Human Motion Tracking by Temporal-Spatial Local Gaussian Process Experts

Xu Zhao, *Member, IEEE*, Yun Fu, *Senior Member, IEEE*, and Yuncai Liu, *Member, IEEE*

***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 a 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 (GP) 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. This leads to a local mixture GP experts system composed of different local GP experts, each of which dominates a mapping behavior with the specific covariance function adapting to a local region. To handle the multimodality, we combine both temporal and spatial information therefore to obtain two categories of local experts. The temporal and spatial experts are integrated into a seamless hybrid system, which is automatically self-initialized and robust for visual tracking of nonlinear human motion. Learning and inference are extremely efficient as all the local experts are defined online within very small neighborhoods. Extensive experiments on two real-world databases, HumanEva and PEAR, demonstrate the effectiveness of our proposed model, which significantly improve the performance of existing models.**

***Index Terms*—Gaussian process regression, human motion tracking, local experts model, pose estimation, temporal-spatial model.**

I. Introduction

VISION BASED human motion tracking has been a fundamental open problem, with pervasive real-world applications [1], such as surveillance, rehabilitation, diagnostics, and human computer interaction. Among the large amount of studies in this field, the discriminative approach [2] has been prevalent due to its feasibility of fast inference in real-world scenarios and flexibility of adapting to different learning methods. The typical objective of these approaches [3]–[8] is to model the direct mapping from visual observations to well-defined human pose configurations. The wide spectrum of such methods ranges from nearest-neighbor retrieval [9], [10] and manifold learning [4] to regression [7], [11] and probabilistic mixture of predictors [2], [5].

Suffering from the intrinsic visual-to-pose ambiguity, however, all the discriminative approaches have the same difficulty of effectively modelling multimodal conditional distributions with small-size training data in a high-dimensional space. The category of model mixtures is the most common technique to handle multimodality. The conditional Bayesian Mixture of Experts (BME) model [2], [5] has been effective in representing the multimodal visual-to-pose mapping, by introducing the input sensitive gate function. However, most parametric models are usually not robust in dealing with high-dimensional data. In addition, the BME model may degrade on small-size training data since its performance heavily depends on the data distribution in ambiguous regions.

Gaussian Process (GP) [12] and its variants, within both discriminative [7], [8], [13] and generative [14] frameworks, have been applied to human pose/motion estimation in a few recent works. Particularly, GP regression model has proven to be a powerful approach. It defines a prior probability distribution over infinite function spaces, which leads to a nonlinear probabilistic regression framework working along with the kernelized covariance function. The flexibilities in kernel selection and non-parametric nature of GP model are advantageous to find efficient solutions of pose/motion estimation on small-scale databases [7], [8], [14]. Within the discriminative framework, human motion estimation is mainly built on the basis of GP regression. However, the full GP regression suffers from two inevitable limitations: 1) relatively expensive computational cost and 2) insufficient capability to handle multimodality [15].

The sparse approximation of full GP [12], [16] has been investigated to relieve the computational difficulty, which typically only use a subset of training inputs [17] or a set of inducing variables [18] to approximate the covariance matrix. Without losing any key characteristics, such models still work within the global voting framework even if the computational expenses can be relatively reduced through such approximations. Therefore, for each test input, all the training samples are involved in the inference process. It might be lacking of effective mechanisms to avoid the averaging effect.

Mixture of Gaussian process experts [19]–[21] is an alternative approach to mitigate the two limitations. As same as mixture of experts architecture [22], the input space of this model is divided into different regions by a gating network, each of which is dominated by a specific GP expert. In the model, the cubic computing cost on the entire dataset is reduced to that on only part of the data. At the same time, the covariance functions

Manuscript received November 12, 2009; revised April 07, 2010; accepted August 19, 2010. Date of publication September 16, 2010; date of current version March 18, 2011. This work was supported in part by the National Basic Research Program (973 Program) of China (No. 2011CB302203), the Key Program of National Natural Science Foundation of China (No. 60833009) and the SUNY Buffalo Faculty Startup Funding. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Patrick J. Flynn.

X. Zhao and Y. Liu are with the School of Electronic, Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: zhaoxu@sjtu.edu.cn; whomliu@sjtu.edu.cn).

Y. Fu is with the Department of Computer Science and Engineering, State University of New York (SUNY) at Buffalo, Buffalo, NY 14260-2000, USA (e-mail: yunfu@buffalo.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2010.2076820

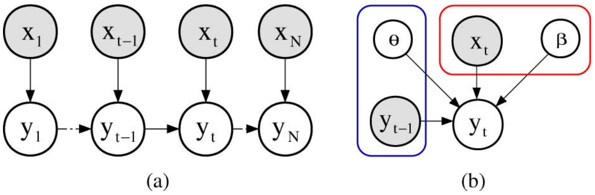


Fig. 1. Graphical model description of our proposed framework. (a) The discriminative framework in the temporal chain. $\mathbf{x}$ represents input and $\mathbf{y}$ output. Shaded nodes indicate the observation and unshaded nodes indicate modeled variables. (b) Detailed description of the graphical model of one time node in the temporal chain. The rectangle enclosed blocks in the left and right side represent temporal and spatial experts respectively. θ and β are the learned hyperparameters.

are localized to adapt to different regions accordingly. However, learning the mixture GP experts is usually intimately coupled with the determining of gating network. Training the gating network itself is often a nontrivial problem.

We propose a novel mixtures of local GP experts model in this work, which incorporates both temporal and spatial information. Theoretically, it is insufficient to effectively handle multimodality only by spatial information since the problem of monocular human motion estimation itself is ill-posed. Introducing temporal information into the model is reasonably necessary. But existing discriminative methods are short of temporal estimation framework. One exception is the parametric model proposed in [2], in which temporal smoothness constraints are added into the BME model. It is also worth noting that in [23], the Gaussian Process Dynamical Model (GPDM) [24] is used to model the dynamics of human motions. As the original GPDM [24] is designed to find a low-dimensional latent space with associated dynamics, it is introduced to capture the motion priors in the latent state space by [23]. Although both our proposed method and the GPDM based method utilize temporal information within Gaussian Process context, actually they work in different frameworks. In [23], after learning the motion prior in the state space, the pose estimation process falls into generative framework by optimizing a likelihood function. However our model is in a regression framework, which is discriminative. In addition, our model is local but GPDM is global.

Inspired by the existing work on human pose inference by sparse GP regression [13], our model, as a discriminative approach, is non-parametric and temporally-spatially integrated. In the existing model, GP experts are trained offline and the local GP regressors are defined online for each test sample. Derived from the test sample neighborhood in the appearance space, each local GP is defined to be consistent in the pose space. This model can avoid the tedious efforts introduced by the mixture of GP experts in computing the gating network. By generalizing the localization strategy of [13], we propose a local GP experts model. The graphical model description of our approach is shown in Fig. 1. In summary, the main contributions of this paper are in four aspects:

1) We propose to define the local GP experts in the unified input-output space, therefore each GP expert is composed of samples that are localized in both input and output spaces. This strategy is different from that proposed in [13], in which the neighborhood is defined separately in input and output spaces. Such scheme is prone to fail in dealing with more-to-one mappings because the neighborhood relationship in the output space may be changed in the input space. In comparison, our model can flexibly handle the two-way multimodality.
2) We build local GP experts model in the temporal chain therefore get the temporal experts. In the unified space, we integrate the temporal and spatial experts into a hybrid system to make prediction and handle multimodality. Basically, human motion has its dynamic behavior. In the state space, the configuration of human pose moves along a special manifold [4]. Using temporal information can alleviate the multimodality and explore the underlying context of the output space. In our model, the temporal experts are trained offline, so once the temporal information is unavailable midway, we can easily switch to spatial local GP experts alone for the prediction.
3) We collect the new Pose Estimation and Action Recognition (PEAR) database to facilitate the research on human pose estimation, motion tracking and action recognition.
4) We evaluate the proposed Temporal-Spatial Local (TSL) GP model on two real databases, HumanEva [25] and PEAR, and achieve significant improvements against both the full GP model and the local sparse GP model.

This work is an extension of our previous research in [15]. Comparing to [15], this work involves more technical details and experiments conducted on both HumanEva and PEAR databases. We also add more experimental studies on the selection of algorithm parameters. The structure of the paper is organized as follows. In Section II, we introduce the local GP experts model and its implementation. In Section III, we describe how the temporal information is integrated into the local GP model. We report the experiments on both HumanEva and PEAR databases in Section IV. The results of extensive comparative experiments are analyzed and discussed. Finally, we conclude the proposed work and envision future research directions in Section V.

II. Local Gaussian Process Experts Model

In this section, we present the sparse strategy of GP regression in the unified input-output space, which leads to our proposed local GP experts model. We review the GP regression in Section II-A and then present the detailed algorithm for our model in Section II-B.

A. *Gaussian Process Regression Revisited*

Gaussian process is the generalization of Gaussian distributions defined over infinite index sets [12]. Suppose we have a training dataset $\mathcal{D} = \{(\mathbf{x}_i, \mathbf{y}_i), i = 1, \ldots, N\}$, composed of inputs $\mathbf{x}_i$ and noisy outputs $\mathbf{y}_i$. We consider a regression model defined in terms of the function $f(\mathbf{x})$ so that

$$\mathbf{y}_i = f(\mathbf{x}_i) + \epsilon_i \quad (1)$$

where $\epsilon_i \sim \mathcal{N}(0, \beta^{-1})$ is a random noise variable and the hyperparameter β represents the precision of the noise. From the Gaussian assumption of prior distribution over functions $f(\mathbf{x})$, the joint distribution of outputs $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^T$ conditioned on input values $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T$ is given by

$$p(\mathbf{Y}|\mathbf{X}) = \int p(\mathbf{Y}|\mathbf{f}, \mathbf{X}) p(\mathbf{f}|\mathbf{X}) d\mathbf{f} = \mathcal{N}(\mathbf{Y}|0, \mathbf{K}) \quad (2)$$

where $\mathbf{f} = [f_1, \ldots, f_N]^T$, $f_i = f(\mathbf{x}_i)$ and the covariance matrix $\mathbf{K}$ has elements

$$K_{i,j} = k(\mathbf{x}_i, \mathbf{x}_j) + \beta^{-1}\delta_{ij} \tag{3}$$

where δ_{ij} is the Kronecker delta function. In this paper, we use a kernel function k which is the sum of an isotropic exponential covariance function, a noise term and a bias term, all with hyperparameters, $\bar{\theta}$. During the training, the hyperparameters $\bar{\theta}$ are learnt by minimizing

$$-\ln p(\mathbf{Y}|\mathbf{X}, \bar{\theta}) = \frac{D}{2}\ln|\mathbf{K}| + \frac{1}{2}\mathbf{Y}^T\mathbf{K}^{-1}\mathbf{Y} + \frac{N}{2}\ln(2\pi) \tag{4}$$

where D is the dimension of the output space. For a new test input $\mathbf{x}_*$, the conditional distribution, $p(\mathbf{y}_*|\mathbf{X}, \mathbf{Y}, \mathbf{x}_*) = \mathcal{N}(\mu, \sigma)$, is a Gaussian distribution with mean and covariance given by

$$\mu(\mathbf{x}_*) = \mathbf{k}_{*,\zeta}\mathbf{K}_{\zeta,\zeta}^{-1}\mathbf{Y}_\zeta \tag{5}$$

$$\sigma(\mathbf{x}_*) = k_{*,*} - \mathbf{k}_{*,\zeta}\mathbf{K}_{\zeta,\zeta}^{-1}\mathbf{k}_{\zeta,*} \tag{6}$$

where ζ's are the indexes of the N training inputs, $\mathbf{K}_{\zeta,\zeta}$ is the covariance matrix with elements given by (3) for $i, j = 1, \ldots, N$, vector $\mathbf{k}_{\zeta,*} = \mathbf{k}_{*,\zeta}^T$ is the cross-covariance of the test input and the N training inputs, and scalar $k_{*,*} = k(\mathbf{x}_*, \mathbf{x}_*) + \beta^{-1}$ is the covariance of the test input.

Note that the mean (5) of the prediction distribution can be written as a function of $\mathbf{x}_*$, in the form

$$\mu(\mathbf{x}_*) = \sum_{n=1}^{N} a_n k(\mathbf{x}_n, \mathbf{x}_*) \tag{7}$$

where a_n is the nth component of $\mathbf{K}_{\zeta,\zeta}^{-1}\mathbf{Y}_\zeta$. In this view, $\mu(\mathbf{x}_*)$ is determined by the linear combination of N kernel functions, with each one centered on a training point. From another viewpoint, the mean prediction (5) is actually a weighted voting from N training outputs

$$\mu(\mathbf{x}_*) = \sum_{n=1}^{N} w_n \mathbf{y}_n \tag{8}$$

where w_n is the nth component of $\mathbf{k}_{*,\zeta}\mathbf{K}_{\zeta,\zeta}^{-1}$. With this insight, we can view the GP regression as a voting process, where each training output has a weighted vote to determine what the test output should be.

In the full GP regression model, all the training data are involved in the voting process for determining the corresponding outputs of the test inputs regardless of the data distribution in the local domain. This global voting system is computationally prohibitive and can lead to biased predictions when the conditional distribution is multimodal. We next introduce a novel local voting mechanism to not only localize the full GP model to adapt the multimodality effectively, but also reduce the computational cost for feasible online inference.

B. Local Mixture of GP Experts

In order to reduce the computing cost and handle multimodality, we have to sparsify the full GP regression model. Current GP sparse techniques [12], [16] mainly focus on globally sparsifying the full training dataset based on some selection criteria such as online learning [26], greedy posterior maximization [27], maximum information gain [28], and matching pursuit [29]. By using this kind of methods, the computational complexity of full GP, $\mathcal{O}(N^3)$, is reduced to $\mathcal{O}(m^3)$ or $\mathcal{O}(Nm^2)$, where N and m are the sizes of the full training dataset and the selected subset respectively. However, for very large database, the reduction is not enough. Moreover, these ideas still work within the global voting framework. It means that for every test input, no matter which local distribution mode they belong to, the selection of the training samples and covariance function are global.

As a non-parametric model, the performance of GP regression is closely related to the kernel function and the examples involved in the computation of covariance matrix $\mathbf{K}$. For a special test input, the training samples within its neighborhood usually have more impacts on the prediction than those far from it. For example, when using monotonically decreasing covariance functions, the covariance matrix is sparse; $K_{i,j}$ is very small for all the entries where the distance between $\mathbf{x}_i$ and $\mathbf{x}_j$ is large. As for the voting, the weights of the local voters are bigger than others (see (8)). In the GP model, kernel function provides a metric to measure the similarity between the inputs. Ideally, this metric should be adjusted dynamically to adapt to different local regions.

We develop the local mixture of GP experts through the above motivation. Similar to the model in [13], for a given test input, we select different local GP experts in its neighborhood. The training samples of each expert are also selected locally. These local experts build up a local mixture GP experts system to make the prediction. Therefore, our model formulates the mean prediction for a given test input $\mathbf{x}_*$ by

$$\mu(\mathbf{x}_*) = \sum_{i=1}^{T} \pi_i \mathbf{k}_{*,\zeta_i}\mathbf{K}_{\zeta_i,\zeta_i}^{-1}\mathbf{Y}_{\zeta_i} = \sum_{i=1}^{T}\sum_{j=1}^{S} \pi_i w_{ij}\mathbf{y}_{ij} \tag{9}$$

where T is the number of local experts, S the size of each expert, ζ_i the index set of samples for the ith expert, π_i the prediction weight of the ith expert, $\mathbf{y}_{ij}$ the jth training output belonging to the ith expert and w_{ij} is its weight. Both T and S are parameters of our model, and practically small values are sufficient to generate satisfactory predictions.

Definition of Local Neighborhood: Different from the localization strategy in [13], our model defines the neighborhood in the input-output unified space $\mathbb{U}$, where the data points are the concatenation of input and output vectors. The advantages of our strategy are twofold [15]:

- The neighborhood relationship is closer to the real distribution in $\mathbb{U}$ than in the single input and output space. For example in pose estimation, two image feature points which are very similar in the feature space might be quite different in the pose space, and vice versa. In $\mathbb{U}$, this kind of ambiguity can be avoided to a large extent.
- Our strategy can deal with two-way multimodal distributions. For the more-to-one (input-to-output) mapping, neighborhood relationship defined in output space may not be kept. The data points can be scattered in the input space by using the neighborhood definition in the output space. But in $\mathbb{U}$, this situation can be avoided.

In implementation, the unified data space $\mathbb{U}$ is divided into R different local regions with a clustering algorithm. Each region is dominated by a local GP expert trained offline. Given a test input, starting from its neighborhood in the input space, we find its local neighbors in $\mathbb{U}$ to build the local mixture of GP experts model.

1) Determination of Prediction Weights: In (9), the prediction weights π_i of the local experts are determined by the probabilities of the local experts given a test input $\mathbf{x}_*$. The mean prediction thereby can be expressed as $\mu(\mathbf{x}_*) = \mathbb{E}\{\mu_j|\mathbf{x}_*\} = \sum_{j=1}^{T} \mu_j p(\mathcal{M}_j|\mathbf{x}_*)$, where $\mathcal{M}_j$ is the local expert and μ_j the local prediction [30]. According to the Bayesian theorem and by taking $p(\mathcal{M}_j, \mathbf{x}_*)$ as the distance measure of $(\mathbf{x}_*, \mu_j)$ to the center of the local expert $\mathcal{M}_j$, we have

$$p(\mathcal{M}_j|\mathbf{x}_*) = \frac{p(\mathcal{M}_j, \mathbf{x}_*)}{\sum\limits_{j=1}^{T} p(\mathcal{M}_j, \mathbf{x}_*)} = \frac{w_j}{\sum\limits_{j=1}^{T} w_j} \tag{10}$$

where w_j is calculated by the kernel function defined in (3). Therefore, we can get the mean prediction

$$\mu(\mathbf{x}_*) = \sum_{j=1}^{T} \left(\frac{w_j}{\sum\limits_{j=1}^{T} w_j} \right) \mu_j. \tag{11}$$

Actually the prediction weight $\pi_j = w_j / \sum_{j=1}^{T} w_j$ can be interpreted as a normalized distance from $(\mathbf{x}_*, \mu_j)$ to the center of the local expert. The algorithm is summarized in Algorithm 1, where the dataset in $\mathbb{U}$ is represented as $\mathbf{D} = [\mathbf{d}_1, \ldots, \mathbf{d}_N]$ with $\mathbf{d}_i = (\mathbf{x}_i, \mathbf{y}_i)$. The function $\mathrm{findNN}(\mathbf{X}, \mathbf{x}, S)$ finds S nearest neighbors of $\mathbf{x}$ in $\mathbf{X}$. The function $\mathrm{kmeans}(\mathbf{D}, R)$ performs k-means clustering on dataset $\mathbf{D}$ and returns the R centers $\mathbf{C}_{\mathcal{R}}$ and clusters $\mathbf{D}_{\mathcal{R}}$.

Algorithm 1 Local mixture of GP experts: learning and inference

```
OFFLINE: Training of the Local Experts
$R$: number of local GP experts $(\mathbf{C}_{\mathcal{R}}, \mathbf{D}_{\mathcal{R}}) = \mathrm{kmeans}(\mathbf{D}, R)$
for $i = 1 \ldots R$ do
    $\{\bar{\theta}^i\} \Leftarrow \min\left(-\ln p\left(\mathbf{Y}_{\mathcal{R}_i}|\mathbf{X}_{\mathcal{R}_i}, \bar{\theta}^i\right)\right)$
end for
ONLINE: Inference of test point $\mathbf{x}_*$
$T$: number of experts, $S$: size of each expert
$\eta = \mathrm{findNN}(\mathbf{X}, \mathbf{x}_*, T)$
for $j = 1 \ldots T$ do
    $\zeta = \mathrm{findNN}(\mathbf{D}, \mathbf{d}_{\eta_j}, S)$
    $t = \mathrm{findNN}(\mathbf{C}_{\mathcal{R}}, \mathbf{d}_{\eta_j}, 1)$
    $\bar{\theta} = \bar{\theta}^t$
    $\mu_j = \mathbf{k}_{*,\zeta}\mathbf{K}_{\zeta,\zeta}^{-1}\mathbf{Y}_{\zeta}$
    $\sigma_j = k_{*,*} - \mathbf{k}_{*,\zeta}\mathbf{K}_{\zeta,\zeta}^{-1}\mathbf{k}_{\zeta,*}$
end for
$p(\mathbf{y}_*|\mathbf{X}, \mathbf{Y}) \approx \sum_{i=1}^{T} \pi_i \mathcal{N}\left(\mu_i, \sigma_i^2\right)$
```

In the framework of our local GP model, a full GP with stationary covariance function is approximated by the local GP experts centered at the neighbors of the given test point. It reduces considerably the computational cost and allows learning and inference with extremely large database. However, the capability of dealing with multimodality in this way depends on the distribution of training data in the multimodal region. If different modes distribute equally in the region, the prediction may also suffer from the averaging effect like full GP. Therefore, theoretically, it is insufficient to handle multimodality accurately by only using the spatial information (See the experiments in Section IV). We next present a more sophisticated mixture of local GP experts by incorporating temporal experts into the model.

III. Temporal-Spatial Local GP Experts

Based on the spatial experts, we introduce the temporal experts as an extension to handle multimodality more effectively. In the temporal-spatial combined GP experts model, the spatial local experts learn the relationship between the input space and output space, while the temporal local experts explore the underlying context of the output space. In the scenario of sequential data, by adding the temporal constraint, the regression models can be formulated as

$$\mathbf{y}_t = f(\mathbf{x}_t) + \epsilon_{x,t} \tag{12}$$

$$\mathbf{y}_t = g(\mathbf{y}_{t-1}) + \epsilon_{y,t} \tag{13}$$

where t is the temporal tag, $\epsilon_{x,t} \sim \mathcal{N}\left(0, \beta_x^{-1}\right)$ and $\epsilon_{y,t} \sim \mathcal{N}\left(0, \beta_y^{-1}\right)$ are noise processes. We use the first-order Markov dynamical model to represent the dependence in the output space. For (13), considering dynamic mapping on dataset $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^T$ in the output space, the joint distribution of $\mathbf{Y}$ is given by

$$p(\mathbf{Y}) = p(\mathbf{y}_1) \int \prod_{t=2}^{N} p(\mathbf{y}_t|\mathbf{y}_{t-1}, \mathbf{g}) p(\mathbf{g}) d\mathbf{g} \tag{14}$$

where $\mathbf{g} = [g_1, \ldots, g_{N-1}]^T$ and $g_i = g(\mathbf{y}_i)$. With a Gaussian prior over g, we can obtain the similar log likelihood function as (4) and learn the hyperparameters for the temporal model. Considering the nonlinear dynamical nature of human motion, we use an RBF plus linear kernel

$$k(\mathbf{y}_i, \mathbf{y}_j) = \theta_0 \exp\left\{-\frac{\theta_1}{2} \parallel \mathbf{y}_i - \mathbf{y}_j \parallel^2\right\} + \theta_2 + \theta_3 \mathbf{y}_i^T \mathbf{y}_j. \tag{15}$$

We use the similar localization strategy described in Algorithm 1 to build the local temporal experts model. Once the local temporal experts generate the prediction $\hat{\mathbf{y}}$, we proceed to make the prediction supported by the local spatial experts in the unified space $\mathbb{U}$. Therefore, this process is described by

$$p(y_t|y_{t-1}, x_t) = \int p(y_t|\hat{y}_t, x_t) p(\hat{y}_t|y_{t-1}) d\hat{y}_t. \tag{16}$$

A. Algorithm Description

As described in Algorithm 2, we build the temporal-spatial combined local GP model as follows. Given the training dataset $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N]^T$ and $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N]^T$, we first learn a set of hyperparameters $\{\bar{\theta}^i\}$ for the local spatial GP experts following the process described in the offline part of Algorithm 1. Then, the local temporal model is built up by the same way using the training data $\mathbf{Y}_1 = [\mathbf{y}_1, \ldots, \mathbf{y}_{N-1}]^T$ and $\mathbf{Y}_2 = [\mathbf{y}_2, \ldots, \mathbf{y}_N]^T$. From the final estimation result $\mathbf{y}_{t-1}^*$ at time instant $t-1$, one can obtain the prediction $\hat{\mathbf{y}}_t$ under the process of local temporal experts model. Finally, at the time instant t, we import $\mathbf{x}_t^*$ and $\hat{\mathbf{y}}_t$ into our temporal-spatial combined local experts model to get the final prediction $\mathbf{y}_t^*$.

TABLE I

COMPUTATIONAL COMPLEXITY. BOTH OF OUR LOCAL MODELS ARE LINEAR IN N FOR BOTH LEARNING AND INFERENCE, WHERE d IS THE DIMENSION OF THE DATA POINTS. IN OUR EXPERIMENTS, T, S, $R \ll N$

	Full GP	Local GP Experts	Temporal-Spatial Local GP Experts
Learning	$O(N^3)$	$O(RS^3 + RdN)$	$O(2RS^3 + 2RdN)$
Inference	$O(N^3)$	$O(TS^3 + TN)$	$O(TS^3 + TN)$

Algorithm 2 Online inference with temporal-spatial local GP experts

```
Require $\mathbf{x}_t^*$, $\mathbf{y}_{t-1}^*$: the output at last time instant
$p\left(\hat{\mathbf{y}}_t|\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{y}_{t-1}^*\right) \approx \sum_{j=1}^{M} \pi_j \mathcal{N}\left(\mu_j, \sigma_j^2\right)$ (see Algorithm 1)
**COMBINATION of two classes of local experts**
$T_1$: number of spatial experts
  $T_2$: number of temporal experts
  $S$: size of each expert
$\eta^{(s)} = \mathrm{findNN}\left(\mathbf{X}, \mathbf{x}_t^*, T_1\right)$;
$\eta^{(t)} = \mathrm{findNN}(\mathbf{Y}, \hat{\mathbf{y}}_t, T_2)$;
$\eta = \eta^{(s)} \cup \eta^{(t)}$;
**ONLINE inference**
$T = T_1 + T_2$: number of all experts
**for** $j = 1 \ldots T$ **do**
    $\zeta = \mathrm{findNN}(\mathbf{D}, \mathbf{d}_{\eta_j}, S)$
    $t = \mathrm{findNN}(\mathbf{C}_{\mathcal{R}}, \mathbf{d}_{\eta_j}, 1)$
    $\bar{\theta} = \bar{\theta}^t$
    $\mu_j = \mathbf{k}_{*,\zeta} \mathbf{K}_{\zeta,\zeta}^{-1} \mathbf{Y}_\zeta$
      $\sigma_j = k_{*,*} - \mathbf{k}_{*,\zeta} \mathbf{K}_{\zeta,\zeta}^{-1} \mathbf{k}_{\zeta,*}$
**end for**
$p\left(\mathbf{y}_t^*|\mathbf{X}, \mathbf{Y}\right) \approx \sum_{i=1}^{T} \pi_i \mathcal{N}\left(\mu_i, \sigma_i^2\right)$
```

It is worth pointing out that the our framework provides the mechanism to flexibly handle temporal discontinuity. It can switch off the local temporal experts once the temporal information is unavailable midway. Like most other temporal prediction approaches, in our model, there still exists the initialization problem. Here, the estimation $\hat{\mathbf{y}}_1$ at the first time instant is given by the local spatial experts alone. Fortunately, in many practical applications, multimodal is not everywhere in the data space. If the regression process starts from the unimodal region, the results will be satisfactory enough.

B. Computational Complexity

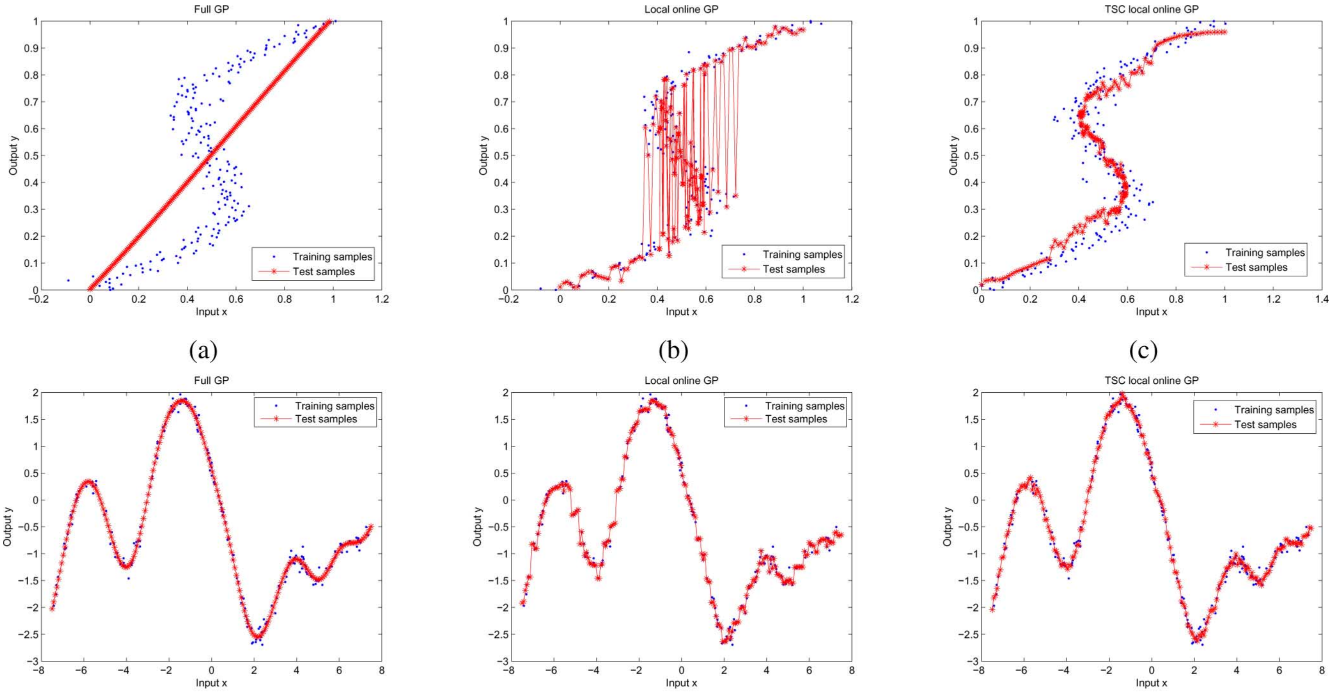
Table I shows the comparisons of computational complexity between our models and the full GP method. The computational complexity of conventional full GP is cubic of the number of involved samples. When the size of database grows to a large scale, the computational cost will become prohibitively high. In contrast, for both learning and inference, our local models are linear in N stemming from the operators of finding nearest neighbors ($\mathcal{O}(RN)$) and k-means clustering ($\mathcal{O}(RdN)$), where N is the total number of examples and d is the dimension of the data point. In particular, the computational complexity of learning in both of our models is the time of learning the R local experts $\mathcal{O}(RS^3)$ with size S, plus the time of k-means clustering $\mathcal{O}(RdN)$ for building the local experts model. The computation cost is doubled in the TSL-GP model because both spatial and temporal models need to be trained at the same time. As for the inference, both local models have the same computational complexity for computing the local GP, $\mathcal{O}(TS^3)$, and finding the neighbors, $\mathcal{O}(TN)$. Note that the complexity of inverting the local GP is not a function of the number of examples, since the local GP experts are of fixed size. When $N \gg S$, the computational cost is significantly reduced. Moreover, the complexity of our model is much smaller than that of full GP since in general R is a small value comparing to N. It is computational beneficial in dealing with very large size databases.

IV. EXPERIMENTS

In this section, we first validate our models on illustrative data generated from both multimodal function and unimodal function. We provide the reasoning of our models by visualizing the proof-of-concept results. We then conduct the experiments on two real-world datasets. The first one is the HumanEva-I database for the evaluation of human pose estimation collected at Brown University [25]. The second dataset we call PEAR (abbreviate of Pose Estimation and Action Recognition) is our recently released dataset for the pose and action related research collected at Shanghai Jiao Tong University. The detailed description of the novel dataset is presented in Section IV-C.

A. Regression on Multimodal and Unimodal Functions

We simulate two sets of toy data in this experiment for proof of concept [15]. The caption of Fig. 2 describes the details of the data. The regression results shown in Fig. 2 contain comparisons of full GP, local sparse (LS) GP (Algorithm 1) and Temporal-Spatial Local (TSL) GP (Algorithm 2). It can be seen that in the first row of Fig. 2, for the multimodal function, the full GP can only globally average the outputs of different modes. The local sparse GP, Fig. 2(b), can partly handle the multimodality and avoid the global averaging effect. However the prediction is still not sufficiently smooth. The outputs frequently skip between different modes in the multimodal regions. This problem can be solved in the TSL GP model due to the utilization of temporal information. Note that in Fig. 2(c), the skips are eliminated and the prediction is smooth. Another dataset provides a unimodal input-to-output mapping. As illustrated in Fig. 2(d–f), the full GP gives ideal regression results because the global voting mechanism can deal with the unimodal mapping very well. The



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 2. Model comparisons between full GP, Local Sparse GP, and TSL GP on two sets of illustrative data [15]. The first dataset (first row) consists of about 200 training pairs of (x, y), where y generated uniformly in (0,1) and evaluated as $x = y + 0.3\sin(2\pi y) + \epsilon$, with ϵ drawn from a zero mean Gaussian with standard deviation 0.05. Note here $p(y|x)$ is multimodal. Test points $(N_t = 200)$ are sampled uniformly from (0,1). The second dataset (second row) is obtained by sampling $(N = 100)$ a GP with covariance matrix obtained from an RBF. About 200 test inputs are sampled uniformly in $(-7.5, 7.5)$. The regression results are shown in: (a), (d) Full GP; (b), (e) Local Sparse GP; (c), (f) TSL GP.

TABLE II
DESCRIPTION OF THE HUMANEVA DATASET SPECIFIED BY FRAME NUMBERS

Set Partition	Action	S1	S2	S3	Total
Training set	Walking	613	438	393	1444
	Box	97	81	507	685
	Jog	228	397	348	973
Test set	Walking	386	433	267	1086
	Box	126	110	271	507
	Jog	85	393	396	874

local sparse GP also gives good results although there still exist some jitters. The TSL GP shows smoother prediction results than the local sparse GP model. In sum, the proposed TSL GP algorithm shows the most accurate and reliable performance in both multimodal and unimodal scenarios.

B. On the HumanEva Dataset

We evaluate our models on the HumanEva dataset [25]. The database provides synchronized video and motion capture streams. The frame rate of the video stream is 60 Hz. It contains multiple subjects performing a set of predefined actions with repetitions. The database was originally partitioned into training, validation, and testing subsets. We use sequences in the original training subset for training and original validation subset for testing. Table II shows the description of the HumanEva dataset we used in the experiments specified by frame numbers. As only consistent frames are considered in each sequence, there are in total 2530 frames for walking motion, 1847 frames for jog motion, and 1192 frames for box motion are used.

The pose is represented by 3-D joint centers, which are processed by subtracting root joint location. The "torsoDistal" is taken as the root joint. There are in total 15 joint points and end points of the limbs. Therefore, the output pose dimension is 45 (X, Y, Z coordinates). As for the image representation, we use three different types of features: Histogram of Oriented Gradients (HOG) [31], Local Binary Pattern (LBP) [32] and Histogram of SIFT [33]. Among the three features, Histogram of SIFT is extracted using bag-of-words model [34]. The local patches are centered on the sampled points on the silhouette and edges. We obtain the human silhouette by simple background subtraction. The number of sampled points in each frame is 400 and the size of code book is set to 300. LBP and HOG features are extracted on the regular overlapped grid patches per image. The dimension of the three features is reduced by PCA to keep at least 95% variance. All the images used in the experiments are captured by the C1 camera.

To evaluate the performance quantitatively, we make use of the metric proposed by [25], in which pose error is computed as an average distance between a set of 15 pose points. Therefore, the 3-D error (mm) can intuitively measures the distance between ground truth and estimated pose. On the sequence level, we compute the error by averaging all the pose errors over the whole sequence.

In the experiments, we evaluate the performance of five models, Nearest Neighbor (NN), full GP, LS-GP(U) defined in the unified space, LS-GP(S) defined in separate space (proposed in [13]), and TSL-GP, respectively. Table III shows the results of performance evaluation on the five models. It shows a big picture of the whole evaluation on all the three subjects and

TABLE III

PERFORMANCE EVALUATION ON THE HUMANEVA DATABASE, FOR THE FIVE MODELS USING HOG FEATURE EXTRACTED FROM C1 CAMERA. AVERAGE ERROR AND STANDARD DEVIATION (MM) OF THREE ACTIONS: WALKING, JOG AND BOX PERFORMED BY S1, S2, S3 ARE REPORTED. THE TRAINING SEQUENCE AND TEST SEQUENCE COME FROM SEPARATE TRIALS

	S1			**S2**			**S3**		
	Walking	Box	Jog	Walking	Box	Jog	Walking	Box	Jog
NN	42.1±23.6	94.6±36.6	90.0±31.9	26.8±19.7	66.1±19.5	67.2±30.2	57.6±28.3	49.1±20.0	61.6±22.7
Full GP	53.2±21.2	88.6±31.4	73.1±18.3	34.4±15.0	62.2±16.6	56.6±16.5	59.5±19.2	47.4±21.1	38.8±14.4
LS-GP(S)	37.4±19.3	74.9±29.4	63.7±21.6	23.4±7.3	60.8±15.6	51.1±19.2	54.3±19.6	45.8±20.8	36.4±13.8
LS-GP(U)	32.0±16.4	71.8±29.1	56.9±30.3	20.8±7.2	59.1±15.0	39.6±15.2	48.2±17.3	42.6±21.1	34.7±13.4
TSL-GP	21.5±7.1	35.9±18.2	30.2±13.5	12.3±5.6	28.3±8.1	23.1±7.4	25.8±8.5	19.2±8.4	15.6±6.9

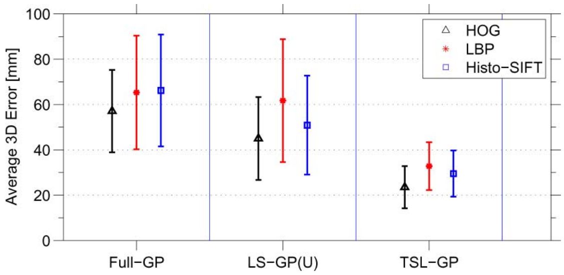


Fig. 3. Performance comparisons between three features, HOG, LBP and Histo-SIFT on three models.

their three actions recorded in the HumanEva database. Average error and standard deviation are reported. It is obvious that the TSL-GP model outperforms other models with significant improvements. The results align well with the conclusion from the previous section of proof-of-concept experiment, which suggests the introduction of temporal experts can effectively improve the performance of the prediction. Both LS-GP(U) and LS-GP(S) models have better performance than full GP although there are similar performances in some sequences. We also find that in the unified space, the local GP achieves some performance improvement although it is not significantly distinct. NN presents average performance comparing to the other four models, but in several sequences it is even better than full GP. The feature reported here is only HOG and the other two features, LBP and Histo-SIFT, show similar performance variations on the five models. Fig. 3 shows the performance comparisons between the three features on full GP, LS-GP(U) and TSL-GP model respectively. The error is averaged over all the sequences. Among the three features, HOG achieves the best performance.

In all the experiments, we take the values of R, T, S in the Algorithm 1 and 2 as 50, 10, 25, respectively. In the TSL-GP model, the number of spatial experts and temporal experts is set as the same value 10. Actually, the values are chosen experimentally and are data dependent. According to the size of our dataset and considering the computational cost, we set the value of R to 50. We study the impacts of the number of local experts and the size of each expert on final performance. Fig. 4 shows the results. We find that for the local models, relative small number of experts defined in close neighborhood can provide satisfactory results.

TABLE IV

CONFIGURATION OF THE PEAR DATABASE SPECIFIED BY FRAME NUMBERS IN EACH SEPARATE SUBSET

Set Partition	Action	S1	S2	S3	S4	Total
Training set	Walk	250	218	350	367	1185
	Jog	250	251	320	372	1193
	Jump	150	97	240	249	736
	Skip	150	124	238	242	754
	Wave	150	119	240	246	755
	Stretch	150	126	240	239	755
Validate set	Walk	233	211	240	335	1019
	Jog	250	208	320	369	1147
	Jump	150	120	215	241	726
	Skip	150	124	240	246	760
	Wave	150	85	240	250	725
	Stretch	150	127	240	231	748
Test set	Walk	250	218	350	361	1179
	Jog	250	251	320	355	1176
	Jump	146	120	239	242	747
	Skip	150	133	240	241	764
	Wave	150	86	240	241	717
	Stretch	149	118	240	234	741

In Fig. 5, the estimation results and ground truth represented by joint angles over the whole sequence of walking and jog actions are plotted. We compare the results of full GP and TSL-GP. It can be observed that the curves of the TSL-GP model are more smooth and close to the ground truth than the full GP model. Fig. 6 shows some sample frames together with the estimated pose and ground truth pose represented as the outline of a cylinder based human model superimposed onto the original images. C1, C2, and C3 cameras are used to show the estimated pose for the overall 3-D visualization.

As far as the computational cost is concerned, after the completion of training, the inference is effective with 2–6 frames per second, using unoptimized Matlab code.

C. *On the PEAR Dataset*

1) Dataset Description: The Pose Estimation and Action Recognition (PEAR) database is originally designed to facilitate the research on human pose estimation and action recognition. While the research on human motion analysis has been thriving in recent two decades, few benchmark datasets with synchronized video and motion capture data are available for 3-D pose estimation. Such dataset is important for the public evaluation of the state-of-the-art approaches. HumanEva database provides a good choice toward this aim. PEAR is another choice which is different from HumanEva in actions, subjects, camera setup, background and so forth.

The PEAR dataset is collected at a studio of Shanghai Jiao Tong University. There are 16 color cameras for video capture

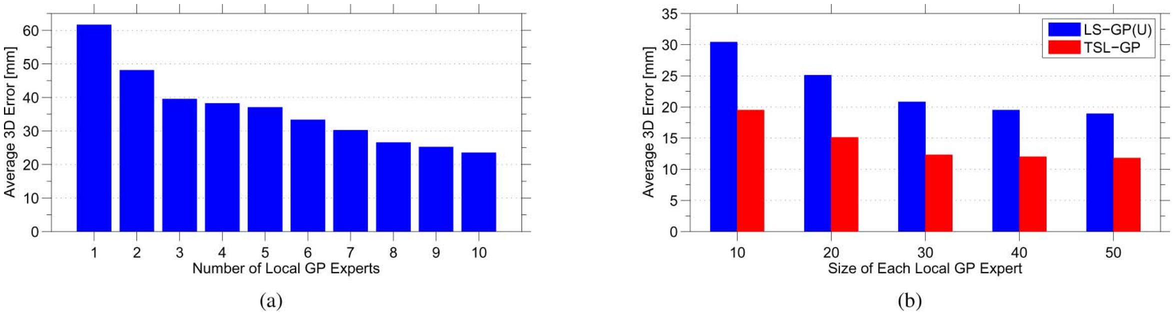


Fig. 4. Impacts of the number of experts (T) (a) and the size of each local expert (S) (b) on the performance. Relative small values of both T and S can provide satisfactory results.

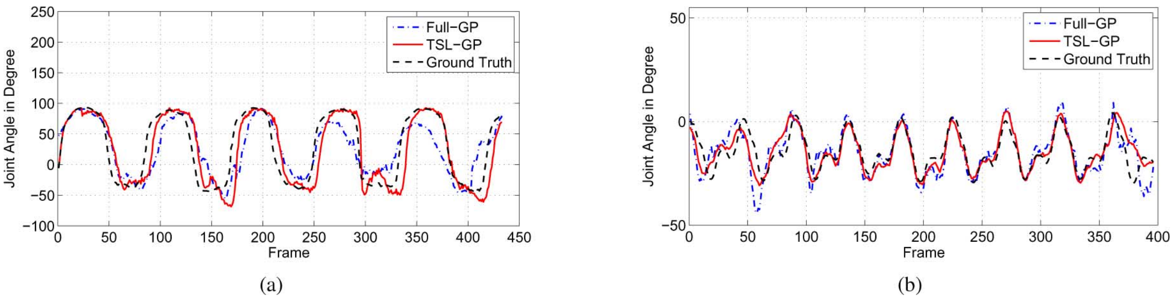


Fig. 5. Curve comparisons of joint angles: ground truth, estimations with TSL-GP and Full GP regression. (a) Left shoulder (x-axis) of subject S2 in walking action. (b) Right hip (x-axis) of subject S3 in jog action.

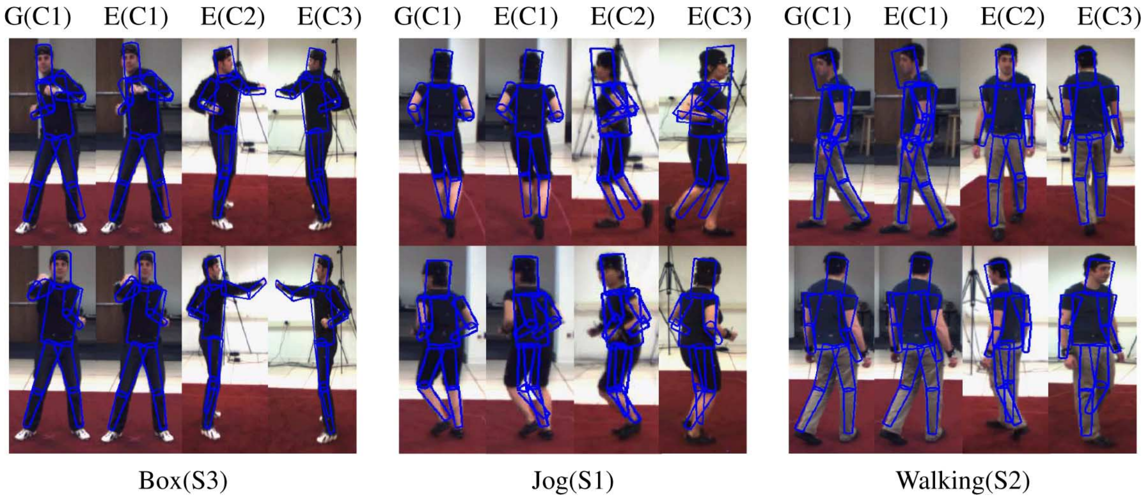


Fig. 6. Sample 3-D pose estimation results. The first column shows the provided ground truth projected onto camera C1. The other three columns show the estimated pose projected onto C1, C2, and C3 cameras, respectively. Each row corresponds to a frame.

and 12 cameras for motion capture setting up at different locations around the main area. The database consists of five subjects performing six predefined actions three times with both video and motion capture data in all trials. The data therefore are divided into three subsets for training, validating and testing respectively. The predefined actions include walk, jog, jump, skip, wave and stretch. In addition, a complex action of traditional Chinese dancing action is also available. The frame rates of video system and motion capture system are 25 Hz and 50 Hz, respectively. The synchronization between video and motion capture stream mainly relies on the hardware, but we have made a further refinement on the results. Since the number of cameras is up to 16, it is sufficient for voxel based pose reconstruction. The configuration of PEAR database is described in Table IV. Some sample images with different subjects performing different actions are shown in Fig. 7.

2) Performance Evaluation: We report the performance on PEAR dataset with the similar features and models evaluated on the HumanEva database. The camera used in the experiments is C1. In the feature level evaluations, HOG feature still outperforms the other two features for all models, so we use the experimental results from HOG feature hereafter.

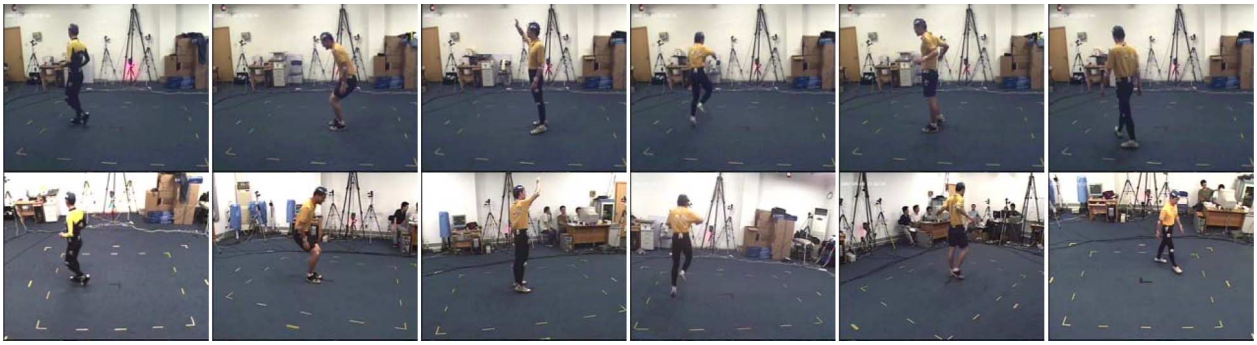


Fig. 7. Sample images in the PEAR database. The actions are jog, jump, wave, skip, stretch and walk respectively from left to right.

TABLE V
PERFORMANCE EVALUATION ON THE PEAR DATABASE. AVERAGE ERROR AND STANDARD DEVIATION (MM) OF THREE ACTIONS: WALK, JOG AND JUMP PERFORMED BY S1, S2, S3 ARE REPORTED. THE PERFORMANCES ARE EVALUATED ON FIVE MODELS USING HOG FEATURE EXTRACTED FROM C1 CAMERA

	S1			**S2**			**S3**		
	Walk	Jog	Jump	Walk	Jog	Jump	Walk	Jog	Jump
Full GP	123.2±58.9	59.1±15.3	96.1±24.3	113.2±50.9	83.1±27.4	116.6±31.5	95.1±33.2	98.9±31.1	108.3±34.6
LS-GP(U)	85.2±29.8	50.7±18.2	77.9±28.9	82.3±27.6	78.7±21.4	91.5±29.7	81.3±28.5	83.6±29.8	79.5±29.7
TSL-GP	42.9±18.3	38.3±15.9	45.6±22.2	45.8±21.4	63.9±17.5	62.2±18.6	56.2±25.9	74.1±21.8	68.4±18.0
NN	194.3±99.7	84.5±37.8	91.1±59.5	129.3±31.4	114.5±32.7	119.1±45.0	127.9±44.6	103.4±49.1	98.0±31.4
LB	44.9±16.6	39.5±19.1	39.9±23.5	53.4±17.5	61.0±21.1	60.8±23.4	54.0±15.1	79.9±20.6	66.7±25.2

For local GP models, we first study the impacts of the number of experts and the size of each expert on the performance. The results are shown in Fig. 8. It can be seen that when the size of each expert is fixed, the performance improves with the increase of the number of experts. But the improvements are insignificant when the number increases to 7. It is the similar phenomenon when changing the size of each expert. Therefore, we empirically set the values of T and S in Algorithm 1 and 2 as 10 and 25, respectively. The detailed experimental results are reported in Table V. The mean errors and standard derivation for three subjects performing three different actions, walk, jog and jump, are listed respectively. To measure the error bound, we compute the "quasi error Lower Bound (LB)" of one sequence by computing the distance between each test sample and its nearest neighbor in the training sequence and then averaging the distances over the whole sequence. In fact this is not a strict lower bound but a reasonable reference for the evaluation. From the table we can see that the TSL-GP model again shows advantages. However, the mean errors are much larger than that on the HumanEva. One major reason for this phenomenon is the difference of frame rates between the two databases. The frame rate of PEAR database is 25, so the mean error between neighbor samples is larger than that on the HumanEva.

In Fig. 9, we show some sample results of pose estimation on the PEAR database. For 3-D visualization, the estimated pose are projected onto four different cameras with calibrated camera parameters.

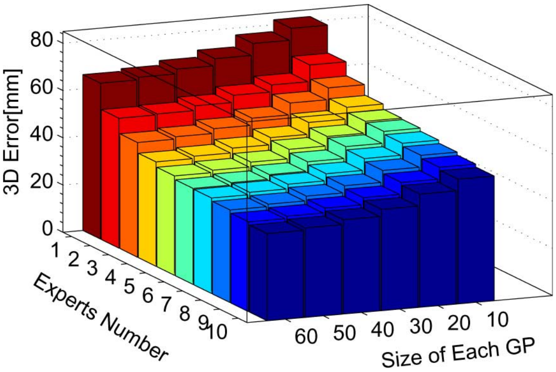


Fig. 8. 3-D bar visualization of the impacts of the number of experts and the size of each expert on the performance.

V. Conclusion

We have presented a novel temporal-spatial combined local GP experts model for efficient estimation of 3-D human pose from monocular images. Our model is essentially a type of mixture of GP experts in which we incorporate both spatial and temporal information into a seamless system to handle multimodality. The local experts are trained in the local neighborhood. Different from previous work, the neighborhood relationship is defined in the unified input-output space. Therefore, we can flexibly handle two-way multimodality. Learning and inference of this model are extremely efficient because both spatial and temporal local experts are defined online within very small neighborhoods. As an extension of our previous research in [15], extensive comparative experiments on the real-world HumanEva database and PEAR database have validated the efficacy of the proposed model by achieving accurate human motion tracking results. As a generalized model, its adaption to other scenarios is feasible and straightforward. In the future work, we will explore the automatic switch mechanism to deal with the large temporal jump and discontinuity.

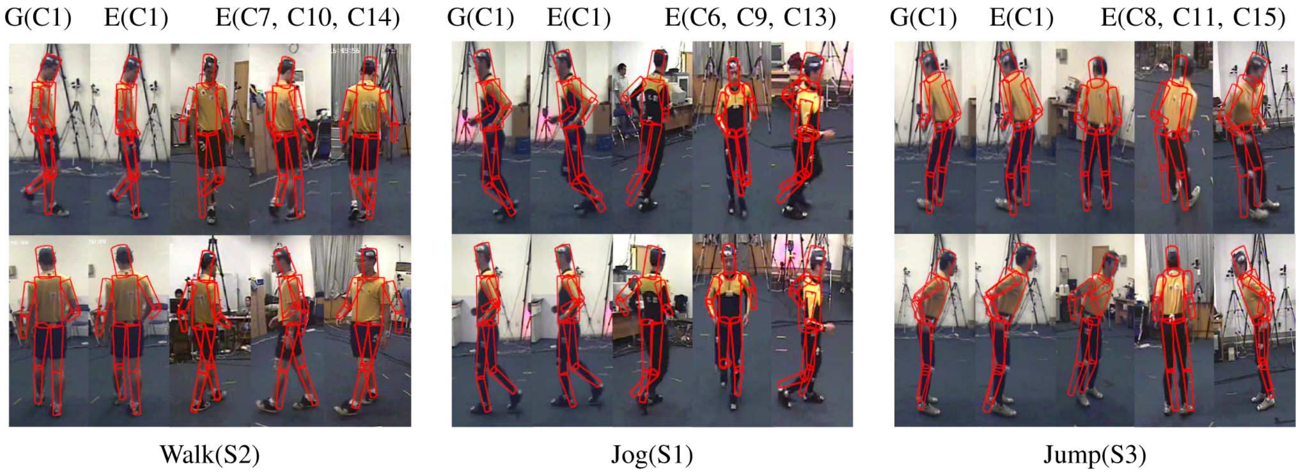


Fig. 9. Sample 3-D pose estimation results. The first and second columns in each action subfigure correspond to the ground truth pose and the estimated pose projected back onto C1 camera. Other three columns correspond to another three cameras. Each row corresponds to a frame.

References

[1] T. B. Moeslund, A. Hilton, and V. Krüger, "A survey of advances in vision-based human motion capture and analysis," *Comput. Vis. Image Understand.*, vol. 104, no. 2–3, pp. 90–126, 2006.

[2] C. Sminchisescu, A. Kanaujia, Z. Li, and D. Metaxas, "Discriminative density propagation for 3-D human motion estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, vol. 1, pp. 390–398.

[3] A. Agarwal and B. Triggs, "Recovering 3-D human pose from monocular images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 28, no. 1, pp. 44–58, Jan. 2006.

[4] A. Elgammal and C. S. Lee, "Inferring 3-D body pose from silhouettes using activity manifold learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2004, vol. 2, pp. 681–688.

[5] H. Ning, X. Wei, Y. Gong, and T. S. Huang, "Discriminative learning of visual words for 3-D human pose estimation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, pp. 1–8.

[6] A. Bissacco, M.-H. Yang, and S. Soatto, "Fast human pose estimation using appearance and motion via multi-dimensional boosting regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2007, pp. 1–8.

[7] X. Zhao, H. Ning, Y. Liu, and T. S. Huang, "Discriminative estimation of 3-D human pose using Gaussian processes," in *Proc. 19th Int. Conf. Pattern Recognit.*, 2008, pp. 1–4.

[8] X. Zhao, Y. Fu, H. Ning, Y. Liu, and T. S. Huang, "Human pose regression through multiview visual fusion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, pp. 957–966, 2010.

[9] G. Shakhnarovich, P. Viola, and T. Darrell, "Fast pose estimation with parameter-sensitive hashing," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, vol. 2, pp. 750–757.

[10] C. Tomasi, S. Petrov, and A. Sastry, "3D tracking = classification + interpolation," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2003, vol. 2, pp. 1441–1448.

[11] A. Agarwal and B. Triggs, "3D human pose from silhouettes by relevance vector regression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2004, vol. 2, pp. 882–888.

[12] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA: MIT Press, 2006.

[13] R. Urtasun and T. Darrell, "Local probabilistic regression for activity-independent human pose inference," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2008, vol. 2, pp. 1–8.

[14] R. Urtasun, D. J. Fleet, A. Hertzmann, and P. Fua, "Priors for people tracking from small training sets," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, vol. 1, pp. 403–410.

[15] X. Zhao, Y. Fu, and Y. Liu, "Temporal-spatial local Gaussian process experts for human pose estimation," in *Proc. 9th Asian Conf. Comput. Vis.*, 2009, vol. 5994, pp. 364–373.

[16] J. Quiñonero-Candela and C. E. Rasmussen, "A unifying view of sparse approximate Gaussian process regression," *J. Machine Learning Res.*, vol. 6, pp. 1939–1959, 2005.

[17] N. D. Lawrence, M. Seeger, and R. Herbrich, "Fast sparse Gaussian process methods: The informative vector machine," in *Proc. Advances in Neural Information Processing Systems*, 2003, pp. 625–632.

[18] E. Snelson and Z. Ghahramani, "Sparse Gaussian processes using pseudo-inputs," in *Proc. Advances in Neural Information Processing Systems*, 2006, pp. 1257–1264.

[19] C. E. Rasmussen and Z. Ghahramani, "Infinite mixtures of Gaussian process experts," in *Proc. Advances in Neural Information Processing Systems*, 2002, pp. 881–888.

[20] V. Tresp, "Mixtures of Gaussian processes," in *Proc. Advances in Neural Information Processing Systems*, 2001, pp. 654–660.

[21] E. Meeds and S. Osindero, "An alternative infinite mixture of Gaussian process experts," in *Proc. Advances in Neural Information Processing Systems*, 2006, pp. 883–890.

[22] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, 1991.

[23] R. Urtasun, S. Epfl, D. J. Fleet, and P. Fua, "3D people tracking with Gaussian process dynamical models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2006, vol. 1, pp. 238–245.

[24] J. M. Wang, D. J. Fleet, and A. Hertzmann, "Gaussian process dynamical models," in *Proc. Advances in Neural Information Processing Systems*, 2005, pp. 1441–1448.

[25] L. Sigal and M. J. Black, "Humaneva: Synchronized Video and Motion Capture Dataset for Evaluation of Articulated Human Motion," Brown University, Providence, RI, 2006, Tech. Report CS-06-08.

[26] L. Csato and M. Opper, "Sparse on-line Gaussian processes," *Neural Computation*, vol. 14, no. 3, pp. 641–668, 2002.

[27] A. J. Smola and P. Bartlett, "Sparse greedy Gaussian process regression," in *Proc. Advances in Neural Information Processing Systems*, 2001, pp. 619–625.

[28] M. Seeger, C. K. I. Williams, and N. Lawrence, "Fast forward selection to speed up sparse Gaussian process regression," in *Proc. 9th Int. Workshop on Artificial Intelligence and Statistics*, Key West, FL, 2003, ISBN 0-9727358-0-1, Online.

[29] S. S. Keerthi and W. Chu, "A matching pursuit approach to sparse Gaussian process regression," in *Proc. Advances in Neural Information Processing Systems*, 2006.

[30] D. Nguyen-Tuong, M. Seeger, and J. Peters, "Local Gaussian process regression for real time online model learning," in *Proc. Advances in Neural Information Processing Systems*, 2008.

[31] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 886–893.

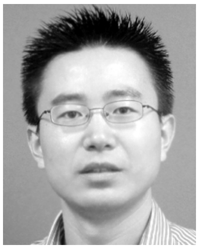
[32] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 7, pp. 971–987, 2002.

[33] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, 2004.

[34] L. Fei-Fei and P. Perona, "A Bayesian hierarchical model for learning natural scene categories," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2005, pp. 524–531.

Xu Zhao (M'10) received the M.S. degree in electrical engineering from China Ship Research and Develop Academy in 2004. He is currently pursuing the Ph.D. degree at the Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University, China. He was a visiting student at the Beckman Institute for Advanced Science and Technology at University of Illinois at Urbana-Champaign from 2007 to 2008.

His research interests include visual analysis of human motion, machine learning, and image/video processing.



Yun Fu (S'07–M'08–SM'11) received the B.Eng. degree in information engineering in 2001 and the M.Eng. degree in pattern recognition and intelligence systems in 2004, both from Xi'an Jiaotong University (XJTU), China, and the M.S. degree in statistics in 2007 and the Ph.D. degree in electrical and computer engineering in 2008, both from the University of Illinois at Urbana-Champaign (UIUC).

He was a research intern with Mitsubishi Electric Research Laboratories, Cambridge, MA, in summer 2005, and with Multimedia Research Lab of Motorola Labs, Schaumburg, IL, in summer 2006. He joined BBN Technologies, Cambridge, MA, as a Scientist in 2008. He held a part-time Lecturer position at the Department of Computer Science, Tufts University, Medford, MA, in the spring of 2009. He joined the Department of Computer Science and Engineering, SUNY at Buffalo, as an Assistant Professor in 2010.

Dr. Fu's research interests include Applied Machine Learning, Human-Centered Computing, Pattern Recognition, Intelligent Vision System. He is the recipient of the 2002 Rockwell Automation Master of Science Award, Edison Cups of the 2002 GE Fund Edison Cup Technology Innovation Competition, the 2003 Hewlett-Packard (HP) Silver Medal and Science Scholarship, the 2007 Chinese Government Award for Outstanding Self-financed Students Abroad, the 2007 DoCoMo USA Labs Innovative Paper Award (IEEE ICIP'07 best paper award), the 2007-2008 Beckman Graduate Fellowship, the 2008 M. E. Van Valkenburg Graduate Research Award, the ITESOFT Best Paper Award of 2010 IAPR International Conferences on the Frontiers of Handwriting Recognition (ICFHR), and the 2010 Google Faculty Research Award. He is a life member of Institute of Mathematical Statistics (IMS) and Beckman Graduate Fellow.

Yuncai Liu (M'94) received the Ph.D. degree in electrical and computer science engineering from the University of Illinois at Urbana-Champaign in 1990.

He worked as an associate researcher at the Beckman Institute of Science and Technology from 1990 to 1991. Since 1991, he had been a system consultant and then a chief consultant of research in Sumitomo Electric Industries Ltd., Japan. In October 2000, he joined the Shanghai Jiao Tong University, China, as a Distinguished Professor. His research interests are in image processing and computer vision, especially in motion estimation, feature detection and matching, and image registration. He also made many progresses in the research of intelligent transportation systems.