2013 IEEE/RSJ International Conference on
Intelligent Robots and Systems (IROS)
November 3-7, 2013. Tokyo, Japan

Vehicle Trajectory Prediction based on Motion Model and Maneuver Recognition

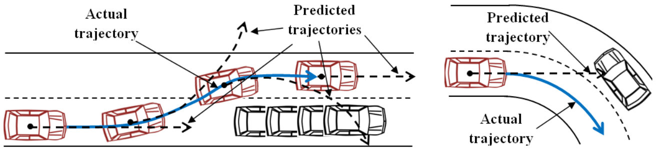
Adam Houenou*, Philippe Bonnifait*, Véronique Cherfaoui*, Wen Yao+

***Abstract*—Predicting other traffic participants trajectories is a crucial task for an autonomous vehicle, in order to avoid collisions on its planned trajectory. It is also necessary for many Advanced Driver Assistance Systems, where the ego-vehicle's trajectory has to be predicted too. Even if trajectory prediction is not a deterministic task, it is possible to point out the most likely trajectory. This paper presents a new trajectory prediction method which combines a trajectory prediction based on Constant Yaw Rate and Acceleration motion model and a trajectory prediction based on maneuver recognition. It takes benefit on the accuracy of both predictions respectively a short-term and long-term. The defined Maneuver Recognition Module selects the current maneuver from a predefined set by comparing the center lines of the road's lanes to a local curvilinear model of the path of the vehicle. The overall approach was tested on prerecorded human real driving data and results show that the Maneuver Recognition Module has a high success rate and that the final trajectory prediction has a better accuracy.**

I. Introduction

Active safety systems and self-driving cars are a promising solution to reduce the number of traffic accidents ([1], [2]). Some Advanced Driver Assistance Systems (ADAS) such as Adaptive Cruise Control, Collision Warning System and Emergency Braking System, that already exist in series vehicles, are able to warn the driver and even to intervene on the state of the vehicle when a hazardous traffic situation is being developed. A Collision Avoidance System (CAS) needs to continuously make a prediction of the evolution of the scene, in order to detect any possible future collision with the ego-vehicle. This means that it is necessary to predict the trajectory of detected vehicles in the surroundings of the ego-vehicle and its own trajectory in the case of ADAS.

Predicting the trajectory of a vehicle is not a deterministic task since it depends on each driver's intention and driving habits. However, certain considerations about vehicle dynamics can provide partial or fuzzy knowledge on the future. For instance it is known that a vehicle moving at a given speed will need a certain time to fully stop and that the curvature of its trajectory has to be under a certain value in order to keep stability. On the other hand, even if each driver has its own habits, it is possible to point out some common driving maneuvers based on traffic rules for instance, or on some common behaviors assuming that every driver keeps some comfort while driving.



(a) Lane changing (b) Entering a bend

Figure 1: Bad trajectory predictions.

In many cases, trajectory prediction is made by assuming a certain motion model. A comparison of different motion models for target tracking was made in [3]. It appears that the model assuming Constant Yaw Rate and Acceleration (CYRA) gives the best results. This model was used in [4] and [5] for vehicle trajectory prediction. So predicted trajectories are very accurate if the vehicle has a monotonic movement that perfectly fits the motion model. For a varying movement, the accuracy is good only for a short time ahead due to inertia, but the prediction can be pretty wrong for longer term (see Fig.1). A CAS needs to predict trajectories for at least a few seconds ahead and according to [6], 50% of accidents occur at intersections or during specific maneuvers where the dynamic of the vehicle could rapidly change.

In several works, a Maneuver Recognition Module (MRM) is used in order to have a better prediction considering the whole prediction. In [7] and [8], a set of trajectories corresponding to different maneuvers is predefined. Then, a Hidden Markov Model is used to select the most likely trajectory of an object, based on its current measurements sequence. In [9] Probabilistic Finite-State Machines are used to model complex driving maneuvers as sequences of basic elements that are specified by a set of rules in a fuzzy logic system. The rules are obtained from a training data set including signals such as velocity, acceleration and steering angle. A Bayes filter approach is employed to recognize a driving maneuver by computing the probability of each basic element in the context of the maneuver model. In [10] the ego vehicle trajectory is predicted by defining a *driving context* which is a vector containing relevant signals from sensors of the testbed such as light indicator or driver's gaze direction. The driving context is continuously recorded over a sliding time frame of 2 seconds and a trained classifier discriminates between *lane change* and *lane keeping* a few seconds before

*Heudiasyc UMR CNRS 7253, Univ. of Techn. of Compiègne, France
+State Key Lab of Machine Perception (MOE), Peking Univ., China
This work is partially supported by the NSFC-ANR Grant (No.61161130528)
adam.houenou@hds.utc.fr

978-1-4673-6357-0/13/$31.00 ©2013 IEEE

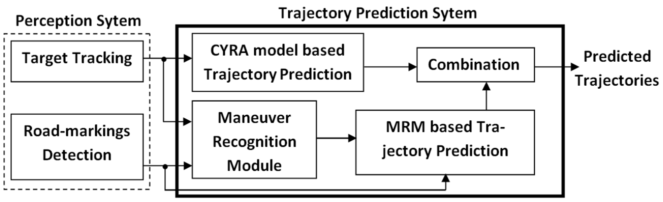

Perception Sytem
Target Tracking
Road-markings Detection
Trajectory Prediction Sytem
CYRA model based Trajectory Prediction
Combination
Maneuver Recognition Module
MRM based Trajectory Prediction
Predicted Trajectories

Figure 2: System overview

the maneuver starts.

This paper proposes a trajectory prediction method that combines a trajectory predicted by CYRA motion model and a trajectory predicted by maneuver recognition. The combination rule takes benefit on the short-term accuracy of the first trajectory and the better accuracy of the second trajectory at longer term. The defined Maneuver Recognition Module (MRM) uses no training data. It is based on the modeling and the comparison of the instantaneous path of the vehicle and the shape of the road. The proposed method is applicable as well for the target vehicles as for the ego-vehicle. In Section II, an overview of the trajectory prediction system is shown. Section III presents the MRM. In section IV, the proposed method is explained in detail. Some experimental results and analyzes are given in section V. Finally, section VI presents our conclusions and future work.

II. System overview

It is assumed that a target tracking system hosted by the ego-vehicle provides for each target vehicle the state vector $\zeta^{(target)}$ and its covariance matrix in a local Cartesian coordinate system:

$$\zeta^{(target)} = [x, y, \theta, v, a, \omega]^T \tag{1}$$

where x and y are the Cartesian coordinates, θ the yaw angle, v and a the longitudinal velocity and acceleration and ω the yaw rate.

For one prediction operation, the working frame is static and corresponds to the current measurement frame. With the same parametrization, the ego-vehicle's state is thus defined as:

$$\zeta^{(ego)} = [0, 0, 0, v, a, \omega]^T \tag{2}$$

where v, a and ω are provided by proprioceptive sensors.

It is also assumed that a camera based system detects road-markings and provides a local parabolic model of their center lines [11] in the same Cartesian frame :

$$y\left(x\right) = c_2 x^2 + c_1 x^1 + c_0 \tag{3}$$

where c_2, c_1 and c_0 are coefficients. The width of the lanes is also measured. The tracker and the road-markings detection system are called the Perception System in the following.

For each vehicle, including the ego-vehicle, the MRM detects the current maneuver. Then, a first trajectory prediction is made, only based on the recognized maneuver. A second prediction is made by using CYRA motion model. The final predicted trajectory is obtained by combining those two predictions with a weighting-function (Fig 2).

III. Maneuver Recognition Module (MRM)

In a normal driving context (e.g. no control loss), the path of a vehicle depends on the maneuvers decided by the driver. These can roughly be limited to these canonical cases:

- *Keep lane*
- *Change lane* (to the right or the left side lane) on the same road
- *Turn* (at an intersection).

A u-turn maneuver is excluded here. An overtaking maneuver can be decomposed as: *change lane - keep lane - change lane*. As it can be noticed, the maneuver is strongly relative to the lanes of the road and so is the path of the vehicle. The MRM is based on an early detection of the lane where the driver is intending to go (or stay). The lane intention detection exploits the current similarity between the path of the vehicle under consideration and the lanes' center lines (the one of its current lane and the ones of the adjacent lanes).

To evaluate the current similarity between the path of the vehicle and a given lane's center line, the path of the vehicle is locally modeled as an arc of circle [12] defined by the state vector $X^{(p)} = [d_l, d_r, \theta, \gamma]^T$. d_l and d_r are respectively the current distances of the the vehicle to the left and right boundaries of the considered lane. Their values are calculated from the lane's center line equation, the current position of the vehicle and the known width of the lane. Appendix-A shows how to compute the distance between a point and a line. θ is the current heading angle of the vehicle (known from state ζ) and γ the current curvature of its path. The estimated value of γ is obtained from Eq. 4, where v and ω are the velocity and the yaw rate of the vehicle.

$$\gamma = \frac{\omega}{v} \tag{4}$$

The variances of d_l, d_r and θ are retrieved from the covariance matrix of ζ. According to [13], if v and ω are Gaussian distributed then the variance of γ can be approximated by Eq. 5, where σ denotes the standard deviation of the subscript parameter.

$$\sigma_\gamma^2 = \left(\frac{\omega.v}{v^2 - \sigma_v^2}\right)^2 - \frac{\omega^2 - \sigma_\omega^2}{v^2 - \sigma_v^2} \tag{5}$$

The center line of the considered lane is also represented with the same state parameters: $X^{(l)} = [d_l, d_r, \theta, \gamma]^T$, with d_l and d_r both equaling half the lane's width. θ and γ are the lane center line's heading angle and its curvature in its closest point to the vehicle's position. Appendix-A shows how to define this point. In a given abscissa x, θ and γ are calculated with Eqs. 6 and 7 where $y\left(x\right)$ is the known equation of the line (Eq. 3).

$$\theta\left(x\right) = \arctan\left(y'\left(x\right)\right) \tag{6}$$

$$\gamma\left(x\right) = \frac{y''\left(x\right)}{\left(1 + y'^2\left(x\right)\right)^{3/2}} \tag{7}$$

where $y'(x)$ denotes the derivative of $y(x)$. Then, the square of the statistical distance between a given lane's center line and a given vehicle path is calculated with Eq.8 where $P^{(\bullet)}$ denotes the covariance matrix of state $X^{(\bullet)}$.

$$D^2 = \left(X^{(l)} - X^{(p)}\right)^T \cdot \left(P^{(l)} + P^{(p)}\right)^{-1} \cdot \left(X^{(l)} - X^{(p)}\right) \quad (8)$$

D^2 is Chi-square distributed. If its value is small, then the path of the vehicle is assumed to be currently quite similar to the lane's center line. In order to avoid disturbance due to punctual erroneous measurements, the distance considered to evaluate the similarity is a weighted average of so defined statistical distance, over the N last sample times (Eq.9).

$$D_k^* = \frac{\sum_{i=0}^{N-1} w_i D_{k-i}}{\sum_{i=0}^{N-1} w_i} \quad (9)$$

However, the weights w_i are set to be smaller for older sampled times. N is set so as not to use measurements older than one sec and to keep reasonable computational requirements. The distances D_{k-i} are calculated with Eq.8.

The intended lane of the vehicle is detected by first calculating the distance between its path and its current lane's center line. If the distance is lower or equal to a threshold T_h, the vehicle is supposed to be following that lane and this is a *keep lane* maneuver. The value of the threshold depends on the accuracy of the measurements and has to be tuned experimentally so as to give realistic results (a use case is shown is section V). If the distance is above the threshold, then the vehicle is either going to leave its current lane or has just entered it. In the first case, the distance with that lane must be increasing and in the second case, it must be decreasing. In order to be able to check the sens of variation of the distance it is necessary to buffer the calculated distances for at least one iteration. If the distance is increasing, then the vehicle is leaving its current lane and the intending lane is the one having the smallest distance, apart from the vehicle's current lane. Here, the maneuver is a *change lane,* if the intended lane is an adjacent one otherwise, it is a *turn* if the intended lane is a connected lane at an intersection. If the distance is decreasing, then the vehicle is entering its current lane and the maneuver is a *keep lane* (even if it is the second part of an actual lane change).

Notice that at the beginning of a *change lane* for instance, the MRM tends to output a *keep lane* until the distance between the vehicle's path and its current lane's center line exceeds the chosen threshold. This may induce a short delay in the detection of new started maneuvers (See section V). The lower the threshold, the lower the delay but a too low threshold will cause instability in case of zigzags within the lane or because of measurement noise.

For the sake of clarity, only *change lane* and *keep lane* maneuvers will be considered in the following. However, the proposed approach is not significantly different for a *turn* maneuver.

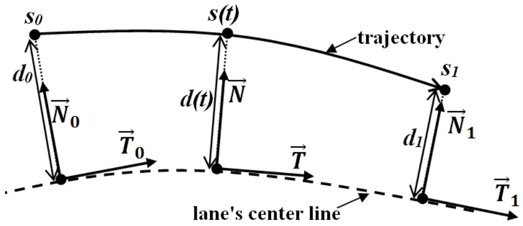


Figure 3: A trajectory in the Frenet frame

IV. Trajectory prediction

The method consists in mixing trajectory prediction based on maneuver recognition and trajectory prediction based on a motion model.

A. Trajectory prediction based on maneuver recognition (T_{man})

The MRM detects the current maneuver but there are many possible realizations for a single maneuver. Depending on the driver's habits, the actual trajectory may be pretty smooth or pretty aggressive. Moreover, the road geometry will also have an influence. So, based on the vehicle current state, the road parameters and the detected maneuver, a set of trajectories are first generated and the best one is selected with respect to a cost function described later. The trajectories are first generated in the Frenet frame along the center line of the current lane of the vehicle (see Fig.3), then converted to the initial Cartesian coordinate system. We know from [14] and [15] that the lateral component $d(t)$ and the longitudinal component $s(t)$ (t being the time) of the trajectory of a vehicle moving from the initial state $F_0 = \left(s_0, \dot{s}_0, \ddot{s}_0, d_0, \dot{d}_0, \ddot{d}_0\right)$ in the Frenet frame to the final state $F_1 = \left(s_1, \dot{s}_1, \ddot{s}_1, d_1, \dot{d}_1, \ddot{d}_1\right)$ can each be optimally modeled as a quintic polynomial. This guaranties the jerk continuity and provides a unique solution.

1) Initial state of the trajectories in the Frenet frame: All the trajectories have the same initial state which is derived from the current state $\zeta_0 = [x_0, y_0, \theta_0, v_0, a_0, \omega_0]$ of the vehicle in the Cartesian frame. Eq.10 shows the transformation.

$$\begin{cases} d_0 & = d_0^* \\ \dot{d}_0 & = v_0 sin\left(\theta_0 - \theta_{\overrightarrow{T_0}}\right) \\ \ddot{d}_0 & = \sqrt{(a_0^2 + \gamma_0 v_0^2)} sin\left(\theta_0 - \theta_{\overrightarrow{T_0}}\right) \\ s_0 & = 0 \\ \dot{s}_0 & = v_0 cos\left(\theta_0 - \theta_{\overrightarrow{T_0}}\right) \\ \ddot{s}_0 & = \sqrt{(a_0^2 + \gamma_0 v_0^2)} cos\left(\theta_0 - \theta_{\overrightarrow{T_0}}\right) \end{cases} \quad (10)$$

d_0^* is the Euclidean distance between point (x_0, y_0) and the lane center line, $\theta_{\overrightarrow{T_0}}$ is the orientation of the tangent vector $\overrightarrow{T_0}$ depicted in Fig.3. Appendix-A describes the calculation of d_0^* and $\overrightarrow{T_0}$. The initial curvilinear abscissa s_0 is arbitrarily set to zero. $\gamma_0 v_0^2$ is the current value of the normal acceleration of the vehicle.

2) Final state of the trajectories in the Frenet frame: Since the trajectory prediction is made in order to detect possible collisions, no constraint about obstacles is taken into account. Only the following assumptions are made: at the end state, the vehicle is moving right on the center line of its intended lane (known from the MRM) and has a constant longitudinal acceleration during the maneuver. Thus, only partial knowledge is available about the final state (see Eq.11).

$$\begin{cases} d_1 & = d_1^* \\ \dot{d}_1 & = 0 \\ \ddot{d}_1 & = 0 \\ \ddot{s}_1 & = a_0 \end{cases} \tag{11}$$

For a *change lane*, d_1^* equals plus/minus the lane's width depending on the direction of the maneuver and is null for a *keep lane*. A complete *change lane* maneuver has a limited duration denoted $t^{(K)}$. We know from the experiments described in section V that $t^{(K)} \approx 6sec$. One may extend or reduce this duration if needed. That does not affect the proposed method but may change the computational requirements. A *keep lane* is obviously shorter. The time interval $\left]0, t^{(K)}\right]$ is then sampled and each sample time is used as maneuver ending time t_1 to define a unique trajectory. The set of trajectories originates from this sampling: $t_1 = \left\{t^{(i)}\right\}_{i=1..K}$. So, an additional piece of information is: $\dot{s}_1 = v_0 + a_0 \cdot t_1$.

3) Lateral component description: The lateral component of each trajectory is of the form:

$$d(t) = c_5 t^5 + c_4 t^4 + c_3 t^3 + c_2 t^2 + c_1 t + c_0 \tag{12}$$

Where $c_{i,i=\{0,1,2,3,4,5\}}$ are coefficients. Given a starting time $t_0 = 0$, a defined ending time $t_1 \in \left]0, t_{max}\right]$ and knowing the initial and final states, the coefficients $c_{i,i=\{0,1,2,3,4,5\}}$ are easily obtained by solving Eq.13

$$\begin{bmatrix} t_0^5 & t_0^4 & t_0^3 & t_0^2 & t_0^1 & 1 \\ t_1^5 & t_1^4 & t_1^3 & t_1^2 & t_1^1 & 1 \\ 5t_0^4 & 4t_0^3 & 3t_0^2 & 2t_0^1 & 1 & 0 \\ 5t_1^4 & 4t_1^3 & 3t_1^2 & 2t_1^1 & 1 & 0 \\ 20t_0^3 & 12t_0^2 & 6t_0^1 & 2 & 0 & 0 \\ 20t_1^3 & 12t_1^2 & 6t_1^1 & 2 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} c_5 \\ c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix} = \begin{bmatrix} d_0 \\ d_1 \\ \dot{d}_0 \\ \dot{d}_1 \\ \ddot{d}_0 \\ \ddot{d}_1 \end{bmatrix} \tag{13}$$

4) Longitudinal component description: Since s_1 is unknown, we loose one degree of freedom and the longitudinal component of the trajectories will be modeled as a quartic polynomial.

$$s(t) = c_4 t^4 + c_3 t^3 + c_2 t^2 + c_1 t + c_0 \tag{14}$$

Where $c_{i,i=\{0,1,2,3,4\}}$ are constant coefficients. The jerk continuity is still guarantied since the polynomial is a least three times derivable. The coefficients $c_{i,i=\{0,1,2,3,4\}}$ are obtained by solving Eq.15.

$$\begin{bmatrix} t_0^4 & t_0^3 & t_0^2 & t_0^1 & 1 \\ 4t_0^3 & 3t_0^2 & 2t_0^1 & 1 & 0 \\ 4t_1^3 & 3t_1^2 & 2t_1^1 & 1 & 0 \\ 12t_0^2 & 6t_0^1 & 2 & 0 & 0 \\ 12t_1^2 & 6t_1^1 & 2 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} c_4 \\ c_3 \\ c_2 \\ c_1 \\ c_0 \end{bmatrix} = \begin{bmatrix} s_0 \\ \dot{s}_0 \\ \dot{s}_1 \\ \ddot{s}_0 \\ \ddot{s}_1 \end{bmatrix} \tag{15}$$

Notice that by dealing separately with longitudinal and lateral components, we do not take into account the non holonomic constraints, which means that this approach cannot be applied to very low speed scenarios. A set of trajectories is obtained, each corresponding to a different ending time (Fig.4).

The trajectories are then converted to the Cartesian coordinate system (see Appendix-B) and the best one is selected with respect to the cost function described hereafter.

5) Best trajectory selection by minimization of a cost function: It is admitted that an average driver will seek to minimize the duration of his current maneuver but will also try to keep some comfort in the cockpit and avoid oscillations and overshoots. The trajectories generated here (denoted $T^{(i)}_{i=1..K}$) have no oscillation but may have overshoots depending on the initial state (see Fig.4) and the road's geometry; which is a realistic fact. Since there is no oscillation and constant longitudinal acceleration is assumed, the comfort is only quantified by the normal acceleration during the maneuver. One can notice in Fig.4 that the trajectories with overshoots are also the longer ones. So, the cost of a trajectory is defined as shown in Eq.16.

$$C\left(T^{(i)}\right) = max\left(\bar{a}\left(t\right)\right) + \alpha \cdot t^{(i)} \tag{16}$$

where $\bar{a}\left(t\right)$ is the norm of the normal acceleration of the vehicle at a given time t during the maneuver, $t^{(i)}$ is the duration of the trajectory and α is a positive weighting coefficients which is used to vary the behavior of the cost function (a use case is shown in section V). This function penalizes trajectories having a long duration and by the way also penalizes trajectories having overshoots. Trajectories with a high normal acceleration peak are also penalized. T_{man} is the trajectory having the smallest cost:

$$T_{man} = \arg\min\left(C\left(T^{(i)}\right)\right)_{i=1..K} \tag{17}$$

Instead of considering the maximum value of the normal acceleration in Eq.16, one may think of computing its average value on the whole maneuver but the fact is that some of the generated trajectories having a very small duration but abnormally high peak of normal acceleration, could then be selected.

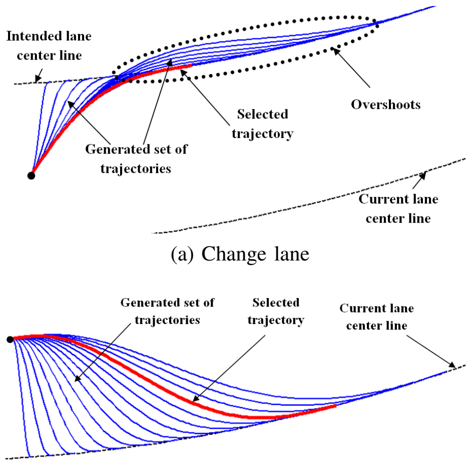
B. *Trajectory prediction with motion model (T_{mdl})*

Assuming CYRA motion model, the components of the velocity along each dimension in the Cartesian frame are

$$\begin{cases} v_x\left(t\right) = & v\left(t\right) \cdot cos\left(\omega_0 \cdot t + \theta_0\right) \\ v_y\left(t\right) = & v\left(t\right) \cdot sin\left(\omega_0 \cdot t + \theta_0\right) \end{cases} \tag{18}$$

where $v(t) = a_0 \cdot t + v_0$. T_{mdl} is obtained on a closed form by integrating the velocity.

$$T_{mdl} = \begin{cases} x\left(t\right) = & \frac{a_0}{\omega_0^2} cos\left(\theta\left(t\right)\right) + \frac{v(t)}{\omega_0} sin\left(\theta\left(t\right)\right) + c_x \\ y\left(t\right) = & \frac{a_0}{\omega_0^2} sin\left(\theta\left(t\right)\right) - \frac{v(t)}{\omega_0} cos\left(\theta\left(t\right)\right) + c_y \end{cases} \tag{19}$$



(a) Change lane

(b) Keep lane

Figure 4: Trajectory generation and selection.

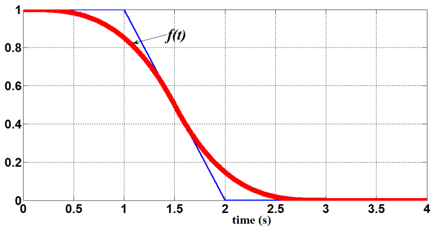


Figure 5: Weight function $f(t)$.

where c_x and c_y are constants fixed with the initial values.

$$\begin{cases} c_x = & x_0 - \frac{v_0}{\omega} sin\left(\theta_0\right) - \frac{a_0}{\omega_0^2} cos\left(\theta_0\right) \\ c_y = & y_0 + \frac{v_0}{\omega_0} cos\left(\theta_0\right) - \frac{a_0}{\omega_0^2} sin\left(\theta_0\right) \end{cases} \tag{20}$$

If $\omega_0 = 0$, the predicted trajectory is rectilinear and given by

$$T_{mdl} = \begin{cases} x\left(t\right) = & \left(\frac{1}{2} \cdot a_0 \cdot t^2 + v_0\right) cos\left(\theta_0\right) + x_0 \\ y\left(t\right) = & \left(\frac{1}{2} \cdot a_0 \cdot t^2 + v_0\right) sin\left(\theta_0\right) + y_0 \end{cases} \tag{21}$$

C. Combining T_{man} and T_{mdl}

As reminded earlier in Section I, T_{mdl} is very accurate only for a short term. For a longer terms and for specific maneuver the motion model does not fit the actual movement. T_{man} is based on maneuver detection and is thus more adapted to longer time prediction than T_{mdl}. So, the final predicted trajectory T_{fin}, is a weighted sum of T_{mdl} and T_{man} as described in Eq.22 where $f(t)$ is a cubic spline function depicted in Fig.5.

$$T_{fin}\left(t\right) = f\left(t\right) \cdot T_{mdl}\left(t\right) + \left(1 - f\left(t\right)\right) \cdot T_{man}\left(t\right) \tag{22}$$

For the first second of prediction, T_{fin} is closer to T_{mdl}. As the prediction time horizon gets longer, T_{fin} gets progressively closer to T_{man}. With this function, not only the accuracy of T_{mdl} for short time horizon is kept, but also, the possible delay in the detection of a new started maneuver is partially covered.

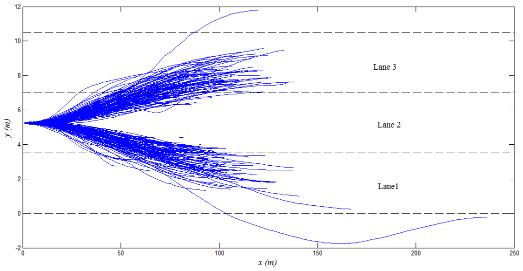


Figure 6: Set of 236 human lane change trajectories

V. Experimental results

A prerecorded human driving data in semi-urban conditions was used to test the maneuver recognition algorithm and the trajectory prediction method. The acquisition platform was POSS-V (PKU Omni Smart Sensing - Vehicle). The integrated system recorded position data of the host vehicle with timestamps as explained in [16]. The experiment was carried on the $3^{rd}$and 4^{th} ring roads of Beijing, China. Drivers were asked to change lanes or to overtake other vehicles when possible, on straight road portions. Then, the lane change maneuver trajectories have been extracted from the recordings and conditioned to fit the same Cartesian coordinate system as depicted in Fig.6. The standard deviation of the localization system is 2m and the vehicle drove approximately at 80km/h.

As it can be noticed, the lane change trajectory extraction is not perfect but most of the extracted trajectories were actual lane changes. All the trajectories start on lane 2. There are 91 right lane changes (RLC), 119 left lane changes (LLC) and 24 trajectories that are either incomplete lane change or unknown maneuvers. The later are not considered in the following.

Since the available data format is not rich enough to fit the requirements of the proposed method, the data set has been preprocessed in order to retrieve the missing information such as yaw angle, velocity, acceleration, yaw rate with their respective variances. The preprocessing consists in performing a Kalman filtering with a CYRA motion model for the prediction steps.

A. Maneuver recognition

The MRM was executed for each sampled data of each extracted trajectory. The goal here is to check if the right maneuver is always detected before the vehicle actually reaches its intended lane and to report the time before the detection and the lateral offset between its initial position and its position at the moment of detection. Then, the mean values were calculated for the set of extracted lane changes. After having tuned the threshold by using a few extracted trajectories, the results for all the 210 ones are shown in Tab.I.

All the lane changes have been detected before the vehicle reaches the intended lane. The mean time of detection is about 1sec. Even if this may seem high, the low value of

	LLC	RLC
Detection	100%	100%
Mean time before detection	1.15s	1.09
Mean lateral offset before detection	0.3m	0.33m

Table I: Lane change maneuver detection and mean time of detection. With standard deviations: $\sigma_{d_l} = \sigma_{d_r} = 0.5$m, $\sigma_\theta = 5°$, $\sigma_\gamma = 0.05m^{-1}$ and $T_h = 2$.

	$[0s, 1s[$	$[1s, 2s[$	$[2s, 3s[$	$[3s, 4s[$
T_{mdl}	0.1m	0.49m	2.3m	4.31m
T_{man}	0.15m	0.2m	0.33m	0.45m
T_{fin}	0.09m	0.17m	0.28m	0.45m

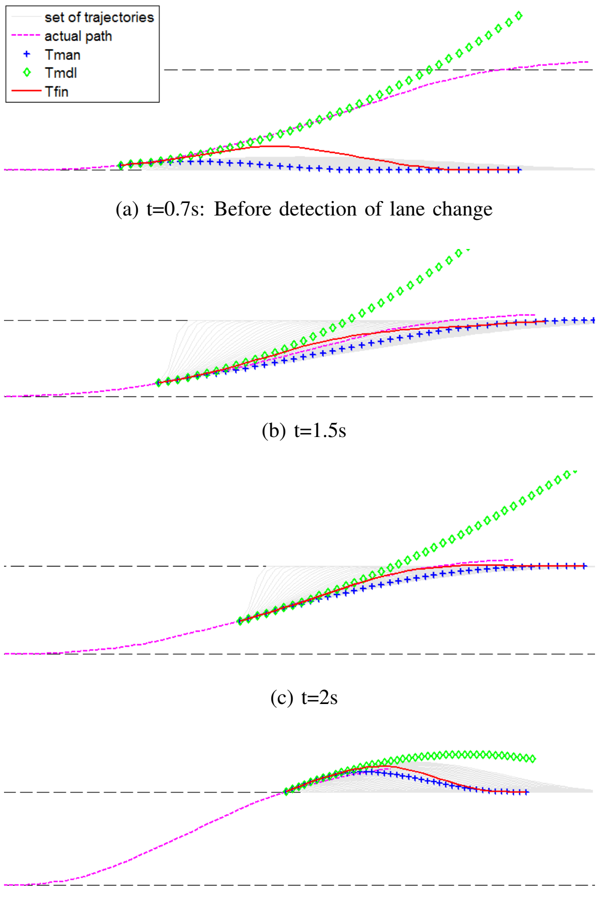
Table II: Accuracy of trajectory prediction

the mean lateral offset before detection indicates that this is because most of the trajectories were pretty smooth, meaning that most of the drivers slowly left the initial lane. For the most aggressive trajectory the reported values were 0.4s and 0.25m.

B. *Trajectory prediction*

The goal is to check the accuracy of the trajectory prediction for different time horizons: $[0s, 1s[$, $[1s, 2s[$, $[2s, 3s[$ and $[3s, 4s[$. Let's first consider only one extracted trajectory. For each sample time, a prediction was made for 4s ahead and for each future pose, the Euclidean distance between the actual position of the vehicle and the corresponding future pose according to the prediction, has been reported. The reports were classified into 4 categories depending on the corresponding time horizon. Then, the mean error of prediction for each time interval was computed as the average of the reported distances. Since the delay in the detection of the lane change at the beginning of the maneuver has an influence on the overall result, only the predictions after the lane change have been considered. The coefficients α in the cost function (see Eq.16) was set to 0.25. Tab.II shows the overall average values for all the extracted trajectories, per type of prediction.

The mean error for T_{mdl} is very low for short time predictions but exponentially increases as the time horizon gets bigger. The accuracy of T_{man} does not vary a lot and the values show that the trajectory prediction based only on maneuver recognition is already close to the actual trajectory. This means that the trajectory generation mechanism and the cost function for the best trajectory selection is relevant. For T_{fin}, in the intervals $[0s, 1s[$ and $[3s, 4s[$, the mean errors are respectively the same as for T_{mdl} and T_{man}. This behavior was expected because of the weighting function in Eq.22. In the intervals $[1s, 2s[$ and $[2s, 3s[$ the accuracy of T_{fin} is better than the accuracies of T_{mdl} and T_{man} because the actual trajectory is, most of the time, between T_{mdl} and T_{man} (see Fig.7 and 8). Globally speaking, T_{fin} is more accurate than T_{mdl} and T_{man} which means that the proposed weighting function and the overall approach are also relevant. The mean errors on the final prediction are very low and can thus fit the requirements of a collision detection system.



(a) t=0.7s: Before detection of lane change

(b) t=1.5s

(c) t=2s

(d) t=3.1s

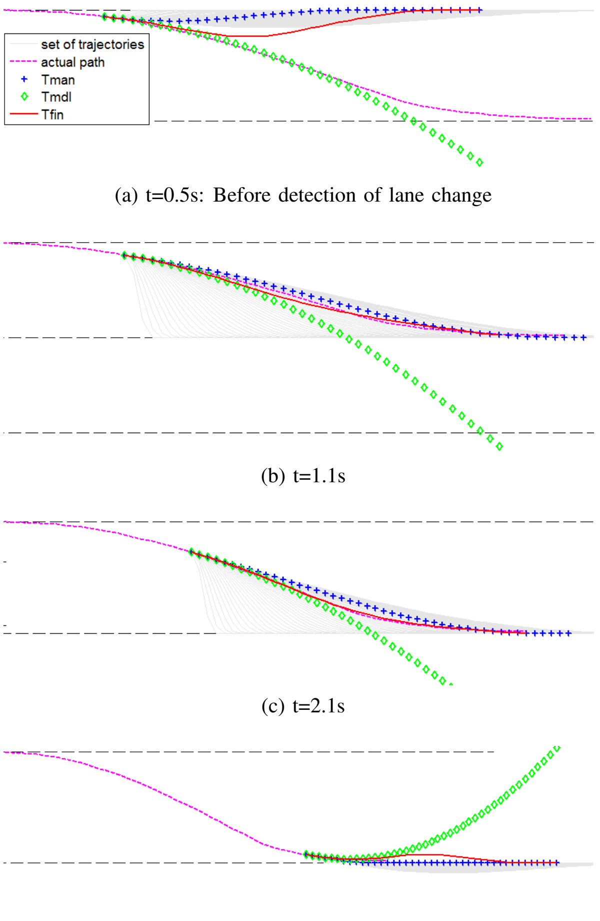
Figure 7: Example of predictions for a left lane change

VI. Conclusion and future works

In this paper, we have presented a new approach for trajectory prediction. The method includes a prediction based on CYRA motion model which is very accurate for a short term and a prediction based on maneuver recognition which is more adapted for longer term prediction. The experimental results on human real driving data proved the relevance of the method. The second contribution of the paper is a deterministic and efficient method for maneuver recognition. It is based only on kinematic measurements and road geometry detection. For real-time implementation, the complexity of the method can be kept low if the number of generated trajectories remains reasonable and if the curvature of the road is constant (in this case, the transformation from the Frenet frame to the Cartesian frame is trivial). Future works include the estimation of the uncertainty along the predicted trajectories in order to estimate the Time-To-Collision with an associated probability of collision for Collision Warning/Avoidance Systems.

References

[1] G. Rieger, J. Scheef, H. Becker, M. Stanzel, and R. Zobel, "Active safety systems change accident environment of vehicles significantly - a challenge for vehicle design," *International Technical Conference on the Enhanced Safety of Vehicles (ESV), Washington DC*, June 2005.



(a) t=0.5s: Before detection of lane change

(b) t=1.1s

(c) t=2.1s

(d) t=3.2s

Figure 8: Example of predictions for a right lane change

[2] G. Silberg and R. Wallace, “Self-driving cars : The next revolution,” white paper, KPMG LLP & Center of Automotive Research., June 2012.

[3] R. Schubert, E. Richter, and G. Wanielik, “Comparison and evaluation of advanced motion models for vehicle tracking,” *International Conference on Information Fusion, Cologne, Germany*, Jul 2008.

[4] A. Berthelot, A. Tamke, T. Dang, and G. Breuel, “Handling uncertainties in criticality assessment,” *IEEE Intelligent Vehicles Symposium (IV), Baden-Baden, Germany*, 2011.

[5] A. Tamke, T. Dang, and G. Breuel, “A flexible method for criticality assessment in driver assistance systems,” *IEEE Intelligent Vehicles Symposium Baden-Baden, Germany*, June 2011.

[6] A. M. Martinez, E. Carter, C. Naing, M. Simon, and T. Hermitte, “Accident causation and pre-accidental driving situations.,” tech. rep., Traffic Accident Causation in Europe (TRACE) & Information Society Technologies, Delivrable D 2.1, 2008.

[7] C. Laugier, I. E. Paromtchik, M. Perrollaz, J.-D. Yoder, C. Tay, M. Yong, A. Nègre, and K. Mekhnacha, “Probabilistic analysis of dynamic scenes and collision risks assessment to improve driving safety,” *IEEE Intelligent Transportation Systems Magazine*, vol. 3, pp. 4 – 19, 2011.

[8] C. Sung, D. Feldman, and D. Rus, “Trajectory clustering for motion prediction,” *IEEE/RJS International Conference on Intelligent Robots and Systems, Vilamoura, Portugal*, Oct 2012.

[9] T. Hülnhagen, I. Dengler, A. Tamke, T. Dang, and G. Breuel, “Maneuver recognition using probabilistic finite-state machines and fuzzy logic,” *IEEE Intelligent Vehicles Symposium, San Diego, CA, USA*, June 2010.

[10] B. Morris, A. Doshi, and M. Trivedi, “Lane change intent prediction for driver assistance: On-road design and evaluation,” *IEEE Intelligent Vehicles Symposium Baden-Baden, Germany*, pp. 5 – 9, June 2011.

[11] K. Kluge, “Extracting road curvature and orientation from image edge points without perceptual grouping into features,” *IEEE Intelligent Vehicles Symposium, Paris, France*, Oct 1994.

[12] D. Müller, J. Pauli, M. Meuter, L. Ghosh, and S. Müller-Schneiders, “A generic video and radar data fusion system for improved target selection,” *IEEE Intelligent Vehicles Symposiun (IV), Baden-Baden, Germany*, June 2011.

[13] J. Hayya, D. Armstrong, and N. Gressis, “A note on the ratio of two normally distributed variables.,” *Management Science*, Jul 1975.

[14] A. Takashi, T. Hongo, Y. Ninomiya, and G. Sugimoto, “Local path planning and motion control for agv in positioning,” *IEEE/RSJ International Workshop on Intelligent Robots ans Systems*, pp. 392 – 397, Sep 1989.

[15] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, “Optimal trajectory generation for dynamic street scenario in a frenet frame,” *IEEE International Conference on Robotics and Automation, Anchorage, Alaska, USA*, May 2010.

[16] W. Yao, H. Zhao, F. Davoine, and H. Zha, “Learning lane change trajectories from on-road driving data,” *IEEE Intelligent Vehicles Symposium, Alcala de Henares, Spain*, June 2012.

APPENDIX

A. Computation of d_0^* and $\overrightarrow{T_0}$ (section IV-A1)

The distance between a point and a curve is the smallest distance between this point and any point belonging to the curve. The square distance between point $X_0 = (x_0, y_0)$ and a point (x, y) belonging to the the lane's center line is $d^2 = (x - x_0)^2 + (y - y_0)^2$. Since we know the equation of the lane's center line as $y(x) = a_2x^2 + a_1x^1 + a_0$, the expression of d^2 can be developed and rewritten as

$$d^2 = b_4x^4 + b_3x^3 + b_2x^2 + b_1x + b_0x_0$$

where $b_{i|i=0..4}$ are constants values depending on x_0, y_0 and coefficients $a_{i|i=0..4}$. One can then easily find the abscissa x_a that minimizes d^2 (and by the way d) and find the corresponding point $A = (x_a, y_a)$ in the line be . So,

$$d_0^* = AX_0$$

$\overrightarrow{T_0}$ is the unit tangent vector of the lane's center line in point A. $y'(x_a)$ being the slope of the tangent of the line in A, we then have

$$\overrightarrow{T_0} = \left(\frac{1}{\sqrt{1 + y'^2(x_a)}}, \frac{y'^2(x_a)}{\sqrt{1 + y'^2(x_a)}} \right)$$

B. From Frenet frame to Cartesian $X = (s, d) \rightarrow X = (x, y)$

Let's consider the point A of the center line, defined in the Frenet frame as $(s, 0)$ and in the Cartesian frame as (x_a, y_a). Let $\overrightarrow{N}$ be the unit normal vector of the line in A and point O be the origin of the Cartesian frame. Assuming $y'(x_a) \neq 0$, we have

$$\overrightarrow{OX} = \overrightarrow{OA} + d \cdot \overrightarrow{N} \text{ with } \overrightarrow{N} = \left(\frac{1}{\sqrt{1 + 1/y'^2(x_a)}}, \frac{-1/y'(x_a)}{\sqrt{1 + 1/y'^2(x_a)}} \right)$$

So, finding x_a leads to defining X in the Cartesian frame. s is the distance from the origin of the Frenet frame to point A along the center line. Thus

$s = \int_{x_0}^{x_a} \sqrt{1 + y'^2(x)} dx$

Since $y(x)$ is a second order polynomial, it is not analytically possible to find a function f so that $x_a = f(s)$. However this can be solved numerically by sampling the X-axis and summing of elementary distances.