2012 Intelligent Vehicles Symposium
Alcalá de Henares, Spain, June 3-7, 2012

Probabilistic Trajectory Prediction with Gaussian Mixture Models

Jürgen Wiest[1], Matthias Höffken[1], Ulrich Kreßel[2], Klaus Dietmayer[1]

***Abstract*— In the context of driver assistance, an accurate and reliable prediction of the vehicle's trajectory is beneficial. This can be useful either to increase the flexibility of comfort systems or, in the more interesting case, to detect potentially dangerous situations as early as possible. In this contribution, a novel approach for trajectory prediction is proposed which has the capability to predict the vehicle's trajectory several seconds in advance, the so called long-term prediction. To achieve this, previously observed motion patterns are used to infer a joint probability distribution as motion model. Using this distribution, a trajectory can be predicted by calculating the probability for the future motion, conditioned on the current observed history motion pattern.**

The advantage of the probabilistic modeling is that the result is not only a prediction, but rather a whole distribution over the future trajectories and a specific prediction can be made by the evaluation of the statistical properties, e.g. the mean of this conditioned distribution. Additionally, an evaluation of the variance can be used to examine the reliability of the prediction.

I. INTRODUCTION

Intersections are still a big challenge for driver assistance systems. Statistics show that the most common types of car accidents occur if a car is turning into or crossing an intersection. To prevent this, Advanced Driver Assistance Systems (ADAS) have to cope with highly complex traffic scenarios and it is often not clear whether a car is crossing the intersection or the driver plans to turn. To handle such situations, the ADAS has to perform two main tasks: first, the system has to detect the objects in the current scene and second, it has to assess whether they are relevant or not. The sooner this assessment is possible, the better the situation can be handled. Up to now, the selection of potentially dangerous objects is restricted by the ability to predict the vehicle's trajectory. The trajectory itself contains not only the position where the car will be at a specific time in the future, the so called prediction horizon, it rather contains the exact course to the predicted position. This is equivalent to predicting the vehicle's state for several seconds into the future. Common methods for motion prediction are standard filter methods (e.g., Kalman filters) where a forecast of the vehicle's position is made by a recursive prediction of the system state to the next time step until the desired prediction horizon is reached. Depending on the accuracy and complexity of the underlying kinematic model, the model-based prediction (e.g., with the assumption of constant velocity and yaw angle) can result in huge deviations from the real trajectory and systematically fail, especially in the case of turning maneuvers.

This contribution presents a different approach. Instead of using a kinematic model, the trajectory prediction is understood as a machine learning problem. The main idea is to learn a motion model out of previously observed trajectories and build a functional mapping of the currently observed history to the most likely future trajectory (see Figure 1). This idea is extended to predict not only the most likely, but rather a whole probability distribution over the vehicle's future trajectories.

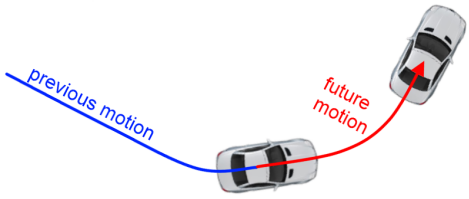


Fig. 1: Trajectory prediction aims to learn a functional mapping of the currently observed history to the most likely future of the vehicle motion.

There are not many previous works that deal with the problem of predicting the vehicles trajectories several seconds into the future. The authors in [3] use an explicit dynamic driver model where the future inputs are modeled as stochastic variables. They compute the likelihood for a potential maneuver and take this distribution to approximate future scenarios using Monte Carlo sampling to derive a statistical threat assessment. Similarly, in [13] a trajectory clustering framework with an efficient coarse-to-fine strategy is proposed. After a smoothing and feature extraction stage, they use a graph-theoretic clustering method called dominant-set clustering with which they generate fifteen prototypes out of 1200 trajectory samples.

To investigate a motion prediction algorithm, [12] applies clustering techniques to learn representative trajectories and uses those motion patterns as a motion model. Estimation of the future is done by calculating the likelihood of the partially seen trajectory belonging to any of the cluster centers and selecting the most probable solution.

In [5] human-motion prediction of a robotic system is performed by interpreting it as a time series. They compared several state-of-the-art algorithms for time series prediction such as Local Modeling, Cluster Weighted Modeling, Echo

[1]J. Wiest, Matthias Höffken and K. Dietmayer are with the Institute of Measurement, Control an Microtechnology, Ulm University, 89081 Ulm, Germany {firstname.lastname} at uni-ulm.de
[2]Ulrich Kreßel is with Daimler AG, Ulm, Germany ulrich.kressel at daimler.com

State Networks (ESN) and Autoregressive Models (AR). An evaluation with real-world trajectories of humans showed that Echo State Networks outperforms the others.

In the context of visual surveillance systems, [4] proposed a non-parametric long-term human-motion prediction framework. Gaussian process regression is used to infer a probabilistic model on the change in state. This is used to create an estimate for the target position at the next time step. Long-term prediction takes place by recursively predicting the target position to the next time step.

Motivated by [17] and [8], the authors in [6] extended this work to estimate the probability density function of the future trajectory of a vehicle. Based on a trajectory database, they introduced the quaternion-based rotationally invariant longest common subsequence (QRLCS) metric as a similarity measure between trajectories and use this in a particle filter framework to track and assess the motion hypotheses within the database. This framework is also used in [9] for pedestrian action classification (walking vs. stopping). The cluster with the highest accumulated particle weight is used as the final predicted state. In a later work, [7] used the trajectory database to extract motion patterns for the history and future in every time step. After a Chebyshev decomposition is applied, the coefficients of this approximation define the feature space and with the help of the Unsupervised Kernel Regression (UKR), the feature vectors are projected into a low dimensional manifold. Under the assumption that every common motion pattern is embedded in this low dimensional sub-space, a particle filter is used to track the history inside the manifold and create a prediction out of the most probable position within the manifold.

In this contribution, the idea of trajectory representation presented in [7] is used and extended to a trajectory prediction framework. This framework has the ability not only to predict a single future trajectory but also to create a whole probability distribution over the space of future trajectories. The vehicle's history and according future trajectories are approximated with Chebyshev polynomials and the coefficients are used to get a uniform feature representation. The probabilistic modeling of the trajectory is described in Section II. Because this modeling requires a probability density function over the input feature space, a density approximation function has to be inferred during the training procedure of the algorithm. Two closely related probability density approximations which are suitable for the trajectory prediction are described in Section III. After the training stage, a prediction can be made by calculating statistical parameters of the conditioned density function for the future trajectory. The proposed algorithm is evaluated for both density functions in Section IV and potential improvements are indicated.

II. PROBABILISTIC TRAJECTORY PREDICTION

A. Trajectory Representation

The fundamental input data of the proposed framework are short trajectory pieces, also called sub-trajectories, snippets or trajectons [14]. In this contribution the term snippet is used due to its intuitive description of the input data.

A trajectory T consists of N pairs of xy-positions in the 2D plane together with associated timestamps t_i

$$T = ((x_0, y_0), t_0), \ldots, ((x_{N-1}, y_{N-1}), t_{N-1}) \tag{1}$$

where $t_i < t_{i+1}$ for $i = 0, \ldots, N-1$. The trajectory T can be bounded by the number of elements N, its length L in the xy-plane or by the timespan $\Delta t = t_{N-1} - t_0$. In other words, a trajectory can either be restricted by a fixed number of samples, a fixed length in meters or by the duration an object needs to create this trajectory. An advantage of the trajectory representation is that it does not depend on a specific sensor type, moreover the trajectory (or parts of it) may originate from different types of sensors [8].

An equivalent representation can be achieved if the trajectory is described by its yaw angle $\varphi(t_i)$ and velocity $v(t_i)$ instead of $x(t_i)$ and $y(t_i)$. Applied to the vehicle motion, this representation has the advantage that a rotation of the trajectory can easily be performed by subtracting the rotation angle of the yaw angle.

B. Trajectory Approximation

As noted before, trajectories can be restricted by several constraints, e.g., by the number of elements, the distance or a time constraint. In both latter cases, the number of elements N can vary because it is not guaranteed that any of the measured values are equidistant in time. To get a uniform representation of the trajectory, a Chebyshev decomposition on the components of the trajectory (e.g., the xy-positions or the equivalent φv-representation) is applied. The coefficients of this polynomial approximation are used as input features of the probabilistic model described in the next section.

The Chebyshev polynomial T_n of degree n is defined by

$$T_n(x) = \cos(n \arccos(x)) \tag{2}$$

which looks trigonometric but can be shown to be polynomials. See [16] for a detailed description. The first two polynomials are defined by

$$T_0(x) = 1 \tag{3}$$

$$T_1(x) = x \tag{4}$$

and with the recursive formula

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x), \quad n \geq 1 \tag{5}$$

one can easily calculate the polynomials of higher order. To approximate an arbitrary function $f(x)$ in the interval $[-1, 1]$, the Chebyshev coefficients are defined by

$$c_n = \frac{2}{N} \sum_{k=0}^{N-1} f(x_k) T_n(x_k) \tag{6}$$

where x_k are the N zeros of $T_N(x)$. The reconstruction formula is defined as

$$f(x) \approx \sum_{n=0}^{m-1} c_n T_n(x) - \frac{1}{2} c_0 \tag{7}$$

where $m \leq N$ can be used to control the approximation quality. For a detailed description on the Chebyshev approximation and its advantages compared to other approximations see [16].

Since in this paper the φv-representation is used to describe a trajectory, both components are transformed to the interval $[-1, 1]$ and the Chebyshev decomposition is applied. This results in two m-dimensional vectors of approximation coefficients, one for the velocity $\mathbf{c}_v$ and another for the yaw angle $\mathbf{c}_\varphi$. A concatenation form the final feature vector $\mathbf{x} = [\mathbf{c}_v, \mathbf{c}_\varphi] \in \mathbb{R}^{2m}$. The number of coefficients is essential for the approximation quality. Experiments showed that $m = 5$ coefficients are enough to obtain a good approximation of the trajectory due to the simple course of the components functions.

C. Probabilistic Trajectory Prediction

As already mentioned, prediction is done by calculating the statistical properties, e.g., the mean and covariance, of the conditional distribution $p(\mathbf{x}_f|\mathbf{x}_h)$, where $\mathbf{x}_f$ denotes the approximated future and $\mathbf{x}_h$ the history trajectory respectively. Remember that a feature vector $\mathbf{x}$ consists of the concatenation of the two coefficients vectors of the Chebyshev approximation. In order to get an applicable uncertainty information of the trajectory prediction, we need to transform these uncertainties to the 2D plane of the vehicle's coordinate system. In the following, the transformation of the velocity is described as an example, the transformation of the yaw angle is straightforward.

Let's assume the predicted approximation coefficients $\mathbf{c}_{v,f}$ of the velocity are Gaussian distributed with

$$p(\mathbf{c}_{v,f}) = \mathcal{N}\left(\mathbf{c}_{v,f}|m(\mathbf{c}_{v,h}), cov(\mathbf{c}_{v,h})\right) \tag{8}$$

where $m(\mathbf{c}_{v,h})$ and $cov(\mathbf{c}_{v,h})$ are functions of the mean and covariance of the coefficients describing the history. Assuming this, the Chebyshev approximation (7) is a linear combination of Gaussian distributed variables and therefore is Gaussian as well. Rewriting the Chebyshev approximation as

$$v(t) = c_{v,0}T_0(t) + ... + c_{v,m-1}T_{m-1}(t) - \frac{1}{2}c_{v,0} \tag{9}$$

$$= \mathbf{c}_{v,f}^T \mathbf{T}(t) \tag{10}$$

which can be written for discrete time values as

$$\mathbf{v} = \mathbf{T}\mathbf{c}_{v,f} \tag{11}$$

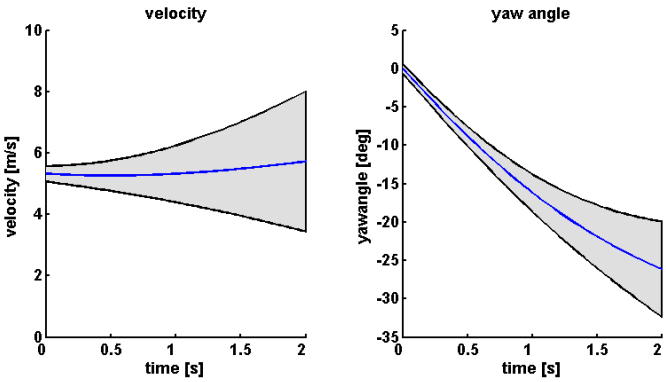
where now $\mathbf{T} = \mathbf{T}_{nk} = T_n(t_k)$ is a matrix of the Chebyshev polynomials evaluated at arbitrary timestamps t_k with $-1 \leq t_k \leq 1$. Using this, the expectation and the covariance of the velocity can be evaluated for every discrete timestamp t_k

$$E\left[\mathbf{v}\right] = \mathbf{T}E\left[\mathbf{c}_{v,f}\right] \tag{12}$$

$$Cov\left[\mathbf{v}\right] = E\left[\mathbf{v}\mathbf{v}^T\right] = \mathbf{T}E\left[\mathbf{c}_{v,f}\mathbf{c}_{v,f}^T\right]\mathbf{T}^T \tag{13}$$

$$= \mathbf{T}Cov(\mathbf{c}_{v,f})\mathbf{T}^T. \tag{14}$$

Therefore the expectation and covariance are functions of t. Figures 2a and 2b show an example of the predicted mean and covariance of the velocity and yaw angle for a prediction horizon of 2 s.



(a) velocity (b) yaw angle

Fig. 2: Mean and covariance function of the predicted velocity and yaw angle for a prediction horizon of 2 seconds.

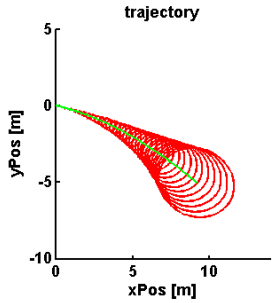


Fig. 3: The final prediction together with the associated uncertainties of the xy-position depicted as covariance ellipses.

With the predicted mean and covariance function of the velocity and yaw angle, the uncertainty in the xy-plane can be calculated again for arbitrary points within the prediction horizon using an unscented transform. This performs a nonlinear transformation of the statistical properties from the φv-domain to the xy-coordinate system. The result is a prediction which consists of the mean trajectory together with the uncertainty information of the evaluated values. Figure 3 shows an example of the final estimation with a prediction horizon of 2 seconds.

III. MIXTURE DISTRIBUTIONS FOR TRAJECTORY PREDICTION

So far, the idea of the probabilistic trajectory prediction was discussed without defining a particular probability density function for $p(\mathbf{x}_f, \mathbf{x}_h)$. Due to their excellent approximation properties, mixture models are often used for density approximation. In this paper, two related probability distributions are examined, the Gaussian Mixture Model (GMM) and its Bayesian extension, the Variational Gaussian Mixture Model (VGMM). While the first model makes a point estimate for the model parameters, the latter introduces prior distributions over the parameters and therefore derives a fully Bayesian treatment.

A. Gaussian Mixture Model

The Gaussian Mixture Model (GMM) is a parametric probability density function which consists of a weighted linear combination of Gaussian component densities. Using an adequate number of mixture components, it is possible to approximate almost every continuous probability density function. The Gaussian mixture density is defined as

$$p(\mathbf{x}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{15}$$

where $\mathbf{x}$ is a d-dimensional random variable, $\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ is a multivariate normal distribution with mean $\boldsymbol{\mu}_k$ and covariance matrix $\boldsymbol{\Sigma}_k$ and π_k are the so-called mixing coefficients for the k components of the distribution $p(\mathbf{x})$ which have to satisfy $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^{K} \pi_k = 1$ to form a convex combination of the mixture components [1]. A powerful algorithm to derive the model parameters from training data is the iterative two-step Expectation-Maximization (EM) algorithm which finds the maximum likelihood solution in a very graceful way [2]. The E-step computes the expectation of the log-likelihood evaluated using the current parameter estimates followed by the M-step, which estimates parameters that maximize the expected log-likelihood found by the E-step.

Applied to the application of trajectory prediction, we infer a joint Gaussian mixture distribution

$$p(\mathbf{x}_f, \mathbf{x}_h) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_h, \mathbf{x}_f|\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{16}$$

over the approximated history and future trajectory snippets $\mathbf{x}_h$ and $\mathbf{x}_f$. Prediction is performed by calculating the conditional mixture density which is given by

$$\begin{aligned} p(\mathbf{x}_f|\mathbf{x}_h) &= \frac{p(\mathbf{x}_h, \mathbf{x}_f)}{\int p(\mathbf{x}_h, \mathbf{x}_f) d\mathbf{x}_f} \\ &= \sum_{k=1}^{K} \tilde{\pi}_k \mathcal{N}(\mathbf{x}_f|\mathbf{x}_h, \tilde{\boldsymbol{\mu}}_k, \tilde{\boldsymbol{\Sigma}}_k) \end{aligned} \tag{17}$$

and therefore still is a Gaussian Mixture Model. The parameters of this model are given by

$$\tilde{\pi}_k = \frac{\pi_k p(\mathbf{x}_h|\boldsymbol{\mu}_{k,x_h}, \boldsymbol{\Sigma}_{k,x_h x_h})}{\sum_{j=1}^{K} \pi_j p(\mathbf{x}_h|\boldsymbol{\mu}_{j,x_h}, \boldsymbol{\Sigma}_{j,x_h x_h})} \tag{18}$$

$$\tilde{\boldsymbol{\mu}}_k = \boldsymbol{\mu}_{k,x_f} + \boldsymbol{\Sigma}_{k,x_f,x_h} \boldsymbol{\Sigma}_{k,x_h x_h}^{-1} (\mathbf{x}_h - \boldsymbol{\mu}_{k,x_h}) \tag{19}$$

$$\tilde{\boldsymbol{\Sigma}}_k = \boldsymbol{\Sigma}_{k,x_f x_f} - \boldsymbol{\Sigma}_{k,x_f x_h} \boldsymbol{\Sigma}_{k,x_h x_h}^{-1} \boldsymbol{\Sigma}_{k,x_h x_f} \tag{20}$$

where

$$\boldsymbol{\mu}_k = \begin{bmatrix} \boldsymbol{\mu}_{k,x_h} \\ \boldsymbol{\mu}_{k,x_f} \end{bmatrix} \tag{21}$$

$$\boldsymbol{\Sigma}_k = \begin{bmatrix} \boldsymbol{\Sigma}_{k,x_h x_h} & \boldsymbol{\Sigma}_{k,x_h x_f} \\ \boldsymbol{\Sigma}_{k,x_f x_h} & \boldsymbol{\Sigma}_{k,x_f x_f} \end{bmatrix} \tag{22}$$

is the partitioning of the means and covariance matrices of the mixture model.

The derived distribution (17) defines a full conditional probability density function of the future trajectories of which expectations (e.g., mean and covariance) can be evaluated. The mean and covariance of a Gaussian Mixture Model is given by

$$\boldsymbol{\mu} = \sum_{k=1}^{K} \pi_k \boldsymbol{\mu}_k \tag{23}$$

$$\boldsymbol{\Sigma} = \sum_{k=1}^{K} \pi_k \left(\boldsymbol{\Sigma}_k + (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T \right) \tag{24}$$

and the probabilistic trajectory prediction derived in the previous section can be applied.

B. Variational Gaussian Mixture Model

The Variational Gaussian Mixture Model (VGMM) is the Bayesian treatment of the standard Gaussian Mixture Model described in the previous section. Bayesian modeling shows good generalization capabilities and overcomes the disadvantages of the maximum likelihood solution, e.g, overfitting, sensibility to outliers and singularities in the covariance matrix. Instead of point estimates, all parameters are given conjugate prior distributions. For the mixing cofficients π_k a Dirichlet distribution

$$p(\boldsymbol{\pi}) = Dir(\boldsymbol{\pi}|\boldsymbol{\alpha}_0) \tag{25}$$

is used which is the conjugate prior of the multinomial distribution. The conjugate prior distribution for the mean $\boldsymbol{\mu}_k$ and precision $\boldsymbol{\Lambda}_k$ is given by

$$p(\boldsymbol{\mu}, \boldsymbol{\Lambda}) = \mathcal{N}\left(\boldsymbol{\mu}|\mathbf{m}_0, (\beta_0 \boldsymbol{\Lambda})^{-1}\right) \mathcal{W}(\boldsymbol{\Lambda}|\mathbf{W}_0, \nu_0) \tag{26}$$

which is an independent Gaussian-Wishart distribution where it is assumed that the joint distribution $p(\boldsymbol{\mu}, \boldsymbol{\Lambda})$ can be factorized [1]. The parameters $\mathbf{m}_0$ and $\mathbf{W}_0$ are the priors for the mean and precision, β_0 is a scaling parameter and ν_0 is the prior for the number of degrees of freedom of the Wishart distribution with $\nu > d - 1$.

Because an analytical solution is intractable, approximate inference is used to obtain the parameters of the posterior distribution. The variational pendant to the standard Expectation-Maximization algorithm, the so called Variational Bayesian Expectation-Maximization (VB EM) algorithm, can be used to infer the Variational Mixture Distribution. For a detailed description of the variational inference framework and its application to Gaussian Mixture Models see [1][15].

The predictive density for a new value $\hat{\mathbf{x}}$ of the observed variable is approximately a mixture of Student's t-distributions [1]

$$p(\hat{\mathbf{x}}) \simeq \frac{1}{\boldsymbol{\alpha}} \sum_{k=1}^{K} \alpha_k St\left(\hat{\mathbf{x}}|\mathbf{m}_k, \mathbf{L}_k, \nu_k + 1 - d\right) \tag{27}$$

where $\mathbf{m}_k$ is the mean and $\mathbf{L}_k$ is the precision of the k-th component with

$$\mathbf{L}_k = \frac{(\nu_k + 1 - d)\beta_k}{1 + \beta_k} \mathbf{W}_k. \tag{28}$$

Applying the trajectory prediction framework and approximating the joint density $p(\mathbf{x}_f, \mathbf{x}_h)$ with the Variational Gaussian Mixture Model, the predictive density (27) is used to calculate $p(\mathbf{x}_f|\mathbf{x}_h)$, where again $\mathbf{x}_h, \mathbf{x}_f \in \mathbb{R}^d$ corresponds to the coefficients of the approximated history and future trajectory. This results in a conditional mixture of Student's t-distributions

$$p(\mathbf{x}_f|\mathbf{x}_h) \simeq \frac{1}{\hat{\boldsymbol{\alpha}}} \sum_{k=1}^{K} \hat{\alpha}_k St\left(\mathbf{x}_f|\mathbf{x}_h, \hat{\mathbf{m}}_k, \hat{\mathbf{L}}_k, \hat{\nu}_k + d\right). \tag{29}$$

The parameters of the conditional density are

$$\hat{\nu}_k = \nu_k + 1 - d \tag{30}$$

$$\hat{\alpha}_k = \frac{\alpha_k \, St(\mathbf{x}_h|\mathbf{m}_{k,x_h}, \mathbf{L}_{k,x_h}, \hat{\nu}_k)}{\sum_{j=1}^{K} \alpha_j \, St(\mathbf{x}_h|\mathbf{m}_{j,x_h}, \mathbf{L}_{j,x_h}, \hat{\nu}_k)} \tag{31}$$

$$\hat{\mathbf{m}}_k = \mathbf{m}_{k,x_f} + \boldsymbol{\Sigma}_{k,x_f x_h} \boldsymbol{\Sigma}^{-1}_{k,x_h x_h} (\mathbf{x}_h - \mathbf{m}_{k,x_h}) \tag{32}$$

$$\hat{\mathbf{L}}^{-1}_k = \frac{\hat{\nu}_k}{\hat{\nu}_k + d - 2} \left(1 + \Delta_k^T \frac{\boldsymbol{\Sigma}^{-1}_{k,x_h x_h}}{\hat{\nu}_k} \Delta_k\right) \boldsymbol{\Sigma}^*_k \tag{33}$$

with

$$\Delta_k = (\mathbf{x}_h - \mathbf{m}_{k,x_h}) \tag{34}$$

$$\boldsymbol{\Sigma}^*_k = \boldsymbol{\Sigma}_{k,x_f x_f} - \boldsymbol{\Sigma}_{k,x_f x_h} \boldsymbol{\Sigma}^{-1}_{k,x_h x_h} \boldsymbol{\Sigma}_{k,x_h x_f} \tag{35}$$

where $\boldsymbol{\Sigma}_k = \frac{\hat{\nu}_k + d - 2}{\hat{\nu}_k + d} \mathbf{L}^{-1}_k$ is the positive definite correlation matrix and the partitioning is analogous as for the conditioned Gaussian mixture distribution [10] [11].

Another possibility is to use the Variational GMM only for the training procedure to benefit from the advantages of the approximate inference. Afterwards, one can calculate the expectations of the parameter distributions and use them as point estimates to approximate a standard Gaussian Mixture Model.

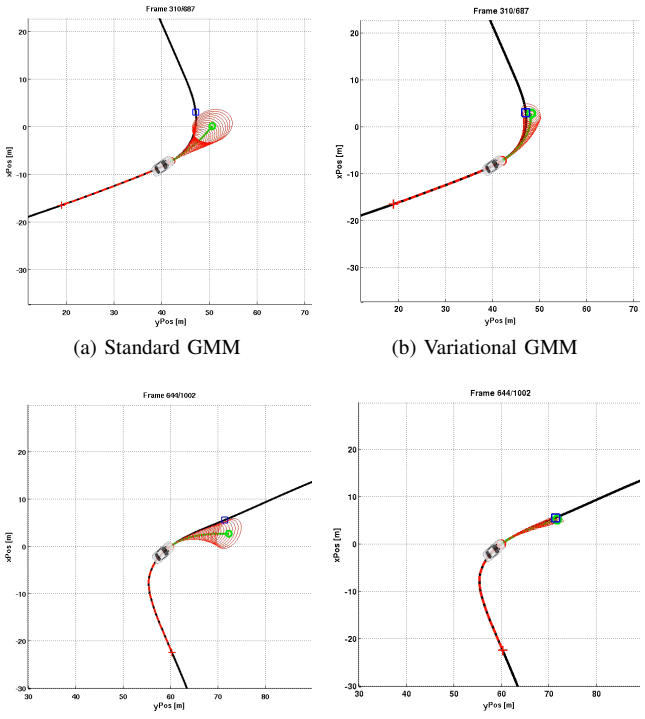
IV. RESULTS

For the evaluation of the proposed algorithms, the dataset of [7] is used which consists of 69 real world trajectories, recorded at three different intersections with varying maneuvers (passing straight, left and right turns). This set has a total length of roughly 24 km and consists of Differential GPS (DGPS) data from which the velocity $v(t_i)$ and yaw rate $\varphi(t_i)$ values were extracted at discrete time steps t_i. As trajectory parameters, a history length of the past 25 meters and a prediction horizon of 2 seconds is used for the generation of the motion patterns. This configuration results in about 120.000 pairs of trajectory snippets. The set is divided into a training set and a test set with 48 and 21 trajectories respectively. The trajectories of the test set originate from an intersection not present in the training data.

For the accuracy of the Chebyshev approximation we restrict ourselves to an approximation of fourth-order which results in a $d = 10$ dimensional feature space for the history $\mathbf{x}_h$ as well as for the future trajectory snippets $\mathbf{x}_f$.

Training for the Gaussian Mixture Model and its variational expansion is done with $K = 75$ components and 500 Iterations with the already mentioned Expectation Maximization algorithms. The parameters of the prior distribution for the Variational Mixture Model are $\beta_0 = 1$, $\alpha_0 = 1e^{-3}$ and $\mathbf{W}_0 = 20\,\mathbf{I}$ where $\mathbf{I}$ is the $2d$-dimensional identity matrix.

Figure 4 illustrates two turning maneuvers at an intersection for both algorithms. The Gaussian Mixture Model tend to predict the upcoming turn very late (see Figure 4a) and overestimates it at the end of the maneuver (see Figure 4c). This behavior is comparable to that of a kinematic model with the assumption of constant yaw angle or constant yaw rate. In contrast, the Variational Gaussian Mixture Model does not show this sluggish behavior. As can be seen in Figure 4b, the algorithm predicts the trajectory very accurate at the beginning of the maneuver and does not overestimate it at the end (see Figure 4d). Also, the predicted covariance describes the uncertainty much better than the one of the standard Gaussian Mixture Model.



(a) Standard GMM (b) Variational GMM

(c) Standard GMM (d) Variational GMM

Fig. 4: Two examples of the trajectory prediction. On the left are two different scenes predicted by the Gaussian Mixture Model and on the right are the predictions at the same timestamp using the Variational Gaussian Mixture Model.

Figure 5 shows the errors for the 21 test trajectories for both algorithms. The error is defined as the Euclidean distance between the true position and the predicted position at the end of the prediction horizon of 2 seconds. The standard GMM has a mean error of roughly 1 meter whereas the prediction error of the Variational GMM in most cases is significantly lower. The high quantiles can be explained by the fact that in most cases, a turn is not predicted until the car started it and so there is temporarily a higher error.

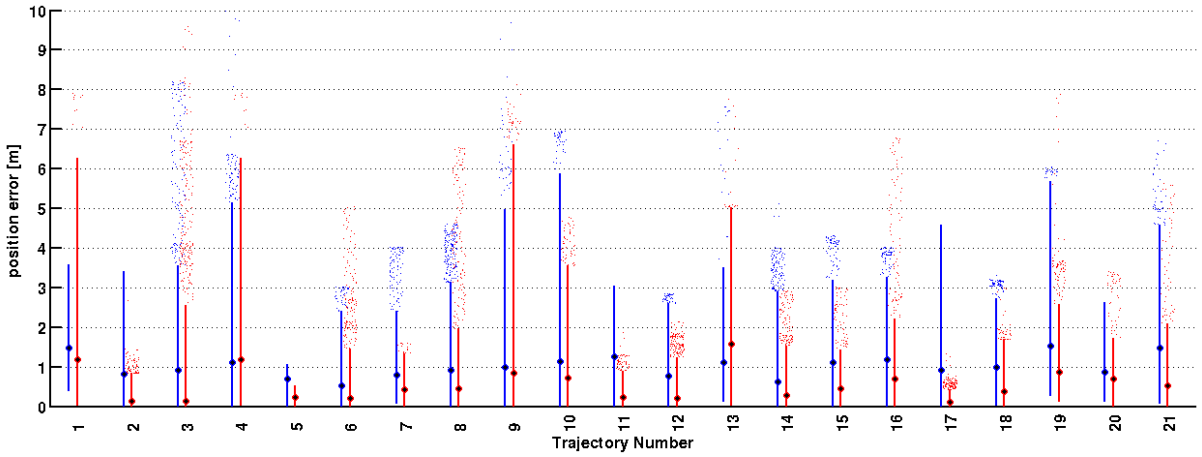


Fig. 5: Prediction error for the 21 trajectories of the test set of the standard Gaussian Mixture Model (left/blue) and the Variational Mixture Model (right/red). The error measurement is defined as the position error at a prediction horizon of 2 seconds. The large dots correspond to the mean error, the bars correspond to the 25 and 75 percent quantiles and the dots outside correspond to outliers.

This effect may be reduced by taking the modes of the distributions instead of the mean.

V. CONCLUSION

In this contribution, a probabilistic trajectory prediction based on two types of mixture models is proposed. Instead of learning a regression function, previously observed motion patterns are used to infer a probability distribution as motion model. The probabilistic modeling makes it possible to transform statistical parameters to the xy-domain of the vehicle's coordinate system, which is described in Section II. After that, two closely related probability density functions suitable for trajectory prediction are described and their conditional distributions are pointed out. The algorithm is evaluated on 21 test trajectories of typical turning maneuvers. It turns out that both density functions are able to predict the vehicle's trajectory with a prediction horizon of 2 seconds. Due to the good generalization capabilities of Bayesian models, the Variational Mixture Model outperforms the standard Gaussian Mixture Model and leads to accurate prediction results.

VI. ACKNOWLEDGMENTS

This work was supported by Daimler AG, Department Environment Perception, Ulm, Germany.

REFERENCES

[1] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006.

[2] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

[3] A. Eidehall and L. Petersson. Statistical threat assessment for general road scenes using monte carlo sampling. *Intelligent Transportation Systems, IEEE Transactions on*, 9(1):137–147, 2008.

[4] David Ellis, Eric Sommerlade, and Ian Reid. Modelling pedestrian trajectory patterns with gaussian processes. 2009.

[5] Sven Hellbach, Julian P Eggert, Edgar Kärner, and Horst-Michael Gross. Time series analysis for long term prediction of human movement trajectories. In Mario Köppen, Nikola Kasabov, and GeorgeEditors Coghill, editors, *Advances in NeuroInformation Processing*, volume 5507, pages 567–574. Springer, 2009.

[6] Christoph Hermes, Alexander Barth, Christian Wöhler, and Franz Kummert. Object motion analysis and prediction in stereo image sequences. In *8. Oldenburger 3D-Tage*, Oldenburg, 28/01/2009 2009.

[7] Christoph Hermes, Jürgen Wiest, Christian Wöhler, Ulrich Kreßel, and Franz Kummert. Manifold-based motion prediction. In *Proc. 6. Dortmunder Auto-Tag*, 2011.

[8] Christoph Hermes, Christian Wöhler, Konrad Schenk, and Franz Kummert. Long-term vehicle motion prediction. In *Proceedings of the IEEE Intelligent Vehicles Symposium (IV)*, pages 652–657, 2009.

[9] Christoph Keller, Christoph Hermes, and Dariu Gavrila. Will the pedestrian cross? probabilistic path prediction based on learned motion features. In *Pattern Recognition*, volume 6835, pages 386–395. Springer Berlin / Heidelberg, 2011.

[10] B. M. Golam Kibria and Anwar H Joarder. A short review of multivariate t-distribution. *Journal of Statistical Research*, 40(1):59–72, 2006.

[11] Samuel Kotz and Saralees Nadarajah. *Multivariate T-Distributions and Their Applications*. Cambridge University Press, 2004.

[12] Frédéric Large, Dizan Alejandro Vasquez Govea, Thierry Fraichard, and Christian Laugier. Avoiding cars and pedestrians using v-obstacles and motion prediction. In *Proc. of the IEEE Intelligent Vehicle Symp.*, Pisa (IT), 6 2004. Poster session.

[13] Xi Li, Weiming Hu, and Wei Hu. A coarse-to-fine strategy for vehicle motion trajectory clustering. In *ICPR (1)*, pages 591–594, 2006.

[14] Pyry Matikainen, Martial Hebert, and Rahul Sukthankar. Trajectons: Action recognition through the motion analysis of tracked features. 2009.

[15] N. Nasios and A.G. Bors. Variational learning for gaussian mixture models. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 36(4):849 –862, aug. 2006.

[16] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, New York, NY, USA, 3 edition, 2007.

[17] Hedvig Sidenbladh, Michael J. Black, and Leonid Sigal. Implicit probabilistic models of human motion for synthesis and tracking hedvig sidenblen. In *In European Conference on Computer Vision*, pages 784–800, 2002.