

Temporal-Spatial Local Gaussian Process Experts with Vision Based Human Motion Tracking

¹Satyanarayana Mummana, ²T. Ravi Kiran

^{1,2}Dept. of CSE, Avanthi Institute of Engg. and Tech., Narsipatnam, Visakhapatnam, AP, India

Abstract

Human pose estimation via motion tracking systems can be considered as a regression problem within a discriminative framework. It is always a challenging task to model the mapping from observation space to state space because of the high dimensional characteristic in the multimodal conditional distribution. In order to build the mapping, existing techniques usually involve large set of training samples in the learning process which are limited in their capability to deal with multimodality. We propose, in this work, a novel online sparse Gaussian process regression model to recover 3-D human motion in monocular videos. Particularly, we investigate the fact that for a given test input, its output is mainly determined by the training samples potentially residing in its local neighborhood and defined in the unified input-output space.

Keywords

MAP, 3-D, Human Motion, Mapping, Modelling, Gaussian Process

I. Introduction

Vision based human motion tracking has been a functions, such as surveillance, rehabilitation, diagnostics, and human computer interaction. Among the large amount of studies in this field, the discriminative approach has been prevalent due to its feasibility of fast inference in real world scenarios and flexibility of adapting different learning methods the typical.

Statistical models for human motion are important for many applications in vision and graphics, notably visual tracking, activity recognition, and computer animation. It is well known in computer vision that the estimation of 3D human motion from a monocular video sequence is highly ambiguous. Many recently reported approaches have relied strongly on training prior models to constrain inference to plausible poses and motions. Specific activities could also be classified and recognized by evaluating the likelihood of the observation, given models for multiple activities. In computer animation, instead of having animators specify all Degrees Of Freedom (DOF) in a humanlike character, the task of animating characters can be simplified by finding the most likely motion, given sparse constraints. One common approach is to learn a probability distribution over the space of possible poses and motions, parameterized by the joint angles of the body, as well as its global position and orientation. Such a density function provides a natural measure of plausibility, assigning higher probabilities to motions that are similar to the training data. The task is challenging due to the high dimensionality of human pose data and to the complexity of the motion. However, poses from specific activities often linear an online Armani fold with much lower dimensionality than the number of joint angles. Motivated by this property, a common approach to define the generative model is to decouple the modeling of pose and motion. The motion is modeled by a dynamical process defined on a lower-dimensional latent space and the poses are generated by an observation process from the latent space. The current literature offers a number of generative models where the dynamics is not directly observed.

Simple models such as Hidden Markov Model (HMM) and Linear Dynamical Systems (LDS) are efficient and easily learned but limited in their expressiveness for complex motions.

More expressive models such as switching Linear Dynamical Systems (SLDS) and Nonlinear Dynamical Systems (NLDS), are more difficult to learn, requiring many parameters that need to be hand tuned and large amounts of training data.

In this paper, we investigate a Bayesian approach to learning NLDS, averaging over model parameters rather than estimating them. Inspired by the fact that averaging over nonlinear regression models leads to a Gaussian Process (GP) model, we show that integrating over NLDS parameters can also be performed in closed form. The resulting GPDynamical Model (GPDm) is fully defined by a set of low-dimensional representations of the training data, with both observation and dynamical processes learned from GP regression. As a natural consequence of the GP regression, the GPDm removes the need to select many parameters associated with function approximates while retaining the power of nonlinear dynamics and observation. Our approach is directly inspired by the GP Latent Variable Model (GPLVM). The GPLVM models the joint distribution of the observed data and their corresponding representation in a low-dimensional latent space. It is not, however, a dynamical model; rather, it assumes that data are generated independently, ignoring temporal structure of the input. Here, we augment the GPLVM with a latent dynamical model, which gives a closed-form expression for the joint distribution of the observed sequences and their latent space representations. The incorporation of dynamics not only enables predictions to be made about future data but also helps regularize the latent space for modeling temporal data in general. The unknowns in the GPDm consist of latent trajectories and hyper parameters. Generally, if the dynamical process defined by the latent trajectories is smooth, then the models tend to make good predictions. We first introduce a Maximum A Posteriori (MAP) algorithm for estimating all unknowns and discuss cases where it fails to learn smooth trajectories. Generally, if the dynamics process defined by the latent trajectories is smooth, then the models tend to make good predictions. To learn smoother models, we also consider three alternative learning algorithms, namely, the balanced GPDm (B-GPDm), manually specifying the hyper parameters, and a two-stage MAP approach. These algorithms present trade-offs in efficiency, synthesis quality, and ability to generalize. We compare learned models based on the visual quality of generated motion, the learned latent space configuration, and their performance in predicting missing frames of test data.

II. Dimensionality Reduction

Many tasks in statistics and machine learning suffer from the "curse of dimensionality". More specifically, the number of samples required to adequately cover a hyper volume increases exponentially with its dimension.

Performance in various algorithms, both in terms of speed and accuracy, is often improved by first obtaining a low dimensional representation of the data.

III. Linear Methods

A natural way to achieve dimensionality reduction is to represent the data in a linear subspace of the observation space. Probabilistic Principal Components Analysis (PPCA) and factor analysis provide both a basis for the subspace and a probability distribution in the observation space. They are straightforward to implement and efficient, and are often effective as a simple preprocessing step before the application of more complex modeling techniques. For purposes of density estimation, however, PCA is often unsuitable since many data sets are not well modeled by a Gaussian distribution. For instance, images of objects taken over the surface of the view sphere usually occupy a nonlinear manifold.

IV. Geometrically Motivated Manifold Learning

Nonlinear dimensionality reduction techniques allow one to represent data points based on their proximity to each other on nonlinear manifolds. Locally Linear Embedding (LLE) and the Laplacian Eigen map algorithm obtain the embedding by observing that all smooth manifolds are locally linear with respect to sufficiently small neighborhoods on the manifold. The Isomap algorithm and its variants C-Isomap, L-Isomap, and ST-Isomap extend multidimensional scaling by ensuring that the “dissimilarity” measure between pairs of data correspond to approximate geodesics on the manifold. In applications such as data visualization and analysis, it is often sufficient to recover a low-dimensional latent representation of the data without closed-form mappings between the latent space and observation space. Although manifold learning methods can be augmented with such mappings as a postprocess, they do not provide a probability distribution over data. Techniques such as mixtures of Gaussians or the Parzen window method can be used to learn a density model in the lower-dimensional space, but as observed in, with human pose data, estimation of mixture models is prone to over fitting and requires tuning a large number of parameters in practice. For LLE and Isomap, an additional problem is that they assume that the observed data are densely sampled on the manifold, which is typically not true for human motion data.

V. Nonlinear Latent Variable Models (NLVMs)

NLVMs are capable of modeling data generated from a nonlinear manifold. NLVM methods treat the latent coordinates and the nonlinear mapping to observations as parameters in a generative model, which are typically learned using optimization or the Monte Carlo simulation when needed. Compared to linear models such as PPCA, a lower number of dimensions can be used in the latent space without compromising reconstruction fidelity. The GPLVM is a generalization of the PPCA that allows for a nonlinear mapping from the latent space to the observation space. The model estimates the joint density of the data points and their latent coordinates. The estimates of the latent coordinates are used to represent a learned model and can be directly used for data visualization. The GPLVM has the attractive property of generalizing reasonably well from small data sets in high-dimensional observation spaces, and fast approximation algorithms for sparse GP regression can be used for learning. Except for ST-Isomap, neither manifold learning nor such NLVM methods are designed to model time series data. For applications in vision and graphics, the training data are typically video and motion capture sequences, where the frame-to-frame dependencies are important. Temporal models can also provide a predictive distribution over future data, which is important for tracking applications.

V. Linear Dynamical Systems

The simplest and most studied type of dynamical model is the discrete-time LDS, where the dynamical and observation functions are linear. The ML parameters can be computed iteratively using the EM algorithm. As part of the E-step, a Kalman smoother is used to infer the expected values of the hidden states. LDS parameters can also be estimated outside a probabilistic framework, and subspace state space system identification (4SID) methods identify the system in closed form but are suboptimal with respect to ML estimation. Although computations in LDS are efficient and are relatively easy to analyze, the model is not suitable for modeling complex systems such as human motion. By definition, nonlinear variations in the state space are treated as noise in an LDS model, resulting in overly smoothed motion during simulation. The linear observation function suffers from the same shortcomings as linear latent variable models,

VI. Nonlinear Dynamical Systems

A natural way of increasing the expressiveness of the model is to turn to nonlinear functions. For example, SLDSs augment LDS with switching states to introduce nonlinearity and appear to be better models for human motion. Nevertheless, determining the appropriate number of switching states is challenging, and such methods often require large amounts of training data, as they contain many parameters. The task of motion synthesis is often to make progress toward a goal state. Since this behavior is naturally exhibited by differential equations with well-defined attractor states or limit cycles, faster learning and more robust dynamics can be achieved by simply parameterizing the dynamical model as a differential equation. The dynamics and observation functions can be modeled directly using nonlinear basis functions. Roweis and Ghahramani use Radial Basis Functions (RBFs) to model the nonlinear functions and identify the system by using an approximate EM algorithm. The distribution over hidden states cannot be estimated exactly due to the nonlinearity of the system. Instead, extended Kalman filtering, which approximates the system by using locally linear mappings around the current state, is used in the E-step. In general, a central difficulty in modeling time series data is in determining a model that can capture the nonlinearities of the data without over fitting. Linear autoregressive models require relatively few parameters and allow closed-form analysis but can only model a limited range of systems. In contrast, existing nonlinear models can model complex dynamics but usually require many training data points to accurately learn models.

VII. Applications

Our work is motivated by human motion modeling for video-based people tracking and data-driven animation. People tracking requires dynamical models in the form of transition densities in order to specify predictive distributions over new poses at each time instant. Similarly, data driven computer animation can benefit from prior distributions over poses and motion.

VIII. Monocular Human Tracking

Despite the difficulties with linear subspace models mentioned above, PCA has been applied to video-based people tracking of humans and other vision applications. To this end, the typical data representation is the concatenation of the entire trajectory of poses to form a single vector in observation space. The lower-dimensional PCA subspace is then used as the state space. In place of explicit dynamics, a phase parameter, which propagates forward in time, can serve as an index to the prior distribution

of poses. Nonlinear dimensionality reduction techniques such as LLE have also been used in the context of human pose analysis. Elgammal and Lee use LLE to learn activity based manifolds from silhouette data. They then use nonlinear regression methods to learn mappings from manifolds back to the silhouette space and to the pose space. Jenkins and Mataric use ST-Isomap to learn embeddings of multiactivity human motion data and robot teleoperation data. Smichisescu and Jepson used spectral embedding techniques to learn an embedding of human motion capture data. They also learn a mapping back to pose space separately. None of the above approaches learns a dynamical function explicitly and no density model is learned in. In general, learning the embedding, the mappings, and the density function separately is undesirable.

IX. Computer Animation

The applications of probabilistic models for animation revolve around motion synthesis, subject to sparse user constraints. Brand and Hertzmann augment an HMM with stylistic parameters for style-content separation. Li et al. model human motion by using a two-level statistical model, combining linear dynamics and Markov switching dynamics. A GPLVM is applied to inverse kinematics by Grochow et al., where ML is used to determine pose, given kinematics constraints. Nonparametric methods have also been used for motion prediction and animation. For example, in animation with motion graphs, each frame of motion is treated as a node in the graph. A similarity measure is assigned to edges in the graph and can be viewed as transition probabilities in a first-order Markov process. Motion graphs are designed to be used with large motion capture databases, and the synthesis of new motions typically amounts to reordering the poses already in the database. An important strength of motion graphs is the ability to synthesis high-quality motions, but the need for a large amount of data is undesirable. Motion interpolation techniques are designed to create natural-looking motions relatively far from input examples. Typically, a set of interpolation parameters must be either well-defined (that is, the location of the right hand) or specified by hand (that is, a number representing emotion) for each example. A mapping from the parameter space to the pose or motion space is then learned using nonlinear regression. Linear interpolation between motion segments by using the spatial-temporal morphable models is possible, provided that correspondences can be established between the available segments. More closely related to our work, Mukai and Kuriyama employ a form of GP regression to learn the mapping from interpolation parameters to pose and motion. In particular, one can view the GPLVM and the GPDM introduced below as interpolation methods with learned interpolation parameters.

X. Gaussian Process Dynamics

The GPDM is a latent variable model. It comprises a generative mapping from a latent space x to the observation space y and a dynamical model in the latent space. These mappings are, in general, nonlinear. For human motion modeling, a victory in the observation space corresponds to a pose configuration, and a sequence of poses defines a motion trajectory. The latent dynamical model accounts for the temporal dependence between poses.

XI. Evaluation of Learned Models

The computational bottleneck for the learning algorithms above is the inversion of the kernel matrices, which is necessary to evaluate the likelihood function and its gradient. Learning by using

the MAP estimation, the B-GPDM, and fixed hyper parameters requires approximately 6,000 inversions of the kernel matrices, given our specified number of iterations. These algorithms take approximately 500 seconds for a data set of 289 frames. The two-stage MAP algorithm is more expensive to run, as both the generation of samples in the E-step and the averaging of samples in the M-step require evaluation of the likelihood function. The experiments below used approximately 400,000 inversions, taking about 9 hours for the same data set of 289 frames.

The potential applications of human motion capture are the driving force of system development, and we consider the following three major application areas: surveillance, control, and analysis. The surveillance area covers applications where one or more subjects are being tracked over time and possibly monitored for special actions. A classic example is the surveillance of a parking lot, where a system tracks subjects to evaluate whether they may be about to commit a crime, e.g., steal a car. The control area relates to applications where the captured motion is used to provide controlling functionalities. It could be used as an interface to games, virtual environments, or animation or to control remotely located implements. For a comprehensive discussion of motion capture in the control application area. The third application area is concerned with the detailed analysis of the captured motion data. This may be used in clinical studies of, e.g., diagnostics of orthopedic patients or to help athletes understand and improve their performance.

XII. Alternative Technologies for Motion Capture

The systems used to capture human motion consist of subsystems for sensing and processing, respectively. The operational complexity of these subsystems is typically related, so that high complexity of one of them allows for a corresponding simplicity of the other. This trade-off between the complexities also relates to the use of active versus passive sensing. Active sensing operates by placing devices on the subject and in the surroundings which transmit or receive generated signals, respectively. Active sensing allows for simpler processing and is widely used when the applications are situated in well-controlled environments. That is in particular the case for the third application area, analysis, and in some of the control applications. Passive sensing is based on "natural" signal sources, e.g., visual light or other electromagnetic wavelengths, and requires no wearable devices. An exception is when markers are attached to the subject to ease the motion capture process. Markers are not as intrusive as the devices used in active sensing. Passive sensing is mainly used in surveillance and some control applications where mounting devices on the subject is not an option. Computer vision with the passive sensing approach has challenged active sensing within all three application areas. Even though the use of markers may seem a good compromise

XIII. Conclusions

The human motion tracking has been a fundamental open problem is verified with pervasive real works application, such as surveillance, rehabilitating, diagnostics and human computer interaction. Among the large amount of studies in this field. The discriminative approach has been prevalent due to its feasibility of fast inference in real world scenarios and flexibility

References

- [1] N.D.Lawrence, M.Seegar, R.Herbrich, "Fast sparse Gaussian process methods: the informative vector machine".
- [2] C.E.Rasmussen, Z.Ghahranai, "Infinite mixtures of Gaussian process experts".
- [3] Lacy, J., Mitchell, D., Schell, W., "CryptoLib: Cryptography in Software", Proc. Fourth USENIX Security Workshop, October 1993.
- [4] Ioannidis, J., Blaze, M., "Architecture and Implementation of Network-Layer Security Under Unix", Proc. Fourth USENIX Security Workshop, October 1993.
- [5] National Bureau of Standards, "Data Encryption Standard", FIPS Publication #46, NTIS, April 1977.
- [6] A. Baggio, "Wireless sensor networks in precision agriculture", in ACM Workshop on Real-World Wireless Sensor Networks (REALWSN 2005), Stockholm, Sweden, June 2005.
- [7] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, J. Anderson, "Wireless Sensor Networks for Habitat Monitoring", in First ACM Workshop on Wireless Sensor Networks and Applications, Atlanta, GA, USA, September 2002.
- [8] G. Fuchs, S. Truchat, F. Dressler, "Distributed Software Management in Sensor Networks using Profiling Techniques", in 1st IEEE/ACM International Conference on Communication System Software and Middleware (IEEE COMSWARE 2006): 1st International Workshop on Software for Sensor Networks (SensorWare 2006), New Dehli, India, January 2006.



Satyanarayana Mummana is working as an Asst. Professor in Avanathi Institute of Engineering & Technology, Visakhapatnam, Andhra Pradesh. He has received his Masters degree (MCA) from Gandhi Institute of Technology and Management (GITAM), Visakhapatnam and M.Tech (CSE) from Avanathi Institute of Engineering & Technology, Visakhapatnam. Andhra Pradesh. His research areas include Image Processing, Computer Networks, Data Mining,

Distributed Systems, Cloud Computing.



T. Ravi Kiran received his B.Tech degree from GMR Institute of Technology, Rajam, AP. He is currently pursuing M.Tech degree from Avanathi Institute of Engg. & Tech., Visakhapatnam, Andhra Pradesh. His research interests include Image Processing, Web Technologies, Search Engines, Data Privacy, Web Search, Network Security, Distributed Systems.