Hierarchical Gaussian Process Latent Variable Models

Neil D. Lawrence NEILL@CS.MAN.AC.UK
School of Computer Science, University of Manchester, Kilburn Building, Oxford Road, Manchester, M13 9PL, U.K.

Andrew J. Moore A.MOORE@DCS.SHEF.AC.UK
Dept of Computer Science, University of Sheffield, Regent Court, 211 Portobello Street, Sheffield, S1 4DP, U.K.

Abstract

The Gaussian process latent variable model (GP-LVM) is a powerful approach for probabilistic modelling of high dimensional data through dimensional reduction. In this paper we extend the GP-LVM through hierarchies. A hierarchical model (such as a tree) allows us to express conditional independencies in the data as well as the manifold structure. We first introduce Gaussian process hierarchies through a simple dynamical model, we then extend the approach to a more complex hierarchy which is applied to the visualisation of human motion data sets.

1. Introduction

The Gaussian process latent variable model (Lawrence, 2004; Lawrence, 2005) has proven to be a highly effective approach to probabilistic modelling of high dimensional data that lies on a non-linear manifold (Grochow et al., 2004; Urtasun et al., 2005; Urtasun et al., 2006; Ferris et al., 2007). The curse of dimensionality is finessed by assuming that the high dimensional data is intrinsically low dimensional in nature. This reduces the effective number of parameters in the model enabling good generalisation from very small data sets using non-linear models (even when the dimensionality of the features, d, is larger than the number of data points, N).

One alternative to manifold representations when modelling high dimensional data is to develop a latent variable model with sparse connectivity to explain the data. For example tree structured models have been suggested for modelling images (Williams & Feng, 1999; Feng et al., 2002; Awasthi et al., 2007), for object recognition (Felzenszwalb & Huttenlocher, 2000; Ioffe & Forsyth, 2001) and human pose estimation (Ramanan & Forsyth, 2003; Sigal et al., 2004; Lan & Huttenlocher, 2005). From the probabilistic perspective (Pearl, 1988) the tree structures (and other sparse probabilistic models) offer a convenient way to specify conditional independencies in the model. In general, it is not clear how such conditional independencies can be specified within the context of dimensional reduction. In this paper we will show how we can construct our dimensionality reduction in a hierarchical way allowing us to concurrently exploit the advantages of expressing conditional independencies and low dimensional non-linear manifolds.

1.1. GP-LVMs

The Gaussian process latent variable model (GP-LVM) is a fully probabilistic, non-linear, latent variable model that generalises principal component analysis. The model was inspired by the observation that a particular probabilistic interpretation of PCA is a product of Gaussian process models each with a *linear* covariance function. Through consideration of non-linear covariance functions a non-linear latent variable model can be constructed (Lawrence, 2004; Lawrence, 2005).

An important characteristic of the GP-LVM is the ease and accuracy with which probabilistic reconstructions of the data can be made, given a (possibly new) point in the latent space. This characteristic is exploited in several of the successful applications of the GP-LVM: learning style from motion capture data (Grochow et al., 2004) learning a prior model for tracking (Urtasun et al., 2005; Urtasun et al., 2006) and robot simultaneous localisation and mapping (Ferris et al., 2007). All make use of smooth mappings from the latent space to the data space.

The probabilistic approach to non-linear dimensionality reduction (MacKay, 1995; Bishop et al., 1998) is to formulate a latent variable model, where the la-

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

tent dimension, q, is lower than the data dimension, d. The latent space is then governed by a prior distribution $p(\mathbf{X})$. The latent variable is related to the observation space through a probabilistic mapping,

$$y_{ni} = f_i(\mathbf{x}_n; \mathbf{W}) + \epsilon_n,$$

where y_{ni} is the ith feature of the nth data point and ϵ_n is a noise term that is typically taken to be Gaussian[1], $p(\epsilon_n) = N\left(\epsilon_n|0, \beta^{-1}\right)$. and $\mathbf{W}$ is a matrix of mapping parameters. If the prior is taken to be independent across data points the marginal likelihood of the data can be written as

$$p(\mathbf{Y}|\mathbf{W}) = \int \prod_{n=1}^{N} p(\mathbf{y}_n|\mathbf{x}_n, \mathbf{W})\, p(\mathbf{x}_n)\, \mathrm{d}\,\mathbf{X},$$

where $p(\mathbf{y}_n|\mathbf{x}_n) = \prod_{i=1}^{d} N\left(y_{in}|f_{in}(\mathbf{x}_n), \beta^{-1}\right)$. If the mapping is chosen to be linear, $f_i(\mathbf{x}_n) = \mathbf{w}_i^{\mathrm{T}}\mathbf{x}_n$, and the prior over the latent variables is taken to be Gaussian, then the maximum likelihood solution of the model spans the principal subspace of the data (Tipping & Bishop, 1999). However, if the mapping is non-linear it is unclear, in general, how to propagate the prior distribution's uncertainty through the non-linearity.

The alternative approach taken by the GP-LVM is to place the prior distribution over the mappings rather than the latent variables. The mappings may then be marginalised and the marginal likelihood optimised with respect to the latent variables,

$$p(\mathbf{Y}|\mathbf{X}) = \int \prod_{i=1}^{d}\prod_{n=1}^{N} p(y_{in}|f_{in})\, p(\mathbf{f}|\mathbf{X})\, \mathrm{d}\,\mathbf{f}. \quad (1)$$

It turns out that if the prior is taken to be a Gaussian process that is independent across data dimensions and has a *linear* covariance function (thus restricting the mappings to linearity) the maximum likelihood solution with respect to the embeddings is given by principal component analysis. However, if the covariance function is one which allows non-linear functions (*e.g.* the RBF kernel) then the model provides a probabilistic non-linear latent variable model.

There are several advantages to marginalising the mapping rather than the latent variable. In particular, a non-linear latent variable model that does not require approximations is recovered. Additionally, we now have a probabilistic model of the data that is expressed in the form $p(\mathbf{Y}|\mathbf{X})$ rather than the more usual form $p(\mathbf{Y}|\mathbf{W})$. Our model is non-parametric, the size of $\mathbf{X}$ is $N \times q$ and each row of $\mathbf{X}$, given by $\mathbf{x}_n$, is associated with an data observation, $\mathbf{y}_n$. This makes it much easier to augment the model with additional constraints or prior information about the data. Interesting examples include adding dynamical priors in the latent space (Wang et al., 2006; Urtasun et al., 2006) or constraining points in the latent space according to intuitively reasonable visualisation criteria (Lawrence & Quiñonero Candela, 2006). In this paper we further exploit this characteristic, proposing the hierarchical Gaussian process latent variable models. In the next section we will illustrate the nature of a simple (one layered) hierarchical model by considering a novel approach to incorporating *dynamics* into the GP-LVM, then in Section 3 we consider more complex hierarchies, focussing on models of human body motion.

[1] We denote a Gaussian distribution over $\mathbf{z}$ with mean $\boldsymbol{\mu}$ and covariance $\boldsymbol{\Sigma}$ by $N(\mathbf{z}|\boldsymbol{\mu}, \boldsymbol{\Sigma})$.

2. Dynamics via a Simple Hierarchy

In a standard latent variable model setting, a dynamical system is modelled by constructing a dynamical prior distribution that, for tractability, typically takes the form of a Markov chain, $p(\mathbf{X}) = p(\mathbf{x}_1)\prod_{t=2}^{T} p(\mathbf{x}_t|\mathbf{x}_{t-1})$. The latent variable, $\mathbf{X}$, is marginalised as before, inducing correlations between *neighbouring* time points. In the GP-LVM we marginalise with respect to the mapping, once this marginalisation is performed, integrating out the latent space and any associated dynamical prior analytically intractable. However, we may instead choose combine a dynamical prior with the GP-LVM likelihood and seek a *maximum a posteriori* (MAP) solution.

2.1. Gaussian Process Dynamics

Seeking a MAP solution is the approach taken by (Wang et al., 2006) who make use of an autoregressive Gaussian process prior to augment the GP-LVM with dynamics. The utility of the approach is nicely demonstrated in the context of tracking by (Urtasun et al., 2006) who show that through dynamics the track is sustained even when the subject is fully occluded for several frames.

The autoregressive approach works by predicting the next temporal location in the latent space given the previous, *i.e.* it models $p(\mathbf{x}_t|\mathbf{x}_{t-1})$. However, since the prediction is given by a Gaussian process it is a unimodal prediction over $\mathbf{x}_t$ given $\mathbf{x}_{t-1}$. This can present problems: consider, for example, the case of a subject walking for several paces before breaking into a run. We expect the walking steps to be broadly periodic, each point from the cycle projecting into a similar point in latent space. However, at the point the sub-

ject begins to break into a run, there is a bifurcation in the dynamics. Such a bifurcation can not be captured correctly by unimodal autoregressive dynamics. Additionally, the autoregressive approach assumes that samples are taken at uniform intervals (perhaps with occasional drop outs) which may not always be the case (as we shall see in Section 3).

2.2. A Simple Hierarchical Model

As the first illustration of a hierarchical GP-LVM we consider an alternative implementation of dynamics. Just as (Wang et al., 2006) we implement the dynamics through a Gaussian process prior and seek a MAP solution. However, in contrast to their approach, our model is not autoregressive. We simply place a Gaussian process prior over the latent space, the inputs for which are given by the time points, $\mathbf{t}$. This approach alleviates the requirement for uniform intervals between time samples and, because the prior over the latent space is no longer a function of the location in latent space, allows the path in latent space to bifurcate at points where the subject, for example, breaks into a run.

Given a set of d-dimensional observations from a motion capture sequence, $\mathbf{Y} = [\mathbf{y}_{1,:}, \ldots, \mathbf{y}_{T,:}]^{\mathrm{T}} \in \Re^{T\times d}$, we seek to model them by a Gaussian process latent variable model,

$$p(\mathbf{Y}|\mathbf{X}) = \prod_{j=1}^{d} N(\mathbf{y}_{:,j}|\mathbf{0}, \mathbf{K}_x), \tag{2}$$

where $\mathbf{y}_{:,j}$ is a column of the design matrix $\mathbf{Y}$, each element being from a different point in the time sequence, and $\mathbf{K}_x$ is a covariance matrix (or kernel) which depends on the q-dimensional latent variables, $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_T]^{\mathrm{T}} \in \Re^{T\times q}$, each element being given by, for example,

$$k_x(\mathbf{x}_i, \mathbf{x}_j) = \sigma^2_{\mathrm{rbf}} \exp\left(-\frac{\|\mathbf{x}_i - \mathbf{x}_j\|^2}{2l_x^2}\right) + \sigma^2_{\mathrm{white}}\delta_{ij},$$

which is an radial basis function (RBF) covariance matrix with a noise term. The parameters of this covariance are the variances of the different terms σ^2_{rbf}, $\sigma^2_{\mathrm{white}}$ and the length scale of the RBF term, l_x. In (2) we have dropped the dependence on these parameters to avoid cluttering notation.

We construct a simple hierarchy by placing a prior over the elements of $\mathbf{X}$. We wish this prior to be temporally smooth, ensuring two points from $\mathbf{X}$ that are temporally close, *e.g.* $\mathbf{x}_{i,:}$ and $\mathbf{x}_{j,:}$ are also nearby in space. A suitable prior is given by a Gaussian process

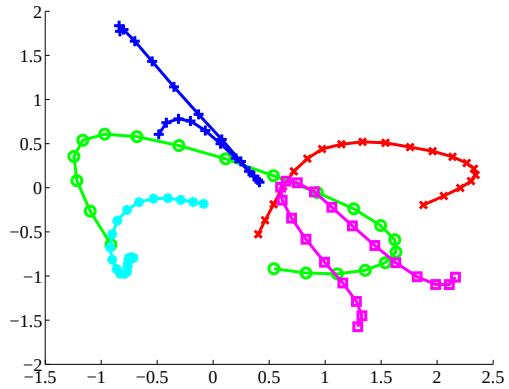

Figure 1. Typical sample paths for the RBF covariance function as temporal prior over the latent space.

in which the input to the Gaussian process is time,

$$p(\mathbf{X}|\mathbf{t}) = \prod_{i=1}^{q} N(\mathbf{x}_{:,i}|\mathbf{0}, \mathbf{K}_t), \tag{3}$$

where $\mathbf{t} \in \Re^{T\times 1}$ is the vector of times at which we observed the sequence, $\mathbf{x}_{:,j}$ is the jth column of $\mathbf{X}$ and $\mathbf{K}_t$ is a covariance matrix of the form

$$k_t(t_i, t_j) = \varsigma^2_{\mathrm{rbf}} \exp\left(-\frac{(t_i - t_j)^2}{2l_t^2}\right) + \varsigma^2_{\mathrm{white}}.$$

For a two dimensional latent space typical sample paths for this covariance function are shown in Figure 1.

The temporal prior in (3) can be combined with the GP-LVM likelihood in (2) to form a new model,

$$p(\mathbf{Y}|\mathbf{t}) = \int p(\mathbf{Y}|\mathbf{X})\, p(\mathbf{X}|\mathbf{t})\, \mathrm{d}\mathbf{X},$$

unfortunately such a marginalisation is intractable. Instead, we seek to make progress by seeking a *maximum a posteriori* (MAP) solution, maximising

$$\log p(\mathbf{X}|\mathbf{Y}, \mathbf{t}) = \log p(\mathbf{Y}|\mathbf{X}) + \log p(\mathbf{X}|\mathbf{t}) + \text{const.}$$

with respect to $\mathbf{X}$. The first term in this equation is the standard objective function for the GP-LVM, the second term has the form

$$\log p(\mathbf{X}|\mathbf{t}) = -\frac{1}{2}\prod_{j=1}^{q} \mathbf{x}_{:,j}^{\mathrm{T}} \mathbf{K}_t^{-1} \mathbf{x}_{:,j} + \text{const.},$$

where $\mathbf{x}_{:,j}$ is the jth column of $\mathbf{X}$. The gradient of this additional term may also be found,

$$\frac{\mathrm{d}\log p(\mathbf{X}|\mathbf{t})}{\mathrm{d}\mathbf{X}} = \mathbf{K}_t^{-1}\mathbf{X}$$

and combined with the gradient of $\log p(\mathbf{Y}|\mathbf{X})$ to find the MAP solution. This can easily be found using gradient based methods.

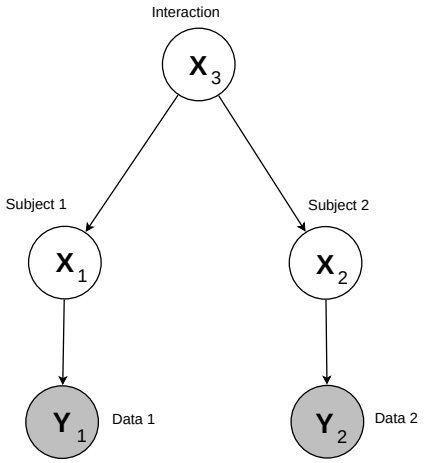

Figure 2. A simple hierarchy for capturing interaction between two subjects where $\mathbf{Y}_1$ is the data associated with subject 1, $\mathbf{Y}_2$ is that of subject 2. Each of these variable sets is then controlled by latent variables, $\mathbf{X}_1$ and $\mathbf{X}_2$. These latent variables are in turn controlled by $\mathbf{X}_3$.

3. More Complex Hierarchies

We now turn to a slightly more complex hierarchy than the dynamical model described in the previous section. Consider a motion capture example with multiple subjects interacting. Given the form of the interaction it should be possible to model each subject independently. This form of conditional independence is well captured by a hierarchical model such as that shown in Figure 2.

The joint probability distribution represented by this graph is given by

$$\begin{aligned} p(\mathbf{Y}_1, \mathbf{Y}_2) &= \int p(\mathbf{Y}_1|\mathbf{X}_1)\dots \\ &\quad \times \int p(\mathbf{Y}_2|\mathbf{X}_2)\dots \\ &\quad \times \int p(\mathbf{X}_1, \mathbf{X}_2|\mathbf{X}_3)\,\mathrm{d}\mathbf{X}_3\mathrm{d}\mathbf{X}_2\mathrm{d}\mathbf{X}_1 \\ &\quad , \end{aligned}$$

where each conditional distribution is given by a Gaussian process. However, once again, the required marginalisations are not tractable. We therefore turn to MAP solutions for finding the values of the latent variables. For this model, this means maximisation of

$$\begin{aligned} \log p(\mathbf{X}_1, \mathbf{X}_2\mathbf{X}_3|\mathbf{Y}_1, \mathbf{Y}_2) &= \log p(\mathbf{Y}_1|\mathbf{X}_1) \\ &\quad + \log p(\mathbf{Y}_2|\mathbf{X}_2) \\ &\quad + \log p(\mathbf{X}_1, \mathbf{X}_2|\mathbf{X}_3)\,, \end{aligned}$$

which is the sum of three Gaussian process log likelihoods. The first two terms are associated with the two subjects. The third term provides co-ordination between the subjects.

3.1. Two Interacting Subjects

To demonstrate this hierarchical model we considered a motion capture data set consisting of two interacting subjects. The data, which was taken from the CMU MOCAP data base[2], consists of two subjects[3] that approach each other and 'high five'.

The algorithm for optimisation of the latent variables proceeded as follows:

1. Initialise each leaf node's latent variable set $(\mathbf{X}_1, \mathbf{X}_2)$ through principal component analysis of the corresponding data set $(\mathbf{Y}_1, \mathbf{Y}_2)$.
2. Initialise the root node's latent variable set $(\mathbf{X}_3)$ through principal component analysis of the concatenated latent variables of its dependents $[\mathbf{X}_1\ \mathbf{X}_2]$.
3. Optimise jointly the parameters of the kernel matrices for each Gaussian process model and the latent variable positions $(\mathbf{X}_1, \mathbf{X}_2, \mathbf{X}_3)$.

The original data is sampled at 120 frames per second. We extracted frames 50 to 113, sub-sampling to 30 frames per second, frames 114 to 155 at the full sample rate and frames 156 to 232 sub-sampling at 30 frames per second. This gives a data set with a variable sample rate. In the context of this data the variable sample rate is important: the section where we used the higher sample rate contains the slapping of the two subjects hands. This motion is rapid and cannot be accurately reconstructed with a sample rate of 30 frames per second. This variable sample rate presents problems for the autoregressive dynamics we reviewed in Section 2.1. However, for the regressive dynamics we introduced in Section 2.2 the variable sample rate can simply be reflected in the vector $\mathbf{t}$. We therefore made use of these dynamics by adding a further layer to the hierarchy,

$$\begin{aligned} p(\mathbf{Y}_1, \mathbf{Y}_2|\mathbf{t}) &= \int p(\mathbf{Y}_1|\mathbf{X}_1)\dots \\ &\quad \times \int p(\mathbf{Y}_2|\mathbf{X}_2) \int p(\mathbf{X}_1, \mathbf{X}_2|\mathbf{X}_3)\dots \\ &\quad \times p(\mathbf{X}_3|\mathbf{t})\,\mathrm{d}\mathbf{X}_3\mathrm{d}\mathbf{X}_2\mathrm{d}\mathbf{X}_1. \end{aligned}$$

[2] `http://mocap.cs.cmu.edu`.

[3] The subjects used are numbered 20 and 21 in the data base. The motion is number 11.

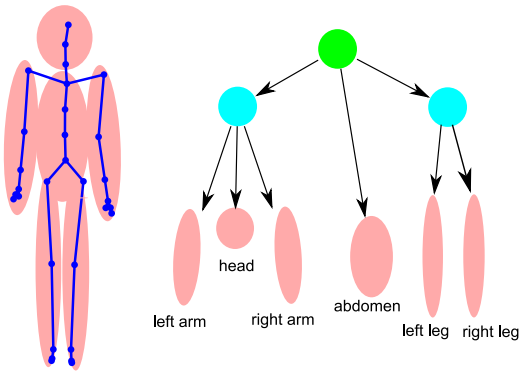

Figure 4. Decomposition of skeleton for hierarchical modelling. By separating the component parts of the skeleton in this manner we can model the space of natural motions for each component part and express them independently or jointly.

However, we do not optimise the parameters of the dynamics: we wish the latent space to be constrained by the dynamics. Finally, we would like the effect of the dynamics to be present as we descend the the hierarchy. To this end, we constrained the noise parameter, σ^2_{white} of the Gaussian process associated with $p(\mathbf{X}_1, \mathbf{X}_2|\mathbf{X}_3)$ to 1×10^{-6}. If we allow this variance to be free, the effect of the dynamics could become diluted as we drop down the hierarchy. By constraining this variance we force the temporal correlations present in the data to be respected.

In Figure 3 we show the results of mapping these motions into this hierarchical model.

4. Subject Decomposition

As well as decomposing the interactions between two subjects into a hierarchy, we can also consider decomposition of a single subject into parts. As we discussed in the introduction, there have been several different approaches to modelling motion capture data through tree based models, but these models typically assume that the nodes of the tree are observed and that the tree rigidly reflects the skeletal structure. Some effort has been made to model additional correlations in motion data by augmenting the tree with an additional, common, latent variable (Lan & Huttenlocher, 2005). However, our hierarchical model is closer in structure to the tree models of (Williams & Feng, 1999) where the tree structure refers to a *hierarchy of latent variables*, rather than a hierarchy of the observed variables.

We considered a data set composed of a walking motion and a running motion, again taken from the CMU MOCAP data base. The run was taken from motion 25 of subject 35 and the walk was taken from motion 01 of subject 35. The data was sub-sampled to 30 frames per second and one cycle of each motion was used. The x and y location of each motion's 'root position' was set to zero so that the subject was running/walking 'in place'. We modelled the subject using the decomposition shown in Figure 4, but to reflect the fact that two different motions were present in the data we constructed a hierarchy with *two roots*. One root was associated with the run and a second root was associated with the walk. The prior induced by the run root was applied only to the run data points in the next layer of the hierarchy (abdomen, legs, upper body). Similarly, the prior induced by the walk root was applied only to data points from the walk data. The upper body, legs and all the leaf nodes were applied to the entire data set. This construction enables us to express the two motion sequences separately whilst retaining the information required to jointly model the component parts of the skeleton. The aim is for nodes in the lower levels of the hierarchy to span the range of motions, whilst the upper layer specifies the particular motion type.

As the motion is broadly periodic, we made use of a periodic kernel (MacKay, 1998) for the regressive dynamics in each latent space (see pg. 92 in (Rasmussen & Williams, 2006) for details). The resulting visualisation is shown in Figure 5.

5. Discussion

We have presented a hierarchical version of the Gaussian process latent variable model. The Gaussian process latent variable model involves a paradigm shift in probabilistic latent variable models where, rather than marginalising the latent variables and optimising the mappings, we marginalise the mappings and optimise the latent variables. This makes far easier to construct hierarchies of these models. The philosophy of optimising versus marginalising is carried through to the hierarchical GP-LVM: we maximise with respect to all the latent variables in the different levels of the hierarchy.

5.1. Overfitting

Modelling with the GP-LVM is characterised by the use of very large numbers of 'parameters' in the form of the latent points. In the standard case, the number of parameters increases linearly as a fraction, $\frac{q}{d}$, of the number of data. As long as $q < d$ (how much less depends on the data set) problems of overfitting

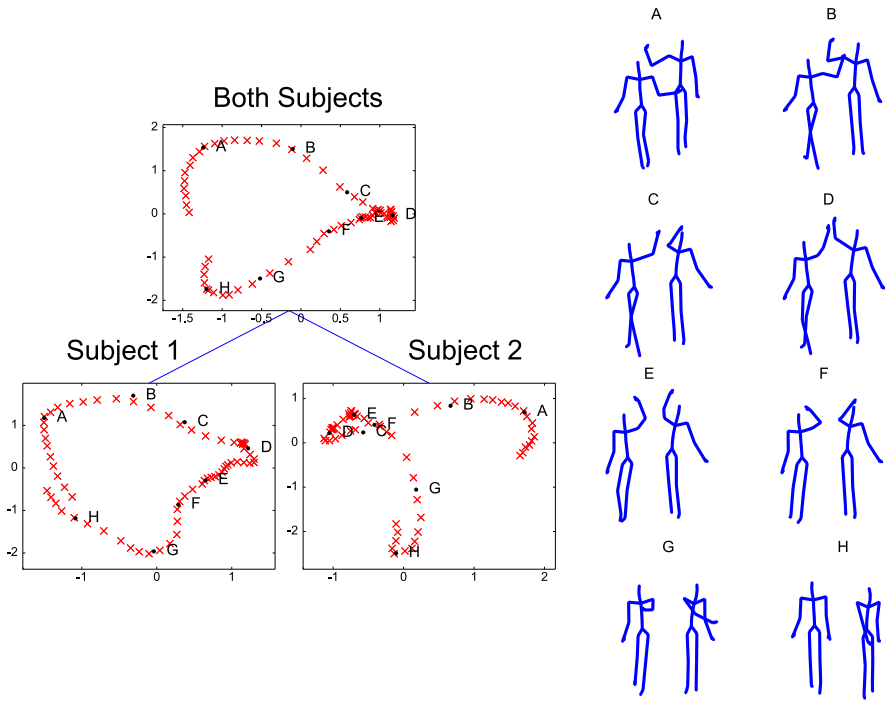

Figure 3. High Five example. Two subjects are modelled as they walk towards each other and 'high five'. The plot shows the simple hierarchy used to model the data. There is a regressive dynamical prior of the type described in Section 3 placed over the latent space of the root node. The root node then controls the two individual subjects. To illustrate the model we have taken points at time (i.e we input these values of t into the dynamical model) frames A: 85, B: 114, C:127, D: 141, E: 155, F: 170, G: 190 and H: 215. These points mapped down through the hierarchy and into the data space. In each of the plots of the two subjects, Subject 1 is on the right and Subject 2 is on the left.

do not normally occur. However, we are now adding additional latent variables, do we not now run the risk of overfitting the data if the hierarchy becomes too deep? The first point to note is that the upper levels of the hierarchy only serve to regularise the leaf nodes: so if the leaf nodes independently do not overfit, neither will the entire model. In other words, we must ensure that the leaf nodes each have $q_i < d_i$ where q_i is the number of columns of $\mathbf{X}_i$ and d_i is the dimensionality of $\mathbf{Y}_i$. However, by modifying the locations of latent variables in nodes higher up the hierarchy we are changing the nature of the *regularisation* of the leaf nodes. If unconstrained the model could simply act in such a way as to remove the regularisation. In our implementation we attempted to counter this potential problem in two ways. Firstly, we provided a fixed dynamical prior at the top level. The parameters of this prior were not optimised, so the top level node is always 'regularised[4]'. However, there is the possibility that this fixed regularisation could be 'diluted' by noise as we descend the hierarchy. To prevent this happening we constrained the noise variance of each Gaussian process that was not in a leaf node to 1×10^{-6}, *i.e.* close to zero but high enough to prevent numerical instabilities in kernel matrix inverses. This strategy proved effective in all our experiments.

5.2. Other Hierarchical Models

Given apparent similarities between the model names, it is natural to ask what is the relationship between the hierarchical GP-LVM and the hierarchical probabilistic PCA of (Bishop & Tipping, 1998)? The two models are philosophically distinct. In hierarchical PCA (and the related hierarchical GTM model of (Tino & Nabney, 2002)) every node in the hierarchy is associated with a probabilistic model in *data space*. The hierarchy is not a hierarchy of latent variables, it is, instead, a hierarchical clustering of mixture components in a discrete mixture of probabilistic PCA models (or GTM models). A similar approach could be taken with the GP-LVM, but it is not the approach we have described here.

[4]The same goal could also be achieved through *back constraints* (Lawrence & Quiñonero Candela, 2006), but we did not explore that approach here.

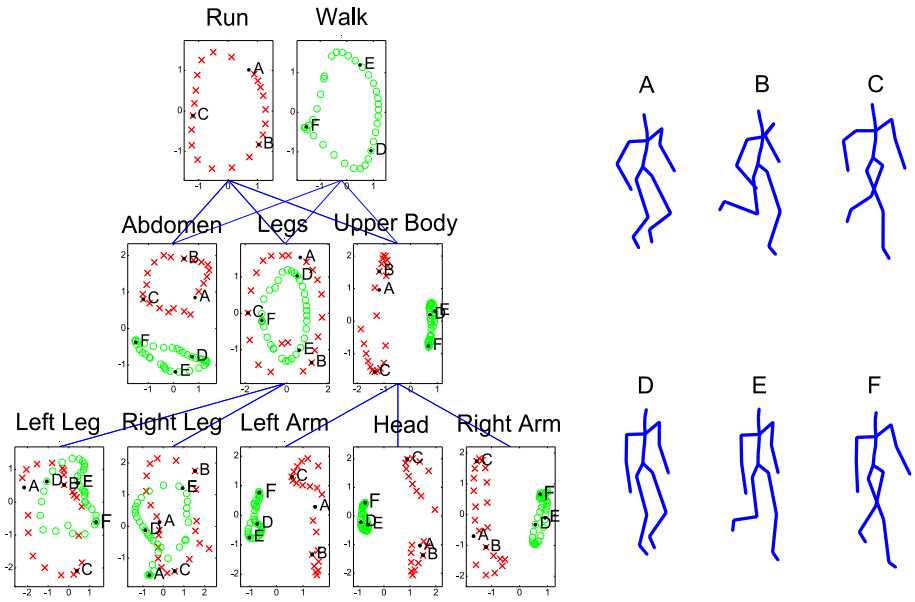

Figure 5. Combined model of a run and a walk. The skeleton is decomposed as shown in Figure 4. In the plots, crosses are latent positions associated with the run and circles are associated with the walk. We have mapped three points from each motion through the hierarchy. Periodic dynamics was used in the latent spaces.

5.3. Applications

We see the hierarchical GP-LVM as an important tool in several application areas. However, there are two application areas in which we believe the algorithm has particular promise. Firstly, the GP-LVM has already been proposed as a prior model for tracking. A key problem with constructing such prior models is that it is difficult to cover the space of all natural human motions. However, using the hierarchical model we expect, inspired by language modelling, to be able to perform a variant of 'back off'. Depending on motion, different models could be swapped in at the top level of the hierarchy, however some actions will still not be well modelled. In this case we suggest 'backing off', which in this context would translate into dropping down the hierarchy and applying the models in the next layer of the hierarchy *independently* to the data. Another application area where we see great promise for the model is animation. Through the hierarchical GP-LVM model different portions of the a character can be animated separately or jointly as circumstances demand. Animator time is becoming a dominating cost in both the games and film entertainment industries where computer special effect techniques are used, through combination of the hierarchical GP-LVM with appropriate inverse kinematic techniques (Grochow et al., 2004) we could seek to ameliorate these costs.

Acknowledgements

This work was funded by the EU FP6 PASCAL Network of Excellence under a pump priming grant. The motion capture data used in this project was obtained from `mocap.cs.cmu.edu`. The database was created with funding from NSF EIA-0196217. We thank Raquel Urtasun for helpful discussions.

A. Recreating the Experiments

The source code for re-running all the experiments detailed here is available from `http://www.cs.man.ac.uk/~neill/hgplvm/`, release 0.1.

References

Awasthi, P., Gagrani, A., & Ravindran, B. (2007). Image modelling using tree structured conditional random fields. *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)* (pp. 2060–2065).

Bishop, C. M., Svensén, M., & Williams, C. K. I. (1998). GTM: the Generative Topographic Mapping. *Neural Computation*, *10*, 215–234.

Bishop, C. M., & Tipping, M. E. (1998). A hierarchical latent variable model for data visualisation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *20*, 281–293.

Felzenszwalb, P. F., & Huttenlocher, D. P. (2000). Efficient matching of pictorial structures. *Proceedings of the Conference on Computer Vision and Pattern Recognition* (pp. 66–73). Hilton Head Island, South Carolina, U.S.A.: IEEE Computer Society Press.

Feng, X., Williams, C. K. I., & Felderhof, S. N. (2002). Combining belief networks and neural networks for scene segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*, 467–483.

Ferris, B. D., Fox, D., & Lawrence, N. D. (2007). WiFi-SLAM using Gaussian process latent variable models. *Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007)* (pp. 2480–2485).

Grochow, K., Martin, S. L., Hertzmann, A., & Popovic, Z. (2004). Style-based inverse kinematics. *ACM Transactions on Graphics (SIGGRAPH 2004)*.

Ioffe, S., & Forsyth, D. A. (2001). Mixtures of trees for object recognition. *Proceedings of the Conference on Computer Vision and Pattern Recognition* (pp. 180–185). Hawaii, U.S.A.: IEEE Computer Society Press.

Lan, X., & Huttenlocher, D. P. (2005). Beyond trees: Common-factor models for 2D human pose recovery. *IEEE International Conference on Computer Vision (ICCV)* (pp. 470–477). Bejing, China: IEEE Computer Society Press.

Lawrence, N. D. (2004). Gaussian process models for visualisation of high dimensional data. *Advances in Neural Information Processing Systems* (pp. 329–336). Cambridge, MA: MIT Press.

Lawrence, N. D. (2005). Probabilistic non-linear principal component analysis with Gaussian process latent variable models. *Journal of Machine Learning Research*, *6*, 1783–1816.

Lawrence, N. D., & Quiñonero Candela, J. (2006). Local distance preservation in the GP-LVM through back constraints. *Proceedings of the International Conference in Machine Learning* (pp. 513–520). Omnipress.

MacKay, D. J. C. (1995). Bayesian neural networks and density networks. *Nuclear Instruments and Methods in Physics Research, A*, *354*, 73–80.

MacKay, D. J. C. (1998). Introduction to Gaussian Processes. In C. M. Bishop (Ed.), *Neural networks and machine learning*, vol. 168 of *Series F: Computer and Systems Sciences*, 133–166. Berlin: Springer-Verlag.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems*. San Francisco: Morgan Kaufmann.

Ramanan, D., & Forsyth, D. A. (2003). Finding and tracking people from the bottom up. *Proceedings of the Conference on Computer Vision and Pattern Recognition* (pp. 467–474). Madison, Wisconsin, U.S.A.: IEEE Computer Society Press.

Rasmussen, C. E., & Williams, C. K. I. (2006). *Gaussian processes for machine learning*. Cambridge, MA: MIT Press.

Sigal, L., Bhatia, S., Roth, S., Black, M. J., & Isard, M. (2004). Tracking loose-limbed people. *Proceedings of the Conference on Computer Vision and Pattern Recognition* (pp. 421–428). Washington, DC, U.S.A.: IEEE Computer Society Press.

Tino, P., & Nabney, I. T. (2002). Hierarchical gtm: Constructing localized nonlinear projection manifolds in a principled way. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*, 639–656.

Tipping, M. E., & Bishop, C. M. (1999). Probabilistic principal component analysis. *Journal of the Royal Statistical Society, B*, *6*, 611–622.

Urtasun, R., Fleet, D. J., & Fua, P. (2006). 3D people tracking with Gaussian process dynamical models. *Proceedings of the Conference on Computer Vision and Pattern Recognition* (pp. 238–245). New York, U.S.A.: IEEE Computer Society Press.

Urtasun, R., Fleet, D. J., Hertzmann, A., & Fua, P. (2005). Priors for people tracking from small training sets. *IEEE International Conference on Computer Vision (ICCV)* (pp. 403–410). Bejing, China: IEEE Computer Society Press.

Wang, J. M., Fleet, D. J., & Hertzmann, A. (2006). Gaussian process dynamical models. *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press.

Williams, C. K. I., & Feng, X. (1999). Tree-structured belief networks as models of images. *Proceedings Ninth International Conference on Artificial Neural Networks, ICANN'99*.