2014 IEEE Intelligent Vehicles Symposium (IV)
June 8-11, 2014. Dearborn, Michigan, USA

Analysis of pedestrian dynamics from a vehicle perspective

Julian F. P. Kooij[†,1,2], Nicolas Schneider[‡,1,2] and Dariu M. Gavrila[†,1,2]

***Abstract*— Accurate motion models are key to many tasks in the intelligent vehicle domain, but simple Linear Dynamics (e.g. Kalman filtering) do not exploit the spatio-temporal context of motion. We present a method to learn Switching Linear Dynamics of object tracks observed from within a driving vehicle. Each switching state captures object dynamics as a mean motion with variance, but also has an additional spatial distribution on where the dynamic is seen relative to the vehicle. Thus, both an object's previous movements and current location will make certain dynamics more probable for subsequent time steps. To train the model, we use Bayesian inference to sample parameters from the posterior, and jointly learn the required number of dynamics. Unlike Maximum Likelihood learning, inference is robust against overfitting and poor initialization.**

We demonstrate our approach on an ego-motion compensated track dataset of pedestrians, and illustrate how the switching dynamics can make more accurate path predictions than a mixture of linear dynamics for crossing pedestrians.

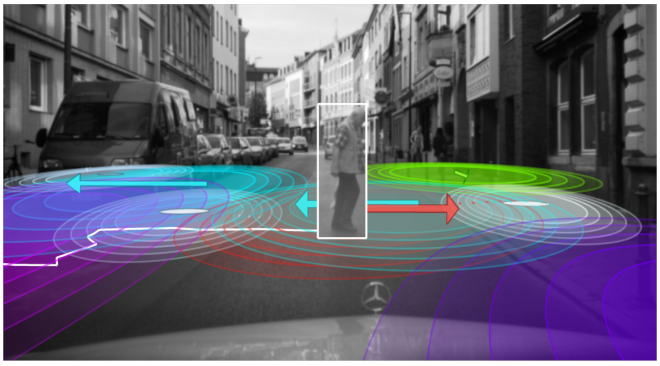
I. INTRODUCTION

Research in the intelligent vehicles domain has over the years resulted in improved driver assistance systems by successfully applying pattern recognition (e.g. computer vision) and machine learning techniques on complex real-world data. Our company Daimler, for example, has introduced an innovative stereo-based pedestrian safety system in its 2013-2014 Mercedes-Benz S-, E-, and C-Class models, which incorporates fully automatic emergency braking.

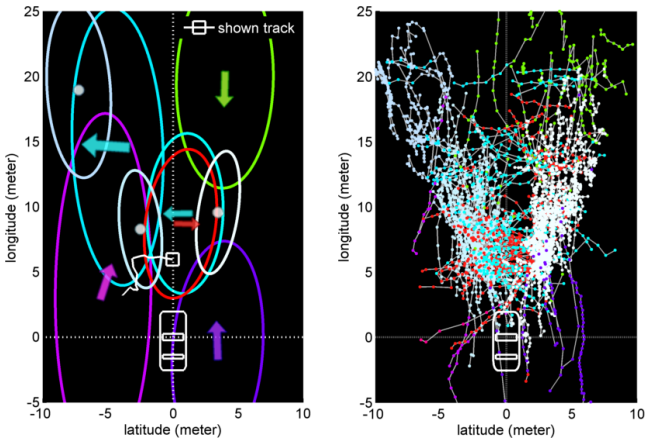
Accurate object motion models are key for sophisticated driver assistance systems, that perform tracking of multiple objects and path prediction for situation analysis. Motion dynamics are commonly modeled using a Linear Dynamical System (LDS) with Gaussian noise, for which the well known Kalman Filter can be used to maintain a optimal probabilistic estimate of the true motion state. The underlying assumption that all target motions can be described by the same dynamics can be too simplistic though. One can improve this model by assuming that targets exhibit one out of a set of possible dynamics, i.e. a Mixture of LDS [1]. However, sometimes targets exhibit various dynamics even within the same track. Especially pedestrians can quickly change from, for example, standing to walking when crossing a road [2]. Their behaviors also exhibit spatial structure, e.g. one typically walks parallel to the road on the sidewalk and does not stand still in front of a vehicle.

In this paper, we model object dynamics with a Switching LDS, extended with spatial distributions on where different types of motion occur near the vehicle. We train the model on real-world pedestrian data recorded from a moving vehicle. Figure 1 illustrates our model and used track data[1].

Unsupervised learning raises questions on determining the number of dynamics needed, and on avoiding overfitting in dynamics with few samples. EM-like schemes also rely on good initial estimates (e.g. [1] required an external clustering method) and Maximum Likelihood learning is prone to overfitting. We therefore use a Bayesian inference scheme to tackle these issues in a more principled way, and learn the number of dynamics from the data without overfitting.



(a) Pedestrian dynamics (mean directions and locations) learned from track data, such as the one shown here of a crossing person, as seen from a vehicle.



(b) Top-down view of the situation shown in (a) (vehicle at origin).

(c) All pedestrian tracks; positions are color-coded with used dynamics.

Fig. 1. (a), (b) Illustration of learned pedestrian dynamics. For each dynamic the average motion direction is shown as an arrow (or as a spot where the average direction is near zero), as well as the spatial region on the ground plane. The process noise of the dynamics (motion uncertainty) is not shown here. (c) Pedestrian tracks from which dynamics were learned.

[†] J.F.P.Kooij@uva.nl, [‡] Nicolas.Schneider@daimler.com,
[†] D.M.Gavrila@daimler.com
[1] Environment Perception, Daimler R&D, Ulm, Germany
[2] Intelligent Systems Laboratory, Univ. of Amsterdam, The Netherlands

[1]A video animation is available by following the links from http://www.gavrila.net

II. RELATED WORK

Motion models are essential to tasks as tracking [3], anomaly detection [4], [5] and path prediction [2], [6], found in a variety of application domains, from intelligent vehicles [1], [2], [6], [7], [8], [9] to fixed-viewpoint surveillance [4], [5], [10], [11]. In intelligent vehicles for instance, trackers commonly build tracks from detections (e.g. for pedestrians [7], [12], [13]), and use motion models to reduce false positives and resolve data association [7], [9].

When both the motion dynamics and observations can be expressed as a linear transformation of a latent state with added Gaussian noise, then the resulting model is a Linear Dynamical System (LDS). The Kalman Filter (KF) [3] can then be used for efficient forward filtering, maintaining uncertainty over the state by a single normal distribution. For batch processing, the Kalman Smoother extends the KF by a backward pass to incorporate information from future time steps in state estimation. The Extended KF (EKF) [6] or Unscented KF (UKF) [9], [14] are used in certain cases where the observations are not linear transformations of the underlying state space. But the LDS is not an optimal model for track data that contains various dynamics. A Mixture of LDS is an extension where a track uses one out of multiple possible dynamics [1], see Fig. 2(a). But in many cases a target can change dynamics within a single track [5], [15], such as when a person starts walking after standing still.

The Switching Linear Dynamical System (SLDS) [16], [17] allows for different dynamics at each time step, conditioning the linear dynamics on an additional hidden Markov chain of switching states (see Fig. 2(b)) which have their own prior and transition probability. Unfortunately, exact inference in a SLDS is intractable since the latent state depends on the full history of switching states [16]. Expert knowledge can be exploited to some extent to define human motion classes [11], [15], but not all model parameters are known or optimal for a target application. Therefore, we must resort to approximate inference for both training and filtering. For on-line filtering, the Interacting Multiple Models (IMM) algorithm [3], [2], [6] filters a track simultaneously with the various dynamics, and mixes their predictions. Alternatively, [16] presents a collapsed Gibbs sampler [18] for the switching states by integrating over the latent positions.

Gibbs sampling can also be used for learning, though other learning schemes using Viterbi or Variational inference are possible too [19]. Fox et al. [17] presented a Gibbs sampler for a SLDS in a Dirichlet Process [20], estimating both the parameters and number of dynamics. In [5] we took a similar approach to learn person dynamics in a surveillance scenario. Additionally, we learned spatial distributions of where different states occur, while hierarchically clustering tracks that exhibit similar state transitions too.

Our contributions in this paper are two-fold. First, instead of one or multiple Kalman filters to model object dynamics, we use a SLDS and extend it with additional spatial distributions to capture where specific dynamics occur. This does not exclude the possibility of the same dynamic (e.g. moving left) at spatially distant regions, since this can be represented by separate switching states with similar process noise but different spatial distributions. We describe how to use Bayesian inference to learn both the number of switching dynamics and their parameters. By sampling from the posterior, this approach is more robust to random initialization and overfitting than EM-based learning. Second, our approach is a novel way in the intelligent vehicle domain to analyze real-world track data captured from a vehicle. We show that the method discovers meaningful and spatially localized dynamics for pedestrians. As an example, we show how this structure could yield more accurate path prediction than a baseline using a Mixture of LDS.

III. APPROACH

To train a particular model $p(X|\Theta)$, one needs to search the parameter space Θ and evaluate how probable a particular solution is for the (training) data X. A common approach to learning is to find the Maximum Likelihood (ML) solution,

$$\theta^{\star}_{ML} = \operatorname*{argmax}_{\theta} p(X|\Theta = \theta). \quad (1)$$

The predictive density for some new data point x' is then simply $p(x'|\theta^{\star}_{ML})$. However, this solution is prone to overfitting, especially when there are many parameters, or the training data is sparse. A more principled way to counter these effects is to use an appropriate prior distribution $p(\theta)$ on the parameters we wish to learn, and use Bayes' rule to find the posterior for a Maximum A-Posteriori (MAP) solution,

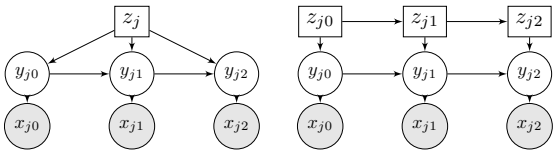
$$\theta^{\star}_{MAP} = \operatorname*{argmax}_{\theta} p(\Theta = \theta|X) = \operatorname*{argmax}_{\theta} p(X|\theta)p(\theta), \quad (2)$$

Even better, the posterior could be used for full Bayesian inference by integrating over all possible parameters and account for uncertainty, $p(x'|X) = \int p(x'|\theta)p(\theta|X)d\theta$.

Generally, this integral cannot be solved analytically and exact Bayesian inference is unfortunately intractable. Still, there are various methods for approximation available [18]. Markov Chain Monte Carlo (MCMC) techniques do this by sequentially generating samples $\theta^{(i)}$ of the posterior, using some stochastic process f such that $\theta^{(i)} \sim f(\theta^{(i-1)})$.

Gibbs sampling is a form of MCMC. It updates one or a few hidden variables at a time, sampling them from their marginal distribution while keeping all other hidden variables fixed to their current value. We thus reduce inference in a complex model to iteratively sampling the hidden variables from a few simpler distributions. In comparison to EM [1], MCMC is less sensitive to the initial state since it can move away from local maxima. On sparse data, samples reflect the prior instead of overfitting. As more data is obtained, sampling might become computationally more expensive, but the posterior also contains less uncertainty. Since there is less variance in the samples, fewer iterations can be used instead.

In this paper, we use Gibbs sampling to explore the parameter space of our model. The sampler could be used as an any-time algorithm, by running it in the background and keeping track of the best MAP parameters found at any time, or to approximate full Bayesian inference.



(a) Track j in Mixture of LDSs (b) Track j in Switching LDS

Fig. 2. Graphical model of a Mixture of Linear Dynamical Systems, and a Switching Linear Dynamical System (SLDS). Here discrete nodes are rectangular, continuous ones are round, and observed nodes are shaded. Each model is depicted for three time slices. When modeling a set of tracks as a Mixture of LDSs, then each track has a single dynamic or class label which indicates which of the LDS parameters it uses. The SLDS on the other hand has a dynamic state not per track but per time instance.

A. Model

Let the training data consist of a set of J tracks, each being a sequence observed positions of the target. We will index tracks by suffix j, and let $X_j = \{x_{j1}, \cdots, x_{jT_j}\}$ be the T_j observations. Observed positions are typically noisy measurements of latent true positions, which we represent as the corresponding sequence $Y_j = \{y_{j1}, \cdots, y_{jT_j}\}$. The SLDS additionally has a latent Markov chain of switching states for each time step, $Z_j = \{z_{j1}, \cdots, z_{jT_j}\}$ that determine the used dynamics, see Figure 2(b). So if there are K dynamics, then it holds that $1 \leq z_{jt} \leq K$.

The process noise $\mathcal{N}(m_k, Q_k)$ is non-zero mean and conditioned on the switching state, resulting in the dynamics

$$y_{jt} = Ay_{jt-1} + q_{jt} \qquad q_{jt} \sim \mathcal{N}(m_{z_{jt}}, Q_{z_{jt}}) \tag{3}$$
$$x_{jt} = Cy_{jt} + r_{jt} \qquad r_{jt} \sim \mathcal{N}(0, R) \tag{4}$$

where q_{jt} is the non-zero mean process noise at time t, and r_{jt} the zero mean observation noise. Matrices A and C are fixed and are determined by the type of states, observations and kinematics used. The temporal transitions in the chain of switching states are parameterized by a $K \times K$ transition matrix $\Pi = [\pi_1; \cdots; \pi_K]$, where each row π_k is a parameter vector of K elements for the Categorical distribution over next states given current state k,

$$p(z_{jt+1} = k' | z_{jt} = k) = \text{Cat}(k' | \pi_k) = \pi_{k(k')}. \tag{5}$$

i.e. the k'-th element of the vector π_k.

We add an additional Normal distribution over the observations to express where a certain motion dynamic is likely to occur. The spatial distribution of switching state k, is parameterized by (μ_k^S, Σ_k^S),

$$x_{jt} \sim \mathcal{N}(\mu_{z_{jt}}^S, \Sigma_{z_{jt}}^S). \tag{6}$$

Now there are two predictive density terms for the observations, i.e. the LDS term (4), and the spatial term (6). We use then the product of both as the final predictive density.

In conclusion, the complete set of model parameters is $\Theta = (R, \forall k \{m_k, Q_k, \pi_k, \mu_k^S, \Sigma_k^S\})$.

B. Conjugate Priors

The priors of the model are conjugate (i.e. which have the convenient property that the posterior distributions have the same parametric form [18]) and shared by all K states,

$$R \sim \mathcal{W}^{-1}(\xi_0^R) \tag{7}$$
$$m_k, Q_k \sim \mathcal{NW}^{-1}(\xi_0^Q) \tag{8}$$
$$\mu_k^S, \Sigma_k^S \sim \mathcal{NW}^{-1}(\xi_0^S) \tag{9}$$
$$\pi_k \sim \text{Stick}(\alpha) \tag{10}$$

with $\xi_0^R = \{\nu_0^R, \Psi_0^R\}$ the parameters of an Inverse-Wishart (IW) distribution, a distribution over covariance matrices, and $\xi_0^Q = \{\mu_0^Q, \kappa_0^Q, \nu_0^Q, \Psi_0^Q\}$ and ξ_0^S the parameters for a Normal-Inverse-Wishart (NIW), a joint distribution over mean and covariance. Intuitively, IW parameters (ν, Ψ) represent ν 'virtual' data points with sample covariance matrix Ψ/ν. For low ν, IW spreads its density over many covariance matrices (i.e. we are uncertain of the true covariance that generated data with that sample covariance) and the distribution peaks on the sample covariance as ν increases (i.e. low uncertainty on the covariance). The NIW extends IW with a Normal over a mean value, see [18] for details.

Finally, α is the concentration parameter of the Stick-breaking distribution. This distribution, from which the multinomial parameters π_k are sampled, is a distribution over vectors of possibly *infinite* elements that sum to one. If the vector of component weights in a mixture model is distributed along a Stick-breaking distribution, it is said to be an infinite or Dirichlet Process mixture model [20], [17]. Note that an infinite mixture model can represent finite mixture models of any size, as these are just cases where all except a few of the infinite weights are zero. Thus instead of fixing the cluster count up front, we can learn how many non-zero weights are needed for the training data. Parameter α expresses a prior preference for a few large or many smaller clusters. The advantage of having a concentration parameter is that the actual number of clusters in the data is typically unknown a-priori, and varies from dataset to dataset.

Instead of implementing Stick-breaking explicitly, which requires additional math and more memory management when clusters are added and removed, it has been shown [20] that we can instead use a Dirichlet distribution with a parameter *vector* α_0 of fixed and finite length K,

$$\pi_k \sim \text{Dir}(\alpha_0), \qquad \text{where } \alpha_0 = [\alpha/K, \cdots, \alpha/K]. \tag{11}$$

This yields the same distribution as the Stick-breaking prior of Eq. (10) in the infinite limit of $K \to \infty$. In practice, we regard K as an upper-limit on the actual number of clusters, which is discovered from the data. As sampling converges, certain elements in the vectors π_k will go to zero, and the actual number of clusters is found by counting the number clusters with non-zero probability.

C. Gibbs sampling procedure

Like in [5], [17], we use Gibbs sampling to search the parameter space by iteratively executing the following steps.

a) Sampling dynamics R, m_k, Q_k, μ_k^S, Σ_k^S: At this stage, we assume that the switching states Z of all tracks are fixed. Due to conjugacy, the posterior distributions of

the model parameters have the same form as their priors. Hence, at the i-th Gibbs iteration they are sampled as,

$$R^{(i)} \sim \mathcal{W}^{-1}(\xi_+^R) \tag{12}$$

$$m_k^{(i)}, Q_k^{(i)} \sim \mathcal{NW}^{-1}(\xi_{+k}^Q) \tag{13}$$

$$\mu_k^{S(i)}, \Sigma_k^{S(i)} \sim \mathcal{NW}^{-1}(\xi_{+k}^S) \tag{14}$$

with the posterior parameters $\xi_+^R, \xi_{+k}^Q, \xi_{+k}^S$ computed from the prior parameters and sufficient statistics of the data. Note that all dynamics share the same priors ξ_0^Q, ξ_0^S, but have different posteriors ξ_{+k}^Q, ξ_{+k}^S after considering the data.

To illustrate, if we would know the true values of all Y_j, then we would also know the true motion vectors $q_{jt} = y_{jt} - Ay_{jt-1}$ (Eq. (3)), and observation noise vectors $r_{jt} = x_{jt} - Cy_{jt}$ (Eq. (4)). The motion vectors q_{jt} with $z_{jt} = k$ are then considered samples from the k-th process noise distribution $\mathcal{N}(m_k, Q_k)$. Accordingly, the posterior $\xi_{+k}^Q = (\mu_{+k}^Q, \kappa_{+k}^Q, \nu_{+k}^Q, \Psi_{+k}^Q)$ would be computed from the sample mean, $\hat{q}_k$, and scatter matrix (c.f. [18]) $\hat{S}_k = \sum_{jt|z_{jt}=k} \left((q_{jt} - \hat{q}_k)(q_{jt} - \hat{q}_k)^\top\right)$ of those N_k vectors, as

$$\mu_{+k}^Q = \frac{\kappa_0^Q \mu_0^Q + N_k \hat{q}_k}{\kappa_0^Q + N_k} \tag{15}$$

$$\kappa_{+k}^Q = \kappa_0^Q + N_k \tag{16}$$

$$\nu_{+k}^Q = \nu_0^Q + N_k \tag{17}$$

$$\Psi_{+k}^Q = \Psi_0^Q + \hat{S}_k + \frac{\kappa_0^Q N_k}{\kappa_0^Q + N_k}(\hat{q}_k - \mu_0^Q)(\hat{q}_k - \mu_0^Q)^\top. \tag{18}$$

However, we do not know the true values of the Y_j. But a forward and backward Kalman filter can provide posterior distributions over the latent positions, i.e. the output of a Kalman smoother, which are all normal. One could then sample *pseudo*-true positions Y_t from those posteriors [17], but this is computationally demanding and may lead to slower convergence. Instead, we use the smoothed estimates of the joint distributions $\mathcal{N}(y_{jt-1}, y_{jt}|X_j, Z_j)$ to compute the distributions over the motion vectors $p(q_{jt}|X_j, Z_j)$, which are normals too and parameterized as $\mathcal{N}(q_{jt}|\mu_{jt}, \Sigma_{jt})$. In Eq. (15)-(18) we then use the expected values of $\hat{q}_k$ and $\hat{S}_k$,

$$\mathbb{E}[\hat{q}_k] = \frac{1}{N_k} \sum_{jt|z_{jt}=k} (\mathbb{E}[q_{jt}]) = \frac{1}{N_k} \sum_{jt|z_{jt}=k} (\mu_{jt}) \tag{19}$$

$$\mathbb{E}[\hat{S}_k] = \sum_{jt|z_{jt}=k} \left((\mu_{jt} - \mathbb{E}[\hat{q}_k])(\mu_{jt} - \mathbb{E}[\hat{q}_k])^\top + \Sigma_{jt}\right). \tag{20}$$

The parameters ξ_+^R and ξ_+^S of the posterior over the observation noise and spatial distribution are computed similarly (c.f. [5]) from the Kalman smoother output for Y_t.

b) Integrating out transitions π_k: The posterior distribution over π_k, given prior α_0, and the evidence consisting of the states z_{jt} following a $z_{jt-1} = k$, is again a Dirichlet distribution, hence the Dirichlet is a conjugate prior [18]. The posterior parameter vector of this distribution are

$$\alpha_{+k} = \left[\alpha_{0(0)} + N_{0|k}, \cdots, \alpha_{0(K)} + N_{K|k}\right] \tag{21}$$

and $N_{k'|k}$ are the number of $z_{jt} = k'$ that follow a $z_{jt-1} = k$. Optionally, as in [17], an extra prior weight τ can be added to self-transitions $N_{k|k}$. This enforces self-transitions even more, and lets sampling converge faster. Finally, it is even possible to integrate all π_k out analytically to make the sampler even more efficient (c.f. [20]). Using (5) and (21),

$$p(z_{jt} = k'|z_{jt-1} = k) = \int \text{Cat}(k'|\pi_k)\text{Dir}(\pi_k|\alpha_+)d\pi_k$$
$$\propto \alpha_{0(k')} + N_{k'|k} \tag{22}$$

So common transitions become more probable. This self-reinforcing property is what drives the clustering process.

c) Sampling switching states Z_j: Now we resample the latent switching states of the tracks, using the updated model parameters. Again, we will not sample values for the hidden positions Y_j, but instead directly sample from the distribution $p(Z_j|X_j, \Theta)$ for all tracks [5], [16], [17]. In [16] it is shown that, for fixed model parameters Θ, a Kalman Information filter can be used for a SLDS to efficiently compute

$$p(z_{jt}|X_j, Z_{j,-t}, \Theta) \tag{23}$$
$$\propto p(z_{jt}|z_{jt-1})p(z_{jt+1}|z_{jt})p(x_{jt}|z_{jt}, x_{j1:t-1})$$
$$\times \int p(x_{jt+1:T}|y_{jt}, z_{jt+1:T})p(y_{jt}|x_{j1:t}, z_{j1:t})dy_{jt}$$

where the first two terms are the transition probability and likelihood, for which we use the result in Equation (22). The third term is the predictive distribution of the observation, found by the forward Kalman filter, and the last integral can be analytically solved and expressed in term of the forward and backward filtering statistics (c.f. [16]).

IV. EXPERIMENTS

We have used our motion model to learn movement dynamics of pedestrians from the perspective of a vehicle.

A. Dataset

The dataset consists of pedestrian trajectories extracted from images recorded in and around various European cities, with an embedded stereo vision system in a car. Since our aim here is to learn dynamics, and not to solve detections (correct vs. false) and track assignment, unoccluded pedestrians have been manually annotated in the video frames with bounding boxes. Then, a pedestrian's distance in each frame was measured in a depth image created from stereo vision, so each bounding box results in a 2D top-down measurement (latitude and longitude) relative to the camera coordinate system. However, tracks built from these measurements contain the combined motion of the moving vehicle and pedestrian. Each track is therefore ego-motion compensated, and placed in a canonical ground plane coordinate system such that they are spatially aligned in a meaningful way. The coordinate system we used was the camera coordinate system in the last frame where the pedestrian is visible, such that paths of pedestrians crossing, or walking along, the road are best aligned. This process is illustrated in Figure 3.

The camera framerate is not constant over all recordings, thus tracks were subsampled to 0.4 seconds per frame (we have also experimented with other framerates, but we did not observe substantial differences in the learned dynamics).

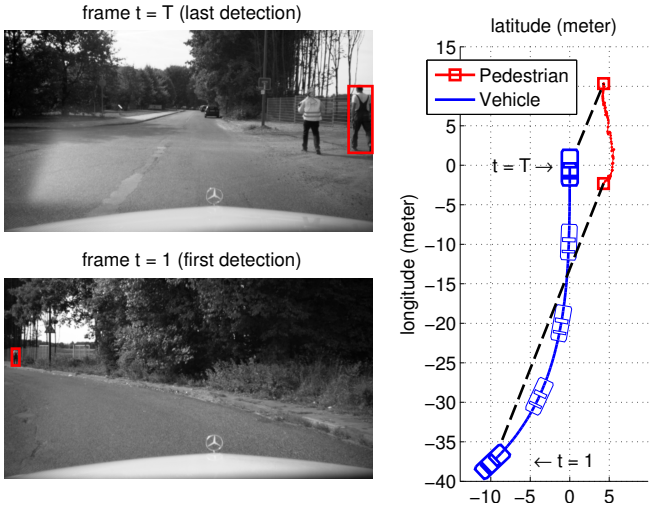


Fig. 3. Constructing a pedestrian track (in red) from the perspective of a driving vehicle (in blue). The pedestrian is seen first at the vehicle's left side and later at the right, due to the vehicle ego-motion. By projecting the ego-motion compensated track in the vehicle's final coordinate system, the track reflects a person walking along the road (i.e. along the longitude axis).

Finally, tracks that are too short or contain gaps due to occlusion are removed. The resulting data consists of 423 tracks and in a total 4103 observations, see Figure 1(c). Most tracks have 6 to 30 observations with a median of 8, though two very long tracks have 55 and 65 time steps. Due to inaccuracies in annotation, depth estimation and ego-motion compensation, the tracks are not noise free, which motivates our use of linear dynamics with observation noise.

B. Learning pedestrian dynamics

The velocity range at which pedestrians move is limited, and a filter should not adapt to high velocities that are anomalous. In our experiments we therefore use as state only a 2D latent position, and set A and C (Eq. (3) and (4)) to identity. This results in a fixed-position model with added velocity determined by the non-zero mean process noise of the switching state, similar to [5]. We applied the Gibbs sampler on the pedestrian track dataset with the upper limit $K = 20$, and set the prior $\alpha_0 = [1/K, \cdots, 1/K]$ (i.e. $\alpha = 1$), $\tau = 100$ for self-transitions, the observation noise prior $\nu_0^R = 10$ and $\Psi_0^R = \begin{bmatrix} .3^2 & 0 \\ 0 & .3^2 \end{bmatrix} \times \nu_0^R$ (i.e. prior has 10 'virtual' noise vectors with standard deviation of 0.3 in both directions, see discussion in Section III-B), the process noise prior $\mu_0^Q = [0, 0]$, $\kappa_0^Q = 1$, $\nu_0^Q = 50$ and $\Psi_0^Q = \begin{bmatrix} .3^2 & 0 \\ 0 & .3^2 \end{bmatrix} \times \nu_0^Q$, and set for the spatial prior μ_0^S to the average observation, $\kappa_0^S = 1$, $\nu_0^S = 50$, $\Psi_0^S = \begin{bmatrix} 5^2 & 0 \\ 0 & 15^2 \end{bmatrix} \times \nu_0^S$. Of course, depending on the application we could use other parameters. For example, we could express preference for more dynamics but with smaller spatial regions.

To initialize the sampler, all Z_j are assigned randomly and initial model parameters are taken from the resulting posterior distributions. Sampling converges after ~ 50 iterations, the so-called *burn-in* period [18]. In our hybrid implementation in Matlab and C++ (for sampling the switching states), one complete sampling run with $K = 20$ on the entire dataset takes ~ 4 seconds on a single 2.6 Ghz core. This could be further improved by porting all code to C++. Note that 50 iterations (≈ 200 seconds) takes less time than recording the actual dataset did ($\approx 4103 \times 0.4 = 1640$ seconds).

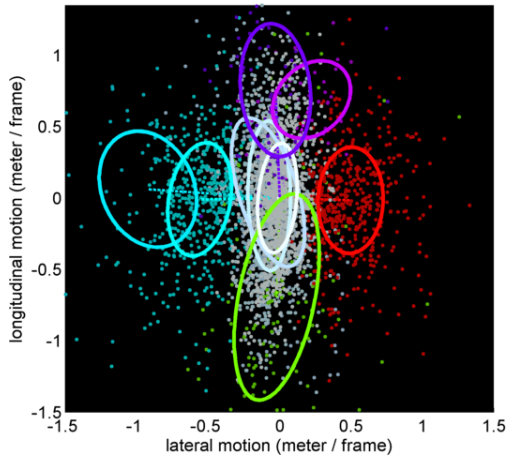
Figure 4(a) shows the sampled process noise after 100 Gibbs iterations, plotted on top of the vectors $\mathbb{E}[q_{jt}]$, the expected true velocities. The hue of each cluster is determined by the velocity direction, while velocities with higher magnitude are more saturated. Due to the Dirichlet prior, we find that only 9 of the 20 possible dynamics are used. In Figure 4(b) the selected values for the Z_j are depicted in the ground plane, after dividing tracks into four sets based on their start and end position for clarity of representation. We see that tracks that have clear lateral motion in front of the car use specialized motion dynamics (in red and blue), and can move from and to states with low lateral motion (colored white). The spatial distributions in Fig. 1(b) show that these low motion areas are located left and right of the car, where longitudinal motion is also typically observed. Indeed, the models captures that longitudinal motion occurs on the sidewalks, and crossing people move fast laterally.

C. Comparison Switching LDS vs. Mixture of LDS

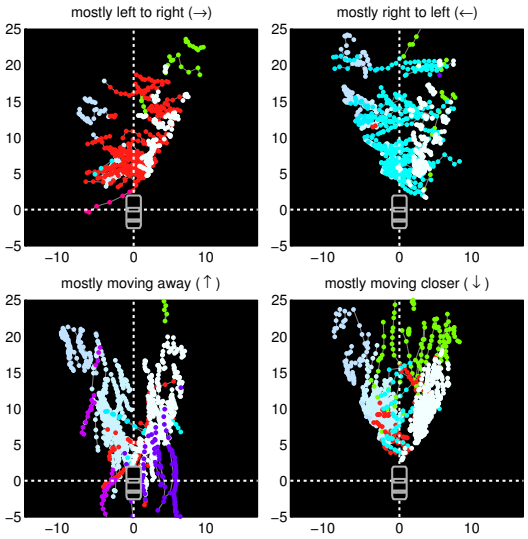
The Gibbs sampler for the SLDS can easily be adapted to a Mixture of LDS, sampling a track's switching state only once and use its dynamics for all time steps. We trained both models on the track data, with the same priors and both with spatial distributions. In figure 5(a) the data log-likelihood $p(X|Y, Z)$ of both models during Gibbs sampling is plotted. It is clear that after the burn-in period the SLDS fits the data better. This difference is illustrated for one track in figure 5(b), where the pedestrian first stands still (observed positions lie close together) and then crosses, and is even more apparent in Figure 5(c) where only the predictive lateral distribution is shown over time. The SLDS selects different states for the standing and walking phase, each with an appropriate mean velocity and low variance. The Mixture of LDS however is forced to use a single dynamic with high variance for such cases that is neither specific for walking nor for standing, and makes therefore less accurate predictions.

V. CONCLUSION

We have presented a SLDS with additional spatial distributions, and demonstrated that it can learn meaningful dynamics that are specialized for different spatial regions around the car. For crossing pedestrians, the capability of the model to switch dynamics enables specialized states for low motion and high motion with a particular direction. This results in dynamics with lower variance (process noise) and thus more accurate predictions. The Gibbs sampler we described is capable of discovering the number of dynamics, starting from random initialization. In this work tracks need to been ego-motion compensated with respect to the last frame. In future work we will look at different ways to incorporate vehicle ego-motion into the prediction.

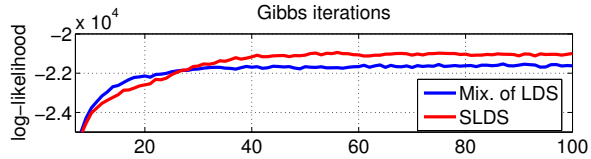


(a) Process noise of switching states

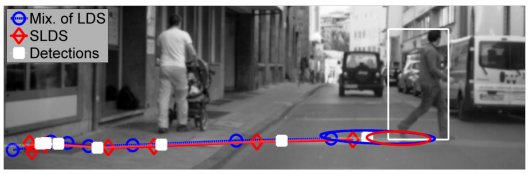


(b) Assigned dynamics (sorted by global motion direction)

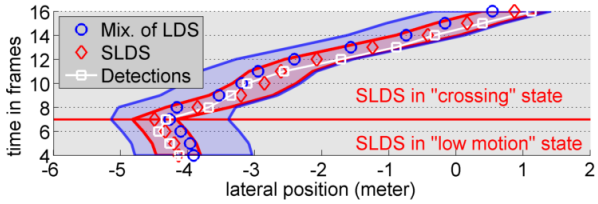
Fig. 4. (a) Ellipses show the process noise $\mathcal{N}(m_k, Q_k)$ of each dynamic. The axes in this plot represent motion, so the origin corresponds to no motion. The dots are the expected true motion vectors $\mathbb{E}[q_{jt}]$. (b) Tracks with assigned switching states of each observation shown in color of (a). Similar to figure 1(c), but splitted in four plots to reduce visual overlap.



(a) The SLDS obtains higher data log-likelihood than Mix. of LDS



(b) Vehicle view of predicted path of pedestrian crossing the road



(c) Lateral distribution of prediction over multiple frames

Fig. 5. (a) Data log-likelihood $p(X|Y, Z)$ on all training data during Gibbs sampling for the Mixture of LDS (in blue) and SLDS (in red). (b) Predictive distributions of both models for a pedestrian that first stands still on the side walk, and then crosses. Markers show mean predictions of previous frames, ellipses show predictive density for current frame. (c) For the same track, the predictive density of just the lateral position over time. The bands show the spatial range of the distribution at two std. dev., markers are on the mean. The red line indicates the moment that the SLDS switches dynamics.

References

[1] V. Romero-Cano, J. I. Nieto, and G. Agamennoni, "Unsupervised motion learning from a moving platform," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2013, pp. 104–108.

[2] C. G. Keller, C. Hermes, and D. M. Gavrila, "Will the pedestrian cross? Probabilistic path prediction based on learned motion features," *Proc. of the DAGM Symposium on Pattern Recognition*, pp. 386–395, 2011.

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

[4] X. Wang, K. T. Ma, G. W. Ng, and W. E. Grimson, "Trajectory analysis and semantic region modeling using a nonparametric Bayesian model," in *Proc. of the IEEE CVPR* , 2008, pp. 1–8.

[5] J. F. P. Kooij, G. Englebienne, and D. M. Gavrila, "A non-parametric hierarchical model to discover behavior dynamics from tracks," in *Proc. of the ECCV*. Springer Berlin Heidelberg, 2012, pp. 270–283.

[6] N. Schneider and D. M. Gavrila, "Pedestrian path prediction with recursive Bayesian filters: A comparative study," in *Proc. of the GCPR*. Springer Berlin Heidelberg, 2013, pp. 174–183.

[7] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf, "Survey of pedestrian detection for advanced driver assistance systems," *IEEE Trans. on PAMI* , vol. 32, no. 7, pp. 1239–1258, 2010.

[8] Z. Chen, D. Ngai, and N. Yung, "Pedestrian behavior prediction based on motion patterns for vehicle-to-pedestrian collision avoidance," in *Proc of the IEEE ITSC*. IEEE, 2008, pp. 316–321.

[9] J. Tao and R. Klette, "Tracking of 2d or 3d irregular movement by a family of Unscented Kalman filters," *J. Inf. Commun. Converg. Eng.*, vol. 10, no. 3, pp. 307–314, 2012.

[10] P. Scovanner and M. F. Tappen, "Learning pedestrian dynamics from the real world," in *Proc. of the IEEE ICCV*, 2009, pp. 381–388.

[11] G. Antonini, S. V. Martinez, M. Bierlaire, and J. P. Thiran, "Behavioral priors for detection and tracking of pedestrians in video sequences," *IJCV* , vol. 69, no. 2, pp. 159–180, aug 2006.

[12] P. Dollár, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *IEEE Trans. on PAMI* , vol. 34, no. 4, pp. 743–761, 2012.

[13] M. Enzweiler and D. M. Gavrila, "Monocular pedestrian detection: Survey and experiments," *IEEE Trans. on PAMI* , vol. 31, no. 12, pp. 2179–2195, 2009.

[14] M. Meuter, U. Iurgel, S.-B. Park, and A. Kummert, "Unscented Kalman filter for pedestrian tracking from a moving host," in *Proc. of the IEEE Intelligent Vehicles Symposium*, 2008, pp. 37–42.

[15] C. F. Wakim, S. Capperon, and J. Oksman, "A Markovian model of pedestrian behavior," in *Proc of the IEEE Int. Conf. on Systems, Man and Cybernetics*, vol. 4. IEEE, 2004, pp. 4028–4033.

[16] A. V. I. Rosti and M. J. F. Gales, "Rao-Blackwellised Gibbs sampling for switching linear dynamical systems," in *Proc. of the ICASSP*, vol. 1, 2004, pp. I–809.

[17] E. Fox, E. Sudderth, M. Jordan, and A. Willsky, "Bayesian nonparametric inference of switching dynamic linear models," *IEEE Trans. on Signal Processing*, vol. 59, no. 4, pp. 1569–1585, 2011.

[18] C. M. Bishop, *Pattern Recognition and Machine Learning*. Springer New York, 2006, vol. 1.

[19] V. Pavlovic, J. M. Rehg, and J. MacCormick, "Learning switching linear models of human motion," in *NIPS*, 2000, pp. 981–987.

[20] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006.