.fr/Prima/Homepages/jlc/papers/SigProc-Fusion.pdf

[3] *Tsogas, M., Floudas, N., Lytrivis, P., Amditis, A., Polychronopoulos, A.* – Combined lane and road attributes extraction by fusing data from digital map, laser scanner and camera, Information Fusion, **Volume 12**, Issue 1, Special Issue on Intelligent Transportation Systems, January 2011, Pages 28-36, ISSN 1566-2535, DOI: 10.1016/j.inffus.2010.01.005

[4] *Serediuc, C.* – Considerații asupra sistemelor de coordonate şi coordonatelor folosite în rețele geodezie convenționale şi GPS (On systems of coordinates and coordinates used in conventional geodetic systems and GPS), RevCAD'06 magazine, **No. 6**, 2006 (in Romanian)

[5] *Klotz, A., Sparbert, J., Hötzer, D.* – Lane data fusion for driver assistance systems, Proceedings of the 7th International Conference on Information Fusion, Stockholm, Sweden, 2004, pg. 657–663.

[6] *Niculescu, M. C.* – Contribuții privind procesarea informației în sisteme multisenzor – cu aplicabilitate în transporturi (Contributions towards information processing in multisensor systems – applications for transports field), Ph.D. thesis, POLITEHNICA University of Bucharest, Bucharest, 2012 (in Romanian)

[7] *Dietmayer, K., Kampchen, N., Furstenberg, K., Kibbel, J., Justus, W., Schulz, R.* – Roadway Detection and Lane Detection using Multilayer Laserscanner, AMAA Handbook, Springer-Verlag, 2005, pg. 197–213.