Short Papers

Maneuver Prediction for Road Vehicles Based on a Neuro-Fuzzy Architecture With a Low-Cost Navigation Unit

Rafael Toledo-Moreo, *Member, IEEE*,
Miguel Pinzolas-Prado, *Member, IEEE*, and
Jose Manuel Cano-Izquierdo, *Member, IEEE*

***Abstract*—Collision avoidance is currently one of the main research areas in road intelligent transportation systems. Among the different possibilities available in the literature, the prediction of abrupt maneuvers has been shown to be useful in reducing the possibility of collisions. A supervised version of dynamic Fuzzy Adaptive System ART-based (dFasArt), which is a neuronal-architecture-based method that employs dynamic activation functions determined by fuzzy sets, is used for maneuver predicting and solving the problem of intervehicle collisions on roads. In this paper, it is shown how the dynamic character of dFasArt minimizes problems caused by noise in the sensors and provides stability on the predicted maneuvers. Several experiments with real data were carried out, and the SdFasArt results were compared with those achieved by an implementation of the Incremental Hierarchical Discriminant Regression (IHDR)-based method, showing the suitability of SdFasArt for maneuver prediction of road vehicles.**

***Index Terms*—Collision-avoidance support, inertial sensors, maneuver prediction, neuro-fuzzy.**

I. Introduction

The problem of intervehicle collisions on roads may be addressed from different points of view. In the current literature, three different main paradigms can be found for collision-avoidance support: 1) systems based on an intelligent vehicle, which are capable of dealing with a number of unfriendly and changing environments; 2) systems that rely on shared information with the road infrastructure, coming, for instance, from the traffic authorities; and 3) cooperative systems in which the conjunction of vehicles involved in the scene intends to avoid possible collisions. Whereas, in the case of the first ones, high investments in the vehicle onboard equipment (OBE) are required, the latter two options demand high penetration rates and, therefore, low-cost devices. In particular, in the case of cooperative systems, the timely prediction of the vehicle maneuver can effectively support the system to prevent risky situations.

In this paper, a low-cost system for maneuver prediction based on an Inertial Measurement Unit (IMU) and odometry sensors is presented. The performance of the IMU sensors highly depends on the technology used and the unit price. Since market considerations are taken into account in this research, only low-cost Microelectromechanical Systems (MEMS) devices are considered in the OBE of the vehicle. On the other hand, these sensors are or will commonly be available in standard road vehicles.

In the approach presented in this paper, we focus on the problem of longitudinal collisions, which are typical in traffic jams and so-called "stop & go" situations. We suggest three longitudinal maneuver states intended to be recognized, i.e., *Acceleration* and *Deceleration (AC), Cruise (CR),* and *Stationary (ST)*. The dynamic Fuzzy Adaptive System ART-based (dFasArt) neural architecture can naturally address this issue. Due to its dynamic quality, dFasArt allows taking into account the time span of input data, without the need to keep buffers of past input or output values. The dynamic characteristic of the activation functions provides a natural way of filtering noise in inputs, whereas the dynamic evolution of the reset signals allows stability in the predictions. In dFasArt, learning is unsupervised and incremental [1], [2]. For details of the dFasArt method, see [2].

In this paper, a supervised version of the dFasArt algorithm (SdFasArt) has been developed. This modification follows the Adaptive Resonance Theory MAP (ARTMAP) philosophy, maintaining the maximum-generalization–minimum-prediction error principle.

Apart from supporting collision avoidance, maneuver recognition may provide higher accuracy in the process of positioning and error estimation by choosing the model that better describes the vehicle dynamics at any time. Although this issue is not discussed in this short paper, future investigations will focus on it.

In addition to the SdFasArt-based algorithm, an implementation of the Incremental Hierarchical Discriminant Regression (IHDR) has been developed, providing efficient results, at the expense of highly intensive training process. Both methods are compared and analyzed in this work.

The rest of this paper is organized as follows: First, relevant works related to this issue are discussed. Next, the SdFasArt and IHDR classification techniques are explained. In Section IV, brief descriptions of the OBE of the vehicle and the multisensor filtering are given, and the results of experimental trials with real data are shown. Main conclusions are finally discussed.

II. Related Work

Collision avoidance is discussed in the literature from many different points of view. A number of works related to the collision-avoidance support systems area can be found based on radar systems [3], [4], vision [5]–[7], or combinations of both to avoid the weak points related to any single approach [8] at the expense of increasing the final price of the OBE. McCall *et al.* [9] made an interesting comparison of previous research distinguishing between driver intent inference and trajectory prediction.

The method proposed in this paper predicts maneuver changes based on the vehicle dynamics being independent of the visibility, weather conditions, or blockages of the Global Positioning System (GPS) signals. The idea of using vehicle dynamics for maneuver recognition has been used previously in aerial navigation [10]. In the road transport field, this topic has been treated in a number of works. Several works are based on the idea of using different vehicle dynamics for object

Manuscript received April 9, 2008; revised January 8, 2009 and July 30, 2009; accepted December 11, 2009. Date of publication January 19, 2010; date of current version May 25, 2010. This work was supported by the Spanish Ministerio de Fomento and Ministerio de Ciencia e Innovación under Grant FOM/2454/2007 and Grant TIN2008-06441-C02-02. The Associate Editor for this paper was D. J. J. Dailey.

R. Toledo-Moreo is with the Department of Electronics, Computer Technology and Projects, Technical University of Cartagena, 30202 Cartagena, Spain, and also with the Research Group of Intelligent Systems, University of Murcia, 30003 Murcia, Spain (e-mail: rafael.toledo@upct.es; toledo@um.es).

M. Pinzolas-Prado and J. M. Cano-Izquierdo are with the Department of Systems Engineering and Automation, Technical University of Cartagena, 30202 Cartagena, Spain (e-mail: miguel.pinzolas@upct.es; josem.cano@upct.es).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2009.2039011

tracking and navigation purposes [11]–[13]. Some authors focused their efforts on recognition of vehicle behaviors by using a set of diverse kinematical models, with each model developed to represent the vehicle behavior in a particular maneuvering state [14]. Other approaches can be found in the literature, such as [15], where finite-impulse-response filters were employed. Nevertheless, many authors find the tuning of the filter parameters problematic due to the very diverse nature of a road vehicle maneuver, depending, for example, on the road shape. Finally, it has been found that increasing the number of vehicle models does not improve the results achieved.

In preliminary experiments, the proposed supervised dFasArt (SdFasArt)-based system was proven to be capable of distinguishing among the vehicle dynamics, avoiding tuning process problems [16]. A more exhaustive study of maneuver prediction is now presented, with further experiments and more detailed analysis of the results. In addition, the SdFasArt solution is compared with an implementation of the IHDR method [17], presenting good results.

III. Classification Systems

Two classification systems have been implemented in our experiments, i.e., an IHDR-based method and a supervised version of dFasArt, which is presented next.

A. SdFasArt

The FasArt model links the ART architecture with fuzzy logic systems, establishing a relationship between the unit activation function and the membership function of a fuzzy set [18]. On the one hand, this allows interpreting each of the FasArt unit as a fuzzy class defined by the membership-activation function associated with the representing unit. On the other hand, the rules that relate the different classes are determined by the connection weights between the units.

Derived from FasArt, dFasArt uses dynamic activation functions, which are determined by the weights of the unit. These weights can be regarded as the defining parameters of a fuzzy set membership function [1]. In dFasArt, learning is unsupervised and incremental. In this paper, a supervised version on the dFasArt algorithm has been developed. This modification follows the ARTMAP philosophy, maintaining the maximum generalization-minimum prediction error principle.

The SdFasArt architecture is shown in Fig. 1(a). SdFasArt uses a dynamic activation function determined by the weights of the unit as the membership function of a fuzzy set. The signal activation is calculated as the AND of the activations of each one of the dimensions when a multidimensional signal is considered. This AND is implemented using the product as a T-norm. Hence, the activity T_j of unit j for an M-dimensional input $\vec{I} = (I_1, \ldots, I_M)$ is given by

$$\frac{dT_j}{dt} = -A_T T_j + B_T \prod_{i=1}^{M} \eta_{ji}\left(I_i(t)\right) \quad (1)$$

where η_{ji} is the membership function associated with the ith-dimension of unit j, which is determined by weights w_{ji}, c_{ji}, and v_{ji}, as shown in Fig. 1(b). As it can be seen, (1) includes a term of passive decay with the decay rate A_T, which reduces the unit activity when the signal moves away from the hyperbox that defines it, and a term of excitation with excitation gain B_T. If the condition $A_T = B_T$ is imposed in this equation and if input $I_i(t)$ remains time constant, when $t \to \infty$, it can be found that

$$T_j \to \prod_{i=1}^{M} \eta_{ji}\left(I_i(t)\right) \quad (2)$$

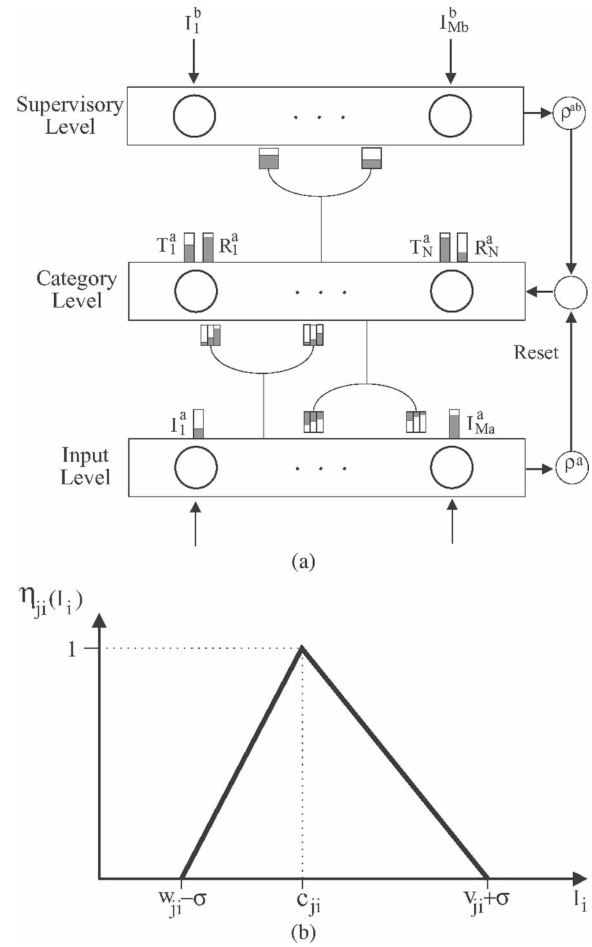


Fig. 1. (a) SdFasArt model architecture. (b) Membership-activation function.

which makes the original equation of the FasArt model without dynamics.

In the Fuzzy ARTMAP model, there is one limitation with regard to the normalization of the inputs. Carpenter *et al.* [19] introduced the possibility of normalizing the inputs, maintaining the relative importance of each component. However, to do so, it is implied that the components of the input vector should be within the [0, 1] interval. In the most general case, this condition is not satisfied, and a prenormalization of the signal must be done. To do so, we must know *a priori* the minimum and maximum values of every signal, which is not always possible. The need for the previous signal processing phase of every input signal constraints the online adaptive character of the system.

One could argue that, in a road vehicle, the minimum and maximum values of acceleration, speed, and angular rate are characterized by the vehicle dynamics features. However, it can be found that the very different environments in which a road vehicle can entail very different vehicle dynamics, and the assumptions done in some cases may not well represent the vehicle behavior in some others [13].

We propose to solve this problem by introducing the parameter σ, which determines the fuzziness of the class associated with the unit. Therefore, in our approach, it is assumed that the variability of the maximum and minimum values of the signal is not known *a priori*. The σ parameter can be calculated as

$$\sigma = \sigma^* |2c_j| + \epsilon \quad (3)$$

where σ^* stands for the generality level of the associated fuzzy set (the values of $\sigma^* \to 0$ make the set less fuzzy, whereas the values of $\sigma^* \to \infty$ increases set fuzziness), the parameter $\epsilon > 0$ fixes the minimum value of σ, and c_j represents a typical value of the category.

The election of the winning unit J is carried out, following the winner-takes-all rule, i.e.,

$$T_J = \max_j \{T_j\}. \tag{4}$$

The learning process starts when the winning unit meets a criterion. This criterion is associated with the size of the support of the fuzzy class that would contain the input if this was categorized in the unit. This value is dynamically calculated for each unit according to

$$\frac{dR_j}{dt} = -A_R R_J + B_R \sum_{i=1}^{M} \left(\frac{\max(v_{Ji}, I_i) - \min(w_{Ji}, I_i)}{|2c_{Ji}| + \epsilon} \right). \tag{5}$$

The R_j value represents a measurement of the change needed on the class associated to the j unit to incorporate the input. To see if the J winning unit can generalize the input, it is compared with the design parameter ρ so that two conditions hold.

1) If

$$R_J \leq \rho \tag{6}$$

the matching between the input and the weight vector of the unit is good, and the learning task starts.

2) If

$$R_J > \rho \tag{7}$$

there is not enough similarity; therefore, the Reset mechanism is fired. This inhibits the activation of unit J, returning to the election of a new winning unit.

If the Reset mechanism is not fired, then the learning phase is activated, and the unit modifies its weights. The *Fast-Commit Slow-Learning* concept is commonly used.

When the winning unit represents a class that had performed some other learning cycle (*committed unit*), the weights are *Slow-Learning* updated according to the following equations:

$$\frac{d\vec{W}}{dt} = -A_W \vec{W} + B_W \min\left(\vec{I}(t), \vec{W}\right) \tag{8}$$

$$\frac{d\vec{C}}{dt} = A_C(\vec{I} - \vec{C}) \tag{9}$$

$$\frac{d\vec{V}}{dt} = -A_V \vec{V} + B_V \max\left(\vec{I}(t), \vec{V}\right). \tag{10}$$

For the case of the uncommitted units, the class is initialized with the first categorized value *Fast-Commit*, i.e.,

$$\vec{W}_J^{\mathrm{NEW}} = \vec{C}_J^{\mathrm{NEW}} = \vec{V}_J^{\mathrm{NEW}} = \vec{I}. \tag{11}$$

Supervision is carried out in the supervisory level by means of vector $\vec{I^b} = (I_1^b, \ldots, I_{Mb}^b)$. In this level, for each time instant, $\vec{I^a}$ activates the corresponding unit. The $\vec{W_k^{ab}}$ matrix of adaptive weights associates, in a many-to-one mapping, units of category level to units of supervisory level. When a unit J is activated for the first time in the category level, weights are adapted by means of a fast-learning process

$$\vec{W_J^{ab}} = \vec{I^b}. \tag{12}$$

If unit J is a committed unit, a matching between the membership value to the predicted category and the "crisp" desired value is carried out.

1) If

$$\left| \vec{W_J^{ab}} \wedge \vec{I^b} \right| \geq \rho^{ab} |\vec{I^b}| \tag{13}$$

then prediction corroborates supervision.

2) If

$$\left| \vec{W_J^{ab}} \wedge \vec{I^b} \right| < \rho^{ab} |\vec{I^b}| \tag{14}$$

then the matching between prediction and supervision is not strong enough. In this case, the Reset signal is fired, and a new prediction is made.

When no supervision is present, SdFasArt will predict the value associated with the weight vector of the winning unit as output, i.e., $\vec{W_J^{ab}}$.

B. IHDR

Designed as an approximate computational model for automatic development of associative cortex, IHDR [20], [21] incrementally builds a decision tree or regression tree, using bottom–up sensory inputs and top–down motor projections. It is particularly adequate to deal with very high-dimensional regression or decision spaces when an online real-time learning and classification system is required. At each internal node of the IHDR tree, information in the output space is used to automatically derive the local subspace spanned by the most discriminating features. Embedded in the tree is a hierarchical probability distribution model used to prune very unlikely cases during the search. The number of parameters in the coarse-to-fine approximation is dynamic and data driven, enabling the IHDR tree to automatically handle data with unknown distribution shapes. The IHDR tree dynamically assigns long-term memory to accommodate new samples and avoid the loss-of-memory problem that is typical with a global-fitting learning algorithm for neural networks.

One of the main contributions of the IHDR algorithm is the sample size-dependent negative-log-likelihood metric. This metric allows efficiently computing distances to classes with very different numbers of samples in them, which is a challenging issue for an incremental classification system.

The IHDR algorithm is particularly appropriate for operation with a high volume of data. In this paper, this capability has been used to take into account the influence of past states in the classification of the current maneuver. To do this, the IHDR classifier is fed with not only information coming from the present output of the sensors but past sensor data as well. If $\vec{S}(k)$ is a vector representing the sensor data gathered at time kT, with T being the sample time, to predict the maneuver being carried out at time kT, the IHDR input is considered to be the vector $\vec{I}(k) = \{\vec{S}(k), \vec{S}(k-1), \vec{S}(k-2), \ldots, \vec{S}(k - \mathrm{NoD})\}$, with NoD being the number of delays, which is a parameter that will experimentally be adjusted.

In addition to NoD, the IHDR also has a number of adjustable parameters by itself. In this paper, we have only considered tuning NoF, which is the number of examples that a node can have before being frozen. This is an important parameter, as it regulates the equilibrium point between the stability and plasticity of the classifier: If NoF is high, plasticity is favored, whereas more stable but less adaptable classifiers are obtained if low NoF values are used.

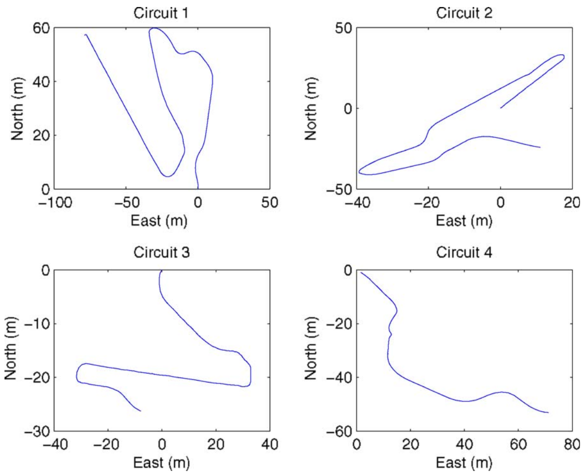


Fig. 2. Trajectories of the circuits employed in the experimental tests.

TABLE I
SUMMARY OF EXPERIMENTS AND CIRCUITS

	E1	E2	E3
Circuit 1	Training	Training	Validation
Circuit 2	Test	Training	Validation
Circuit 3	Validation	Test	Test
Circuit 4	Validation	Validation	Training

IV. EXPERIMENTAL RESULTS

Four different circuits (the trajectories of which are shown in Fig. 2) were used in the three tests carried out. These circuits represent real road traffic conditions during approximately 10 min of driving. Global Navigation Satellite Systems (GNSSs) and odometry and inertial measurements were collected during these tests. The use of odometry and inertial measurements supplies continuous positioning, even in cases without GNSS coverage [22].

The most common dead-reckoning deployment, consisting of an odometer and one gyro for heading estimation, has been complemented with one longitudinal accelerometer. Apart from its benefits for navigation purposes due to the typical problems of the odometry such as unequal wheel diameters or effective wheel diameter uncertainty, the use of one accelerometer results in particular convenience in the "stop & go" situations under consideration. In these cases, it is quite common that odometry systems suffer large transitory errors caused by slides and slips, which appear more often when the vehicle accelerates or decelerates. Moreover, without the support of the inertial measurements, the detection of longitudinal maneuvers would become too dependent on external factors, such as the friction between the road and the tires.

In addition to that, the inclusion of inertial measurements provides high-frequency measurements that result in being useful in representing the dynamics of road vehicles.

Table I summarizes the experiments. The use of different data sets for training, evaluation, and validation avoids possible negative influences in the algorithm response due to the data selection. In every experiment, the parameters are tuned in the training and test phases, whereas the validation phase shows the results obtained by the corresponding data set with those parameters. Validation results are therefore analyzed in this paper to check the system consistency. Before discussing the results, brief explanations of the OBE and the multisensor filtering used in the experiments are given.

Fig. 3. Vehicle prototype prepared by the research group of Intelligent Systems and Telematics (UMU).

A. OBE

Inertial sensors and an odometry captor are installed aboard the vehicle prototype. In addition, for navigation purposes, a GPS receiver and geographic information system maps were also included. Due to MEMS technology, low-cost inertial sensors can be considered, at the expense of higher measurement noise and low level of performance [23]. For vehicle-to-vehicle communications, wireless local area network ad hoc networks are used [24]. The sensors employed in the tests were EGNOS-capable GPS and differential GPS sensors by Novatel and Trimble, MEMs-based IMUs by Crossbow and Xsens, and the odometry of the vehicle.

Fig. 3 shows the vehicle prototype employed in the tests. A detail of the customized interface panel is shown in the lower left corner of the image.

B. Multisensor Data Fusion

To integrate the data coming from different sensors, a set of extended Kalman filters with different kinematic models, each of which was oriented to an intended maneuver, was run. This allows the application of the model that better represents the vehicle behavior at the current time and the improvement of the positioning and error estimations [13]. However, this issue is out of the intended scope of this paper and will not be discussed.

C. Results With IHDR

Circuit 1 used for validation is shown in Fig. 4. From top to bottom, the trajectory, the manually labeled maneuver classification, and the IMU and odometry measurements collected in this circuit are shown. The IHDR-based method has been tested using the experiments presented in Table I, and a summary of the results can be seen in Table II. We can appreciate the percentage of correct matches between the manual labeling and the IHDR results along the experiments. The parameters of the IHDR method for the experiments are given by the following, with NoD and NoF being the numbers of delays and freeze, respectively:

1) Experiment 1: NoD $= 35$ & NoF $= 60$;
2) Experiment 2: NoD $= 64$ & NoF $= 50$;
3) Experiment 3: NoD $= 32$ & NoF $= 10$.

At first sight, the solution appears consistent, with the percentage of matching between both methods being more than 70% in almost

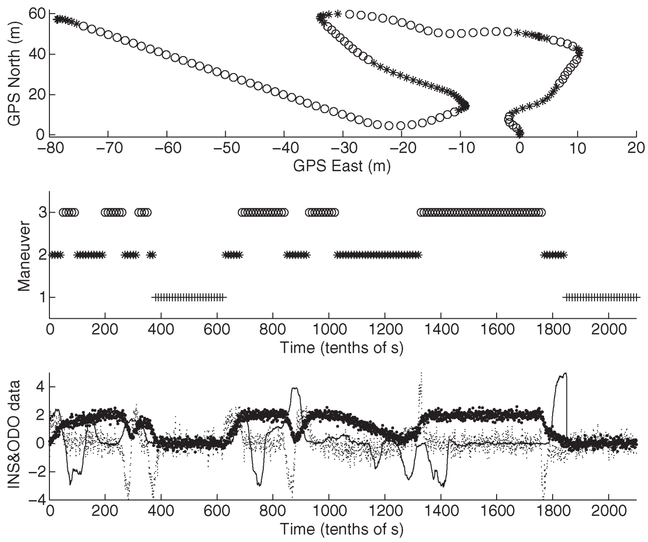


Fig. 4. Circuit 1. (Top) GPS trajectory of the circuit. (Middle) Manually labeled maneuver classification: AC ($*$), CR ($\circ$), and ST ($+$). (Bottom) Velocity value from the (thick line) odometry v and (dotted line) IMU a and (solid line) ω. Real values of a and ω have been scaled to fit in figure limits.

TABLE II
SUMMARY OF RESULTS OBTAINED WITH IHDR

	Experiments					
	E1		E2		E3	
Phase	Circ.	Res.(%)	Circ.	Res.(%)	Circ.	Res.(%)
Training	1	-	1&2	-	4	-
Test	2	74.74	3	76.44	3	56.39
	3	61.64	-	-	1	81.05
Validation	4	82.19	4	76.73	2	68.81
	3&4	70.20	-	-	1&2	76.15

TABLE III
PARAMETERS OF SUPERVISED dFasArt

Parameter	Value	Description
Aw	0.1	Time constant of weight's dynamic
Av	0.1	Time constant of weight's dynamic
Ac	0.1	Time constant of weight's dynamic
δ	0.2	Minimum fuzziness of the fuzzy categories
α	$1e^{-7}$	Activation value for new classes
RESET	0.19	Reset level
Ar	1.21	Time constant of RESET's dynamic
At	0.98	Time constant of activation's dynamic

all the tests and validations performed. The lowest values are obtained in those tests, as well as validations carried out with circuit 3, when circuit 2 was not used for training. What encourages thinking is that, during the course of circuit 3, the vehicle presented a dynamic state that is not found in circuits 1 and 4. On the other hand, when the algorithm is trained with a higher number of data (as is the case of E2), the results are hovering around the 76% value. This confirms the improvements achieved by the IHDR solution when the number of training data increases. Additionally, the value of NoD in the experiments, which is between 3.5 and 6.4 s., is consistent with the maneuvering times reported in the literature [25].

D. Results With SdFasArt

The same tests performed with the IHDR algorithm were carried out with the SdFasArt method under consideration. The tuning parameters and the results of SdFasArt can be seen in Tables III and IV,

TABLE IV
SUMMARY OF RESULTS OBTAINED WITH SdFasArt

	Experiments					
	E1		E2		E3	
Phase	Circ.	Res.(%)	Circ.	Res.(%)	Circ.	Res.(%)
Training	1	-	1&2	-	4	-
Test	2	85.07	3	62.71	3	61.57
	3	63.57	-	-	1	82.05
Validation	4	82.20	4	82.00	2	75.51
	3&4	71.29	-	-	1&2	79.34

TABLE V
CONFUSION MATRICES IN EXPERIMENT 1

	ST	AC	CR
ST	**501**	0	0
AC	31	**566**	142
CR	0	136	**724**

Circ. 1

	ST	AC	CR
ST	**105**	0	0
AC	28	**288**	78
CR	0	103	**798**

Circ. 2

	ST	AC	CR
ST	**100**	0	0
AC	68	**455**	66
CR	82	294	**335**

Circ. 3

	ST	AC	CR
ST	**200**	0	0
AC	40	**383**	118
CR	0	20	**239**

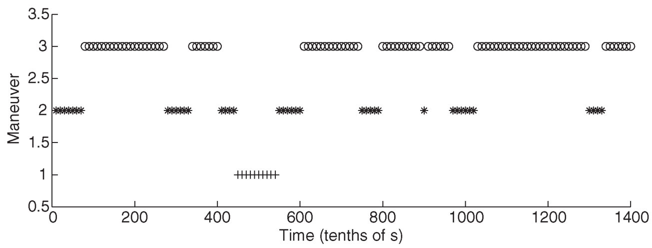
Circ. 4

respectively. By comparing the results obtained by IHDR with those achieved by SdFasArt, we can see that the latter offers better performance for Experiments 1 and 3, whereas in Experiment 2, results differ from the test phase to validation but still present good results. The only case when the SdFasArt approach performs worse is when the algorithms are trained with data coming from circuits 1 and 2 and tested with circuit 3, confirming the observation made in the previous section. This shows the importance of representative training data sets and the inclusion of all possible maneuvering dynamics on them. Comparing the results achieved by both methods, we can affirm that SdFasArt supplies good and consistent predictions. In addition, the computational cost of SdFasArt is much lower.

Paying attention to the confusion matrices of the different experiments, further conclusions may be obtained. Table V shows these values for Experiment 1. As can be seen, the units dedicated to interpret the stationary maneuvering state are totally clear. There were no cases during the experiments when a value of stationary in the manual labeling is wrongly matched by the algorithm. On the contrary, in some cases and, specifically, when validating using circuit 3, the predicted state is stationary, with the manual labeling pointed to the AC or CR state. Having a look at the situations in which these wrong matches appeared, we confirm that they correspond to states of very low motion, which are very hardly distinguishable from stationary states. Even in the manual labeling process, its categorization is ambiguous. The same situation can be seen in the confusions between the CR and AC maneuvering states. An also valid but different manual labeling would lead to better values in these matrices since very low dynamic changes can be categorized in both states. However, the significance of the results would not have changed due to this since, *de facto*, the categorization of one or another maneuvering states depends on the user criteria and the final application.

V. REMARKS ON THE RESULTS

Fig. 5 shows one of the validation results obtained by SdFasArt. As can be seen, the differences between the manual labeling and the results given by the proposed algorithm are mainly due to delays in the maneuver prediction (whether in the manual labeling or in the algorithm response) and some spurious detections. In a test carried out with circuit 1, the percentage of correct matches increases from



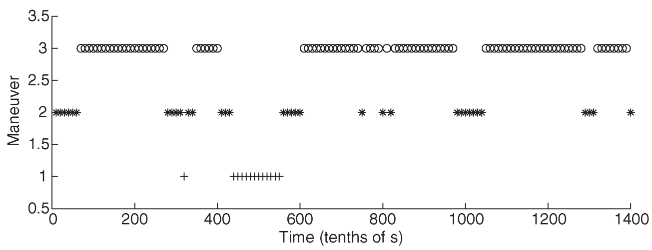


Fig. 5. Results on the validation circuit. (Top) Desired maneuver classification: AC ($*$), CR ($\circ$), and ST ($+$). (Bottom) Predicted maneuver classification: AC ($*$), CR ($\circ$), and ST ($+$).

85.28% up to 86.14% by simply applying a temporal window of 0.3 s, which is an appropriate value for the problem under consideration. Nevertheless, the majority of wrong matches appear in those instants in which it is not clear how to categorize the maneuver. In the same test, only 3.76% of the errors correspond to wrong matches of the algorithm, obtaining similar improvements in all the tests evaluated and, thus, giving a good idea of the actual consistency of the proposed method.

VI. Conclusion

In this paper, a supervised version of dFasArt has been proposed and tested for maneuver prediction of road vehicles. The combination of the dynamic character of dFasArt with a supervisory module has resulted in a robust classifier that is capable of providing stable outputs in spite of noisy time-varying input data. The neural architecture has been tested using real data gathered from inertial and odometry sensors mounted on a vehicle, showing good performance and consistent results.

The results were compared with those of an IHDR-based method, obtaining significant improvements, at lower computational charges. The consideration of different but valid manual classifications for the tests and validations may increase the performance of the system to values of 97% of correct predictions, with latency values of 0.3 s., which is an appropriate value for the problem under consideration. In addition, transitions from stationary to cruise-maneuvering states, or *vice versa*, are never predicted by the algorithm, showing the consistency of the maneuver categorization with the vehicle dynamics.

Future investigations into this topic will be dedicated to a more precise analysis of the activation units and lateral vehicle maneuvers, such as lane changes. In addition to that, further analysis in terms of accuracy enhancements and the quality of the error estimates is planned.

Finally, some other alternative (and possibly complementary) methods to SdFasArt and their combinations are also under consideration.

Acknowledgment

The authors would like to thank the people of the NEUROCOR Group for their support in this work. This work was partly carried out within the research group of Intelligent Systems and Telematics (University of Murcia, Murcia, Spain), which received an excellence research group award in the framework of the Spanish Plan de Ciencia y Tecnología de la Región de Murcia (04552/GERM/06).

References

[1] J. Cano-Izquierdo, M. Almonacid, J. Ibarrola, and M. Pinzolas, "Use of dynamic neuro fuzzy model dFasArt for identification of stationary states in closed-loop controlled systems," in *Proc. EUROFUSE*, Jaen, Spain, 2007.

[2] J. M. Cano-Izquierdo, M. Almonacid, M. Pinzolas, and J. Ibarrola, "dFasArt: Dynamic neural processing in FasArt model," *Neural Netw.*, vol. 22, no. 4, pp. 479–487, May 2008.

[3] S. Shen, L. Hong, and S. Cong, "Reliable road vehicle collision prediction with constrained filtering," *Signal Process.*, vol. 86, no. 11, pp. 3339–3356, Nov. 2006.

[4] A. H. Jamson, F. C. H. Lai, and O. M. J. Carsten, "Potential benefits of an adaptive forward collision warning system," *Transp. Res., Part C*, vol. 16, no. 4, pp. 471–484, Aug. 2008.

[5] G. Toulminet, M. Bertozzi, S. Mousset, A. Bensrhair, and A. Broggi, "Vehicle detection by means of stereo vision-based obstacles features extraction and monocular pattern analysis," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2364–2375, Aug. 2006.

[6] S. Vacek, S. Bergmann, U. Mohr, and R. Dillmann, "Fusing image features and navigation system data for augmenting guiding information displays," in *Proc. IEEE-MFI*, Heidelberg, Germany, 2006, pp. 323–328.

[7] J. Melo, A. Naftel, A. Bernardino, and J. Santos-Victor, "Detection and classification of highway lanes using vehicle motion trajectories," in *Proc. IEEE ITSC*, Toronto, ON, Canada, 2006, pp. 188–200.

[8] A. Amditis, A. Polychronopoulos, N. Floudas, and L. Andreone, "Fusion of infrared vision and radar for estimating the lateral dynamics of obstacles," *Inf. Fusion*, vol. 6, no. 2, pp. 129–141, Jun. 2005.

[9] J. C. McCall, D. P. Wipf, M. M. Trivedi, and B. D. Rao, "Lane change intent analysis using robust operators and sparse Bayesian learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 431–440, Sep. 2007.

[10] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Norwood, MA: Artech House, 1999.

[11] C. Hoffmann and T. Dang, "Cheap joint probabilistic data association filters in an interacting multiple model design," in *Proc. IEEE-MFI*, Heidelberg, Germany, 2006, pp. 197–202.

[12] D. Huang and H. Leung, "EM-IMM based land-vehicle navigation with GPS/INS," in *Proc. IEEE ITSC*, Washington, DC, 2004, pp. 624–629.

[13] R. Toledo, M. Zamora, B. Ubeda, and A. Skarmeta, "High-integrity IMM-EKF-based road vehicle navigation with low-cost GPS/SBAS/INS," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 491–511, Sep. 2007.

[14] K. Weiss, D. Stueker, and A. Kirchner, "Target modelling and dynamic classification for adaptive sensor data fusion," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2003, pp. 132–137.

[15] S. Hwan Park, W. Hyun Kwon, O.-K. Kwon, and P. So Kim, "Maneuver detection and target tracking using state-space optimal FIR filters," in *Proc. Amer. Control Conf.*, San Diego, CA, Jun. 1999, pp. 4253–4257.

[16] R. Toledo, M. Pinzolas, and J. M. Cano, "Supervised dFasArt: A neuro-fuzzy dynamic architecture for maneuver detection in road vehicle collision avoidance support systems," in *Proc. IWINAC*, vol. 4528, *LNCS*, La Manga del Mar Menor, Spain, 2007.

[17] W. S. Hwang and J. Weng, "Incremental hierarchical discriminating regression for indoorvisual navigation," in *Proc. IEEE Int. Conf. Image Process.*, 2001, pp. 810–813.

[18] J. Cano-Izquierdo, Y. Dimitriadis, E. Gómez, and J. Coronado, "Learning from noisy information in FasArt and FasBack neuro-fuzzy systems," *Neural Netw.*, vol. 14, no. 4/5, pp. 407–425, May 2001.

[19] G. Carpenter, S. Grossberg, N. Markuzon, J. Reynolds, and D. Rosen, "Fuzzy ARTMAP: A neural network architecture for incremental supervised learning 58 of analog multidimensional maps," *IEEE Trans. Neural Netw.*, vol. 3, no. 5, pp. 698–713, Sep. 1992.

[20] J. Weng and W. Hwang, "An incremental learning algorithm with automatically derived discriminating features," in *Proc. Asian Conf. Comput. Vis.*, Taipei, Taiwan, 2000, pp. 426–431.

[21] J. Weng and W. Hwang, "Online image classification using IHDR," *Int. J. Doc. Anal. Recognit.*, vol. 5, no. 2/3, pp. 118–125, Apr. 2003.

[22] R. Toledo, "A high integrity navigation system for road vehicles in unfriendly environments," Ph.D. dissertation, Univ. de Murcia Publishing, Murcia, Spain, 2005.

[23] B. Barshan and H. F. Durrant-Whyte, "Inertial navigation systems for mobile robots," *IEEE Trans. Robot. Autom.*, vol. 2, no. 3, pp. 328–342, Jun. 1995.

[24] R. Toledo, C. Sotomayor, and A. Gomez-Skarmeta, "Quadrant: An architecture design for intelligent vehicle services in road scenarios," in *Proc. Monograph Adv. Transp. Syst. Telematica*, 2006, pp. 451–460.

[25] N. Kaempchen, K. Weisst, M. Schaefer, and K. C. J. Dietmayer, "IMM object tracking for high dynamic driving maneuvers," in *Proc. IEEE IV Symp.*, Parma, Italy, 2004, pp. 825–830.

Supporting Agile Change Management by Scenario-Based Regression Simulation

Christian Berger and Bernhard Rumpe

***Abstract*—Many system-development projects today are mainly driven by the complexity of their software interacting with sensors or actuators in an embedded context. Autonomous vehicle development is a domain where it seems inevitably necessary to apply modern development techniques to cope with complexity, increase development efficiency, and ensure appropriate quality. Furthermore, changes that are triggered by customers or inventions of competitors, as well as bugs, enforce a comprehensible, if necessary, yet agile development process with stringent quality management. In this paper, we describe the agile efficiency- and quality-focused change management mainly based on scenario-driven regression simulation used in the CarOLO project for the development of an autonomously driving vehicle to compete in the 2007 Defense Advanced Research Projects Agency (DARPA) Urban Challenge program. The main contribution is the demonstration of the modern software engineering techniques' applicability to develop distributed embedded systems.**

***Index Terms*—Agile software engineering, autonomous driving, Caroline, CarOLO, change management, Defense Advanced Research Projects Agency (DARPA) urban challenge, intelligent algorithms, regression testing.**

I. Introduction and Motivation

More than one third of major project risks for software and system-development projects arise from late changes—either in a customer's needs or due to incorrect or incomplete requirements [1]. Further analysis of the Standish Group report yields that only 16.2% of the examined projects were in time and budget mainly because of the aforementioned reason. Five years later, that ratio has not improved significantly [2]. Therefore, the Standish Group suggests an iterative development process, including direct contact with the customer itself. Modern development processes for small-size to midsize projects like Extreme Programming (XP) [3] or the Rational Unified Process (RUP) [4] are directly based on an iterative principle considering that suggestion.

However, besides the iterative development process, further interpretation of the report yields that a consistent change-management process addressing the main risks of projects is necessary to complement an iterative development process itself. The main goal of the change-management process is to record and track change requests like rearrangements in a customer's needs and to schedule their processing in a regular iteration cycle.

Manuscript received March 31, 2008; revised July 20, 2009; accepted February 21, 2010. Date of publication March 29, 2010; date of current version May 25, 2010. The Associate Editor for this paper was L. Li.

The authors are with the Department of Software Engineering, RWTH Aachen University, 52074 Aachen, Germany (e-mail: berger@se.rwth-aachen.de; rumpe@se-rwth.de).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2010.2044571

Although agile development is mainly applied in business system development nowadays [5], it is still waiting for an adaption for embedded software development. One reason is the inherent need for documentation of quality-ensuring activities in embedded systems such as vehicles that are inherently incompatible with today's agile methods. Another maybe more important reason is the pretended inapplicability of agile development to traditional engineering techniques as known by mechanical and electrical engineers. Although very successful in their domains, these processes do not match very well, and thus, an alignment of these two different styles of development needs to be carefully realized. In particular, research projects, like the development of autonomous vehicles, are particularly suitable in demonstrating how agile development of complex embedded systems is possible when relying on an appropriate tool support.

We use the 2007 Defense Advanced Research Projects Agency (DARPA) Urban Challenge as an example to illustrate the agile change management process in the CarOLO project. Besides being able to add and adapt technology as well as architecture in an agile way, we were able to keep track of problems, ideas, and concepts while working with only a limited set of personal resources available. Thus, for keeping the tight schedule in the competition, we applied an agile change management for processing change requests.

To embrace changes, we were forced to preserve the system's quality without involving manual testing procedures all the time. We, thus, realized an automatic regression testing system that uses CxxTest [6] together with unattended and automatic system simulations allowing the software to virtually unattendedly drive complex scenes for the so-called *regression simulation*.

Having decided to use an agile development process, which uses and adapts substantial elements from XP and Scrum [7], we also had to keep in mind that although developers could meet regularly, we had to tackle the split-up of the development sites in Germany and the U.S. We, thus, used our change management process and tooling to handle globally distributed software development for ensuring the quality of software contributed from different sites—sometimes without being tested on the vehicle before being committed to the versioning system because of the limited availability of the vehicle itself, due to either a tight schedule, or the physical presence on the other continent.

In the following, we describe the competition and our contribution, followed by an overview of the change management that is used in the project. Furthermore, we show the integration of system simulation in our development and the change management process by outlining the software architecture and briefly describing our regression simulation framework. Finally, we show the applicability of our change-management process during the regular development and during the semifinal of the 2007 DARPA Urban Challenge.

II. CarOLO Project

The CarOLO project run by the Technische Universität Braunschweig developed an autonomously driving vehicle called "Caroline" for the 2007 DARPA Urban Challenge. In the following, the competition and our vehicle Caroline are described.

A. 2007 DARPA Urban Challenge

The 2007 DARPA Urban Challenge was the continuation of the well-known Grand Challenge series from 2004 and 2005. The goal in the DARPA Grand Challenges was to autonomously drive through an *a priori* unknown terrain by mainly following a route of predefined Global Positioning System waypoints. The challenge was to safely navigate through the unknown and rough terrain with only stationary