Modelling Smooth Paths Using Gaussian Processes

Meng Keat Christopher Tay[1] and Christian Laugier[2]

INRIA Rhône-Alpes, Laboratory LIG, team e-Motion
[1]tay@inrialpes.fr, [2]christian.laugier@inrialpes.fr

Summary. A generative model based on the gaussian mixture model and gaussian processes is presented in this paper. Typical motion paths are learnt and then used for motion prediction using this model. The principal novel aspect of this approach is the modelling of paths using gaussian processes. It allows the representation of smooth trajectories and avoids discretization problems found in most existing methods. Gaussian processes not only provides a comprehensive and formal theoretical framework to work with, it also lends itself naturally to path clustering using gaussian mixture models. Learning is performed using expectation maximization where the E-Step uses variational methods to maximize its lower bound before optimization over parameters are performed in the M-Step.

1 Introduction

The analysis of motion patterns is important for performing motion prediction in robotic applications. For most environments, objects do not move around randomly. The objects often move according to a set of typical motion pattern. One such example is the traffic cross junction where vehicles follow a certain pattern of movements according to the traffic rules. It is hence possible to learn the set of typical motions in a certain scene. Prediction can then be performed based on the knowledge of the typical motions learnt from past observations. Such predictions can be used in applications such as anomaly detection or improved navigation capabilities [5] [3] [4].

Most existing models for modelling motion patterns requires the discretization of the state space [8] [5] [3]. These approaches mostly model motion trajectories as transitions between discretized states. The main disadvantage of discretization is the need to determine the discretization of the state spaces and the association of observations to these discretized state spaces.

Most existing models are based on the "learn then predict" approach where a database of observations are available to perform batch learning before prediction can then be performed. In these approaches, the discretization problem again makes the problem of path clustering unnatural and difficult. However, [8] described a "learn and predict" approach where learning and prediction can be performed on line while continuously making a stream of observations. The

C. Laugier and R. Siegwart (Eds.): Field and Service Robotics, STAR 42, pp. 381–390, 2008.
springerlink.com

model described in [8] does not require the clustering of paths because the model is able to adapt the topology of the state space within the hidden markov model [6] using growing neural gas [2].

In this paper, we propose a novel path modelling approach based on gaussian processes [7]. Gaussian processes makes it possible to represent paths as continuous functions in a probabilistic manner. The problem of discretization is conveniently side stepped. Prediction on the future path taken can be performed in a theoretically proper probabilistic framework.

The proposed model enables one to pose questions in a probabilistic manner such as the distribution of predicted trajectories (see fig. 1). Similarly, it is also possible to obtain probability distribution of trajectories in unobserved portions (see fig. 2). Since we model a single motion trajectory as a gaussian process, the probability distribution over trajectories are analytic and easily evaluated.

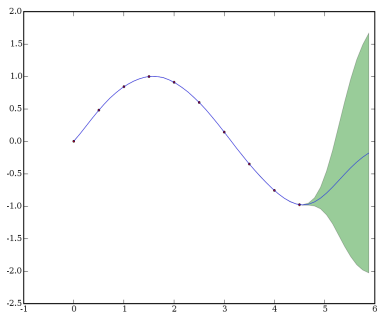

Fig. 1. Variance and mean of motion trajectory prediction. Dots represents observations.

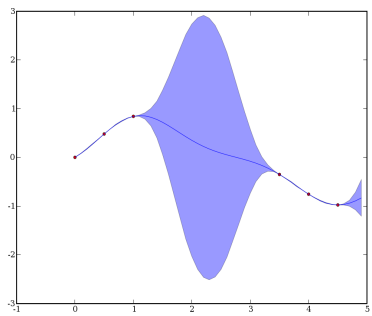

Fig. 2. Variance and mean of unobserved motion trajectory portion

The approach described in this paper falls into the "learn then predict" category. Thanks to the gaussian process model for paths, the well known mixture of gaussians model is naturally used to perform clustering. In this paper, the expectation maximization approach is adopted. During the E-step, variational approximation is used to estimate the probability distribution over latent variables. The M-step maximizes the lower bound with respect to gaussian process parameters. This approach enables the automatic determination of the number of clusters while obtaining a point estimate for gaussian process parameters.

In section 2, we present the generative model based on the gaussian mixture model to model the set of paths in a scene. Learning of the model and its parameters are described in section 3 using expectation maximization. Path predictions based on the gaussian mixture model and gaussian process are described in section 4. The paper finishes with some results in section 5 and conclusion in section 6.

2 Probabilistic Model

We consider the problem of learning a set of typical paths in a certain scene. Inputs into the system consists of the path of objects in the scene. The path taken by each object is a sequence of $(x, y)_t$ cartesian coordinates representing the position of the object at various instances. Each typical path is represented by a gaussian process and the different typical paths in a scene is modelled using the mixture of gaussians.

2.1 Representing Typical Paths Using Gaussian Processes

A short description of gaussian processes is presented here. For more details, please refer to [7]. A gaussian process is a generalization of the gaussian probability distribution in function space. Given a collection of random variables, any finite number of which are gaussian distributed [7]. In this case, the set of random variables are $\{f(x_1), f(x_2), ..., f(x_N)\}$. It can be represented mathematically using the mean function and covariance function $k(x, x')$ as:

$$f(x) \sim G(m(x), k(x, x')) \tag{1}$$

$$m(x) = E[f(x)] \tag{2}$$

$$k(x, x') = E[(f(x) - m(x))(f(x') - m(x'))] \tag{3}$$

Where $G(\mu, \Sigma)$ represents a gaussian distribution with mean μ and covariance Σ. $k(x, x')$ is expressed as a covariance function which are functions with domain from the input space. In this paper, the squared exponential covariance function is adopted:

$$cov(f(x), f(x')) = k(x, x') = \theta_0^2 exp - \frac{(x - x')^2}{\theta_1^2} + \theta_2^2 \delta_{ij} \tag{4}$$

In our model, each path corresponding to a typical path that an object takes in a given environment is represented as 2 gaussian processes, one each to represent the path in x and y axes as we assume the movements in the x and y axes to be independent. The mean of these gaussian processes is the mean function (in the x and y axes each) representing the path. A gaussian distribution over the different paths for a typical path can thus be represented using gaussian processes. As the gaussian distribution takes place within a function space, it is infinite dimensional and no discretization is required.

A single observation of a path is represented as two D dimensional vectors. One vector represents the sequence of positions along the cartesian x axis and the other for the y axis. The likelihood for a set of N path observations for a single typical path can then be written as:

$$L_x = \prod_{n=1}^{N} G(x_n | \mu_x, \Sigma_x) \tag{5}$$

$$L_y = \prod_{n=1}^{N} G(y_n | \mu_y, \Sigma_y) \tag{6}$$

Where x_n and y_n are vectors of x and y positions for the nth observation. μ_x, μ_y, Σ_x and Σ_y are the mean vectors and covariances of the x and y positions for the typical path. In almost all cases, the sequence of observations of position for path observations are all of different length. In this case, a fixed dimension D can be chosen and the D positions can be obtained from D samples equally distributed along the length of the path.

2.2 Multiple Typical Trajectories Using Gaussian Mixture Model

Since each observed path corresponds to two D dimensional vectors and that a typical motion path is gaussian distributed, the observed paths are gaussian distributed from the generative point of view. A single typical motion path is a D dimensional gaussian distribution and this view can be easily extended to the case of representing several typical motion paths using mixture of gaussians. Each path observation n is probabilistically associated with one of the mixture components k via the variable $Z_{nk} = \{0, 1\}$ where $\sum_k Z_{nk} = 1$. The prior distribution for all Z_{nk}, all path observations and component means are:

$$P(Z|\pi) = \prod_{n=1}^{N} \prod_{k=1}^{K} \pi_k^{Z_{nk}} \tag{7}$$

$$P(X|Z, \mu_x, \Theta) = \prod_{n=1}^{N} \prod_{k=1}^{K} G(x_n|\mu_{xk}, C(\Theta))^{Z_{nk}} \tag{8}$$

$$P(Y|Z, \mu_x, \Theta) = \prod_{n=1}^{N} \prod_{k=1}^{K} G(y_n|\mu_{yk}, C(\Theta))^{Z_{nk}} \tag{9}$$

$$P(\mu_x|\Theta) = \prod_{k=1}^{K} G(\mu_{xk}|m_{xk}, b_{xk}C(\Theta)) \tag{10}$$

$$P(\mu_y|\Theta) = \prod_{k=1}^{K} G(\mu_{yk}|m_{yk}, b_{yk}C(\Theta)) \tag{11}$$

where x_n and y_n are the x and y vectors of a path observation. μ_{xk} and μ_{yk} are x and y means of cluster component k, with each component weight being π_k. m_{xk} and m_{yk} are means of the prior distribution over the mean components. b_{xk} and b_{yk} are scaling constants to $C(\Theta_k)$, the covariance matrix constructed using the squared exponential function (eq. 5) parameterized by Θ_k.

The decomposition of the joint distribution over the variables can be expressed as:

$$\begin{aligned} P(X, Y, \mu_x, \mu_y, Z|\pi, \Theta) = {} & P(X|Z, \mu_x, \Theta)P(Y|Z, \mu_y, \Theta) \\ & P(\mu_x|\Theta)P(\mu_y|\Theta)P(Z|\pi) \end{aligned} \tag{12}$$

3 Model Learning

The parameters for the probabilistic model in section 2 are learnt from the data using the expectation maximization (EM) algorithm. Consider the general model of data X, latent variables Z and model parametrized by Θ. The goal in EM is to maximize the log likelihood:

$$\begin{aligned} \ln P(X|\Theta) &= \ln \int_Z P(X, Z|\Theta) dZ \\ &\geq L(Q, \Theta) + KL(Q \parallel P) \end{aligned} \tag{13}$$

where $Q(Z)$ is the joint distribution over latent parameters. $KL(Q \parallel P)$ is the kullback-leibler divergence and lower bound L defined as:

$$L(Q, \Theta) = \int_Z Q(Z) \ln \frac{P(X, Z|\Theta)}{Q(Z)} dZ \tag{14}$$

Maximization of eqn. 13 is equivalent to minimizing $KL(Q \parallel P)$. This occurs when $Q(Z) = P(X|\Theta)$. This minimization can be performed by approximating $Q(Z)$ with a constrained family of variational approximations:

$$Q(Z) = \prod_i Q_i(Z_i) \tag{15}$$

In this case, the minimization of $KL(Q \parallel P)$ can be obtained by iteratively computing for each $Q_i(Z_i)$:

$$Q_i(Z_i) = \frac{exp\langle \ln P(X, Z|\Theta) \rangle_{i \neq j}}{\int exp\langle \ln P(X, Z|\Theta) \rangle_{i \neq j} dz_i} \tag{16}$$

This variational maximization of the lowerbound $L(Q, \Theta)$ constitutes the E-Step. During the M-Step, the log likelihood is maximized with respect to parameter Θ:

$$\Theta_{max} = \arg\max_{\Theta} \ln P(X|\Theta) \tag{17}$$

The E-Step and M-Step are iteratively computed till convergence.

We shall next present in more detail the expectation and maximization step especially within the context of trajectory modelling.

3.1 Expectation Step

This technique can be applied to the mixture of gaussians model. A similar model can be found in [1]. The variational approximation and posteriors can thus be obtained as follows:

$$Q(\mu_x, \mu_y, Z) = Q(\mu_x) Q(\mu_y) Q(Z) \tag{18}$$

$$Q(Z) = \prod_{n=1}^{N} \prod_{k=1}^{K} r_{nk}^{z_{nk}} \tag{19}$$

$$Q(\mu_x) = \prod_{k=1}^{K} G(\mu_{xk}|M_{xk}, B_{xk}C(\Theta)) \tag{20}$$

$$Q(\mu_y) = \prod_{k=1}^{K} G(\mu_{yk}|M_{yk}, B_{yk}C(\Theta)) \tag{21}$$

$$\tag{22}$$

The variational update equations are as follows:

$$\begin{aligned}\ln P_{nk} = &-\ln|C(\Theta)| - D\ln 2\pi \\ &-\frac{1}{2}\left\langle (x_n - \mu_{xk})^T C(\Theta)^{-1}(x_n - \mu_{xk})\right\rangle_{\mu_x k} \\ &-\frac{1}{2}\left\langle (y_n - \mu_{yk})^T C(\Theta)^{-1}(x_n - \mu_{yk})\right\rangle_{\mu_y k} + \ln \pi_k\end{aligned} \tag{23}$$

$$r_{nk} = \frac{P_{nk}}{\sum_j P_{nj}} \tag{24}$$

$$N_k = \sum_{n=1}^{N} r_{nk} \tag{25}$$

$$\bar{x}_k = \frac{1}{N_k}\sum_{n=1}^{N} r_{nk}x_n \tag{26}$$

$$B_{xk}^{-1} = N_K + b_{xk}^{-1} \tag{27}$$

$$M_{xk} = (N_K\bar{x}_k + m_{xk}b_{xk}^{-1})B_{xk} \tag{28}$$

Equations 26, 27 and 28 are similar for $\bar{y}_k, B_{yk}^{-1}$ and M_{yk}.

3.2 Maximization

The maximization involves the optimization of the marginal log likelihood given by:

$$\ln P(X, Y|\Theta) = \ln \sum_Z \int_{\mu_x}\int_{\mu_y} P(X, Y, \mu_x, \mu_y, Z|\pi, \Theta)d\mu_x d\mu_y \tag{29}$$

Since eqn 29 is not tractable, we can approximate it with the lower bound (refer also to eqn. 14):

$$\begin{aligned}L = &\langle lnP(X|Z, \mu_x, \Theta)\rangle + \langle lnP(Y|Z, \mu_y, \Theta)\rangle \\ &\langle lnP(\mu_x|\Theta)\rangle + \langle lnP(\mu_y|\Theta)\rangle + \langle lnP(Z|\pi)\rangle \\ &-\langle lnQ(\mu_x)\rangle - \langle Q(\mu_y)\rangle - \langle Q(Z)\rangle\end{aligned} \tag{30}$$

Optimization of the lower bound in equation 30 with respect to the gaussian process parameters Θ. Optimization algorithms such as conjugate gradient or grid based search can be used.

By setting the first derivate of the lower bound with respect to π to zero and imposing the constraint of $\sum_i \pi_i = 1$ with lagrange multipliers, π can be updated by:

$$\pi_i = \frac{N_i}{N} \tag{31}$$

4 Path Prediction

When performing path prediction, the input is a partially observed path of dimension $M < D$. For the case of a D dimensional gaussian with x_1 of dimension M and x_2 of dimension $D - M$:

$$P'(x_1, x_2) \sim G\left(\begin{bmatrix}\mu_1 \\ \mu_2\end{bmatrix}, \begin{bmatrix}\Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}\right) \tag{32}$$

The probability of a partial path observation of dimension M belonging to a gaussian of dimension D is evaluated by integrating over the $D - M$ dimensions of the gaussian distribution to yield the marginal gaussian distribution:

$$P'(x_1) \sim G(\mu_1, \Sigma_{11}) \tag{33}$$

The prediction of a path x_2 given observation x_1 can be obtained by the gaussian conditional distribution for each cluster k:

$$P'_k(x_2|x_1) \sim G\left(\mu_2 + \Sigma_{21}\Sigma_{11}(x_1 - \mu_1), \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{21}^T\right) \tag{34}$$

In order to choose the suitable clusters that corresponds to the observations made so far, the mahanalobis distance can be calculated and then gated based on the appropriate chi-square values.

Figure 3 shows an example of the prediction where the predicted path mean and variance are represented by the 'bars'. Clusters for prediction were selected according to the chi-square statistic corresponding to the 95% confidence interval. For each cluster, the gaussian distribution of the predicted path can be obtained using eq. 34.

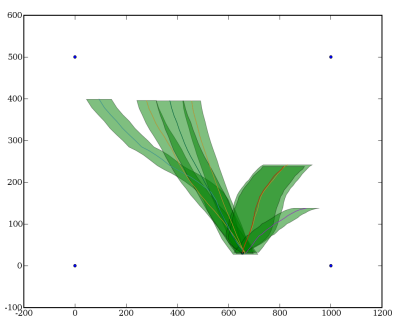

Fig. 3. Prediction paths and path variance to 2 standard deviations

5 Results

Experiments were conducted based on simulated data. 36 trajectories typical motion were manually defined for the generation of the different motions. Figure 4 shows the ensemble of motion in the training data set.

It has been noted in [1] that the variational updates were initialized using the K-Means clustering algorithm. We assumed a maximal number of 100 clusters during the K-Means stage. And the expectation maximization algorithms were performed with 100 clusters with equal cluster component weights. Result of the k-means cluster algorithm is show in figure 5.

The number of cluster components can be determined from the weight of the cluster components. The learning algorithm on the training data set allowed

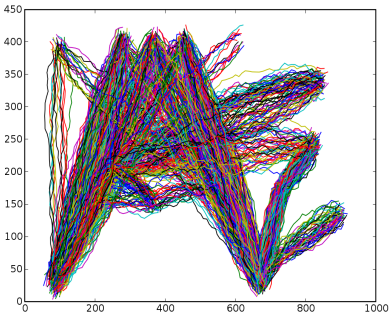

Fig. 4. Training Data Motions

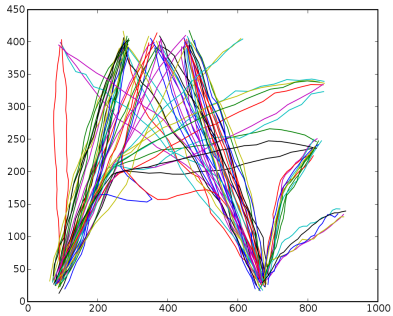

Fig. 5. The output from the K-Means algorithm with 100 clusters

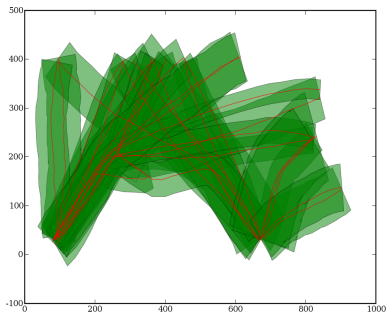

Fig. 6. 26 typical motion recovered from the training data (different gaussian process hyperparameters for each cluster)

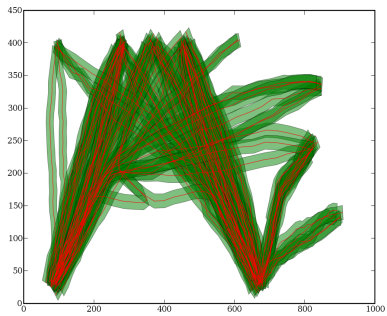

Fig. 7. Typical motion recovered from the training data set (same gaussian process hyperparameters across all clusters)

the suppression of clusters with component weights of nearly zero ($< 10^{-5}$) very quickly early in the EM iterations. The learning algorithm allows the domination of gaussian components over the others when clustering in the same regions. 26 gaussian components were recovered after the clustering as shown in figure 6.

The suppression of unrepresentative cluster components with negligible weights works well only when gaussian process hyperparameters are different for each cluster. However, when the gaussian process hyperparameters are shared across clusters, the variance of the different clusters are not well expressed and as a result, it is difficult to obtain cluster components with negligible component weights (see figure 7).

6 Conclusion and Perspectives

We described a generative approach to model motion paths using gaussian processes. The typical motion paths are smooth and avoids the problems associated with discretization. The representation of motion paths with gaussian process lends itself naturally to clustering using the gaussian mixture model. The "learn then predict" approach is performed using the EM algorithm. The lower bound of the EM algorithm is maximized using variational methods in the E-Step. The M-Step optimizes the model parameters.

We are currently working on extending this method such that it can work in real time in an incremental fashion. It is also interesting to explore the modelling of dependencies between the x and y axes. Applications to path planning are also envisaged.

Finally the authors would like to acknowledge the financial support from the French Embassy of Singapore. Thanks goes as well to Dizan Vasquez and Pierre Dangauthier for their interesting discussions.

References

1. Corduneanu, A., Bishop, C.M.: Variational bayesian model selection for mixture distributions. In: Proceedings Eighth International Conference on Artificial Intelligence and Statistics, pp. 27–34 (2001)
2. Fritzke, B.: A growing neural gas network learns topologies. Advances in Neural Information Processing Systems (1995)
3. Hu, W., Xiao, X., Fu, Z., Xie, D., Tan, T., Maybank, S.: A system for learning statistical motion patterns. IEEE PAMI 28(9) (September 2006)
4. Kruse, E., Gutsche, R., Wahl, F.M.: Acquisition of statistical motion patterns in dynamic environments and their application to mobile robot motion planning. In: Proc. of the IEE Int. Conf. on Robotics and Systems, pp. 712–717 (1997)
5. Bennewitz, M., Burgard, W., Cielniak, G., Thrun, S.: Learning motion patterns of people for compliant robot motion. The International Journal of Robotics Research (IJRR) 24(1) (2005)
6. Rabiner, L.R.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceedings of the IEEE 77(2), 257–286 (1989)

7. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. MIT Press, Cambridge (2006)
8. Vasquez, D., Fraichard, T., Aycard, O., Laugier, C.: Intentional motion on-line learning and prediction. In: Proc. of the Int. Conf. on Field and Service Robotics, Port Douglas (AU) (July 2005)