An integrated physical-learning model of physical human-robot interactions with application to pose estimation in bikebot riding

Kuo Chen¹, Yizhai Zhang², Jingang Yi¹ and Tao Liu³

Abstract
Modeling physical human-robot interactions (pHRI) is important in studying human sensorimotor skills and designing human assistive and rehabilitation systems. One of the main challenges for modeling pHRI is the high dimensionality and complexity of human motion and its interactions with robots and the environment. We present an integrated physical-learning pHRI modeling framework with applications to the bikebot riding example. The modeling framework contains an integrated machine-learning-based model for high-dimensional limb motion with a physical-principle-based dynamic model for the human trunk and an interacted bicycle-like robot (bikebot). A new axial linear embedding algorithm is used to obtain the low-dimensional latent dynamics for highly redundant human limb movement. The integrated physical-learning model is then used to estimate human motion through an extended Kalman filter design without using any sensors attached to the limb segments. Extensive bikebot riding experiments are conducted to validate and demonstrate the integrated pHRI model. Comparison results with other machine-learning-based models are also presented to demonstrate the superior performance of the proposed modeling framework for bikebot riding.

Keywords
Human-robot interactions, machine learning, model reduction, pose estimation, Kalman filter

1. Introduction
Humans with trained motor skills can fluidly and flexibly interact with machines, while smart machines or robots can also provide motor assistance and enhancement to facilitate a human’s motor skills learning (Reinkensmeyer and Patton, 2008). Modeling of physical human-robot interactions (pHRI) is important in understanding the role of human sensorimotor control in trained motor skills with machines or robots and to design human assistive and rehabilitation systems. One of the main challenges for modeling pHRI is the high dimensionality and complexity of human motion and its interactions with machines (Ikemoto et al., 2012). The goal of this paper is to present an integrated physical-learning pHRI model with applications to the bikebot (i.e. bicycle-based robot) riding example.

The proposed pHRI modeling framework uses physical principles to model the dynamic motions of the robot and the human trunk while a machine-learning-based method is employed to capture human limb motion in a low-dimensional latent space (Safonova et al., 2004). The physical model and the learning model are interconnected and integrated to describe the pHRI. The rationales of using the integrated physical-learning pHRI modeling approach are twofold. First, one main challenge of the modeling and control of pHRI is the high-dimensional human motion and anatomical redundancy of human body segments. Using physical principles for rigid body dynamics, such as Lagrangian or Newtonian mechanics, generates high-dimensional models that are difficult to use for control systems and estimation design. For many human activities, high-dimensional motions in physical joint space are highly coordinated and can thus be represented by low-dimensional latent dynamics (Rosenhahn et al., 2008). Secondly, the physical modeling approach is commonly used to describe robot dynamics. The physical interactions between the human and the robot, such as forceful contacts and coordinated movements, provide additional properties and constraints for both the human and robot dynamics. The two modeling methods are complementary and the integration of the physical model for the robot and the learning model for human motion naturally provides a means to capture and incorporate the interaction characteristics.

¹Rutgers University, USA
²Northwestern Polytechnical University, China
³Zhejiang University, China

Corresponding author:
Jingang Yi, Department of Mechanical and Aerospace Engineering, Rutgers University, Piscataway, NJ 08854, USA.
Email: jgyi@rutgers.edu
Dimensional reduction and manifold learning methods are used to capture human motion characteristics for applications such as humanoid robot control (Artemiadis and Kyriakopoulos, 2010; Chalodhorn et al., 2010; Havoutis and Ramamoorthy, 2010) and human tracking and activity recognition (Elgammal and Lee, 2009; Schwartz et al., 2008; Wang et al., 2008; Zhao et al., 2011). Low-dimensional manifolds are learned and used to capture high-dimensional human movements. In Havoutis and Ramamoorthy (2013), motion planning is conducted on the learned skill manifolds without the need to build analytical robotic models. Embedded skill manifolds are also presented in Feix et al. (2013) and Romero et al. (2013) to represent and encode human hand motion and compare the motion capability of the robotic hands with the human hands. Nonlinear dimensionality reduction and Gaussian process latent variable models (GP-LVM) are used in Feix et al. (2013) and Romero et al. (2013) to obtain the skilled manifolds.

One potential drawback of using a machine learning or data-driven modeling approach is that commonly-used dimensionality reduction algorithms (e.g. principal component analysis (PCA) or locally linear embedding (LLE)) do not preserve the physical meaning of the obtained low-dimensional latent variables. It is difficult to interpret these variables and build physical connections with joint angles. To overcome this shortcoming, we present an axial linear embedding (ALE) algorithm to conduct the dimensional reduction that preserves the physical interpretation of the latent variables. Besides the physical interpretation, the use of ALE also demonstrates superior performance to other commonly used dimensional reduction approaches, such as PCA or LLE (Rosenhahn et al., 2008).

In Wang et al. (2008), a Gaussian process dynamic model (GPDM) is proposed to capture high-dimensional time series data with application to human motion animation. In Ko and Fox (2011), a GPBF-LEARN model is introduced to improve GPDM for robotics applications by incorporating control input and initialization of latent coordinates. A predictive model in the low-dimensional latent space and an observation model mapping from the latent space to the high-dimensional data space are presented and established. Because of the non-convex optimization problem, the solution is subject to a local optimum and hence the initialization (labels) of the latent coordinates is critical for performance. The GPBF-LEARN approach shows the advantage of labeling latent coordinates with their observed values. However, the latent coordinates obtained by GPBF-LEARN in Ko and Fox (2011) do not maintain the physical meanings that correspond to high-dimensional joint angles. In this paper, we adopt GPBF-LEARN’s structure and incorporate the ALE method for dimensionality reduction to construct physically meaningful labels for latent coordinates corresponding to high-dimensional joint angles. Bikebot riding experiments show that this method outperforms other algorithms such as PCA and LLE for latent coordinates initialization.

Ko and Fox (2009) tried to use the physical model with the learning model. The model integration in Ko and Fox (2009) is different from what we present in this paper. Instead of establishing the dynamic relationship between the inputs and the outputs, the learning model in Ko and Fox (2009) captures the difference between the experiment data and the predictions made by the physical model. Therefore, the learning model serves as a correction to the physical model. Instead, we consider physical and learning modeling approaches for different types of coupled physical systems (i.e. robot and human body segments, respectively) and use the physical interactions and constraints to enhance the modeling accuracy. Such treatment takes advantages of both the models’ attractive, complementary properties, such as the simple, low-dimensional representation of the latent models, and the high-fidelity, physically interpretable dynamics models.

Bikebot riding is used as an example to test and validate our modeling method. Bikebot is an actively controlled bicycle-based robot that was developed for studying human balancing motor skills (Zhang et al., 2014b). We use bicycle-like riding as the testbed primarily for two reasons. First, riding the bicycle requires the coordinated control of multi-limb and body movements and thus offers an attractive platform for studying human postural balance motor skills. Secondly, the interactions between the rider and the bikebot are through multiple forceful contacts, such as the steering handlebar, seat and pedaling etc. These interactions provide dynamic and geometric constraints between the rider and the bikebot. In Zhang et al. (2013), a pose estimation scheme is presented to predict trunk and bicycle orientations from the inertial measurement units (IMU) and seat force sensor measurements in real-time. Unlike the work in Zhang et al. (2013), the main advantage of using the physical-learning model is that no IMU sensors need to be attached to the rider’s limb segments.

The work presented in this paper is a significant extension of Chen et al. (2013, 2014) in several aspects. Bikebot and trunk dynamics are used in this paper while these dynamics are not considered in Chen et al. (2013). Moreover, we extend and enhance the machine learning models by incorporating the new ALE dimensionality reduction algorithm, while only the GPDM is used with the PCA in Chen et al. (2013). Comparing with the previous conference publication (Chen et al., 2014), this paper includes significant extensions. We present new multiple-subject experiments to validate the integrated physical-learning model, while Chen et al. (2014) contains only single-subject indoor experiments. This paper also presents new experimental results and discussions such as perturbed rider experiments by the bikebot actuators, extensive experiments and comparisons with other learning methods, the physical model validation experiments, the whole-body pose estimation results and the latent dynamics comparisons, etc. The main contribution of this work lies in the new physical-learning pHRI modeling framework and the new ALE dimensionality reduction algorithms. The latent space preserves the
physical interpretation and thus enables the analysis and characterization of the pHRI properties on a reduced low-dimensional manifold. We also demonstrate an application to estimate the attitudes of the body segments without the use of wearable sensors attached to them in real time.

The remainder of this paper is organized as follows. The physical-learning modeling approach is presented in Section 2. We present the modeling application to the rider-bikebot system in Section 3. The extended Kalman filter (EKF)-based pose estimation is discussed in Section 4 and experiments are presented in Section 5. Finally, we conclude the paper in Section 6.

2. Physical-learning modeling framework

2.1. Model overview

The integrated physical-learning model uses physical principles to capture the dynamic motion of the robot, the human trunk and the machine learning-based model to describe the limb dynamic motion. The human-robot interaction forces and geometric constraints are used to integrate the physical and learning models. Figure 1 illustrates the physical-learning pHRI modeling framework.

The physical-learning pHRI model is written as \( \Sigma_{pl} = \Sigma_r \times \Sigma_t \times \Sigma_{dl} \times \Sigma_{dt} \), where \( \Sigma_r \) is robot dynamics, \( \Sigma_t \) is human trunk dynamics and \( \Sigma_{dl} \) and \( \Sigma_{dt} \) are the upper- and lower-limb dynamics in the low-dimensional latent space. Note that \( \Sigma_r \) and \( \Sigma_t \) are obtained through physical principles (e.g. Lagrangian mechanics) while \( \Sigma_{dl} \) and \( \Sigma_{dt} \) are obtained through dimensionality reduction and machine learning techniques. Let \( \mathbf{q}_r \in \mathbb{R}^{n_r} \) and \( \mathbf{q}_t \in \mathbb{R}^{n_t} \) denote the generalized coordinates (with dimensions \( n_r \) and \( n_t \)) for the robot and the human trunk motions, respectively. Let \( \mathbf{q}_{rt} = [\mathbf{q}_t^T \mathbf{q}_r^T]^T \in \mathbb{R}^{n_t+n_r} \) and the robot-trunk dynamics are obtained as

\[
M(q_t)\ddot{q}_t + C(q_t, \dot{q}_t, q_r) + G(q_t) = u_t
\]

where \( M(q_t) \), \( C(q_t, \dot{q}_t, q_r) \), and \( G(q_t) \) are the inertia matrix, Coriolis and gravity vectors, respectively, and \( u_t \) is the torque input for human trunk and the robot. In Section 3, we will illustrate how to obtain equation (1) for the rider-bikebot system.

Our choice to use physical models to capture human trunk motion and learning models for limb motion primarily lies with several considerations. The first consideration is the dimensionality of the trunk and limb motions in human locomotions and activities. For many human activities such as riding a bicycle (bikebot), the trunk movement can be captured as a rigid 3-degrees-of-freedom (DOF) inverted pendulum (Zhang et al., 2013), while the motions of each limb have more than 5 DOFs. It is undesirable to use physical models to capture the constrained high-DOF limb motions due to their complexity. Another motivation to use learning models to capture the limb motion lies in the coordinated motion pattern among limbs in many human activities, such as bicycling, walking, running, etc. It is advantageous to use learning models to capture these coordination characteristics on low-dimensional manifolds rather than using any constrained physical models.

The dynamics of the bikebot-trunk given in equation (1) are used as the physical models because the motions result from forces and torques in the pHRI. Moreover, for the pose estimation applications, we take advantage of the non-drifting property of the measured human-robot interactions forces/torques to reduce or eliminate the drifts due to inertial sensor noises. Integration of the force and inertial measurements provides a robust means to completely eliminate the estimation drifts due to inertial sensor noises (Zhang et al., 2013). Therefore, using the bikebot-trunk dynamic constraints can improve the accuracy of the pose estimation as shown in the experiments in Section 5. Although the learning models presented in this work are built on conveniently obtained joint angles, the physical-learning modeling framework is not restricted to the kinematics information, and the dynamics of limb motions can also be used and integrated into the learning models.

2.2. Learning model for human limb motion

We adopt a machine-learning-based latent dynamic model to represent the limb motion. The latent dynamic model consists of two parts: predictive latent state dynamics and an observation model that maps low-dimensional latent variables to high-dimensional joint angles. We denote the latent state variable as \( x \in \mathbb{R}^d \) and the limb joint angles as \( y \in \mathbb{R}^D \), where \( d \) and \( D \) \((d \ll D)\) are the dimensions of the latent
space and the joint angle space, respectively. Figure 2 illustrates the latent dynamics model structure. For presentation clarity, we use a discrete-time representation for the learning model. To capture the physical human-robot interaction, a control input $u_i \in \mathbb{R}^{n_h}$ and a geometric constraint $z \in \mathbb{R}^{n_c}$ are used in the latent dynamic model, where $n_h$ and $n_c$ are the dimensions of the control inputs and the constraints, respectively.

The latent dynamics $\Sigma_i$ for limb motion are formulated as

$$\Delta x(k) = f(x(k-1), u_i(k-1), \alpha) + w_p,$$

$$y(k) = g(x(k), \beta) + w_o,$$

where $\Delta x(k) := x(k) - x(k-1)$, $\alpha$ and $\beta$ are system parameters, $w_p$ and $w_o$ model the noises. For the training data set $\{(y(k))^{N} \}$ and control input set $\{u_i(k)^{N}\}$, $N$ is the number of the training data points, we estimate maps $f(\cdot, \cdot, \cdot)$ and $g(\cdot, \cdot)$ in equation (2) by identifying $\alpha$ and $\beta$. Denoting $X = \{x(k)\}^{N}$, $Y = \{y(k)\}^{N}$ and $U = \{u_i(k)\}^{N}$, the system identification problem is formulated as maximizing a-posterior distribution $P(X, \alpha, \beta | Y, U, \hat{X})$. Here $\hat{X} = \{\hat{x}(k)\}^{N}$ (i.e. label of $X$) is used to initialize $X$ in the optimization process. We adopt the GPBF-LEARN structure to factorize the objective function as

$$P(X, \alpha, \beta | Y, U, \hat{X}) \propto P(Y|X, \beta) P(X|U, \alpha) P(X|\hat{X}) P(\alpha) P(\beta).$$

The first term $P(Y|X, \beta)$ is factorized as the product of $D$ Gaussian process regression models with each one corresponding to the regression of the $i$th dimension of $y \in \mathbb{R}^{D}$. Denoting $y_{i} = \{y_{i}(k)\}^{N}$, we have

$$P(Y|X, \beta) = \prod_{i=1}^{D} P(y_{i}|X, \beta_{i}) = \prod_{i=1}^{D} \mathcal{N}(y_{i} | 0, K_{y_{i}} + \sigma_{n_{y_{i}}}^{2} I),$$

where $K_{y_{i}}$ is an $N \times N$ kernel matrix with $K_{y_{i}}[p, q] = k_{y_{i}}(x_{p}, x_{q})$. A squared exponential kernel is chosen as $k_{y_{i}}(x_{p}, x_{q}) = \sigma_{x_{i}}^{2} e^{-\frac{1}{2}(x_{p} - x_{q})^{T} W_{i} (x_{p} - x_{q})}$, where $W_{i}$ is diagonal matrix for weighting different inputs. $\beta = \{\sigma_{x_{i}}, \beta_{i}, \sigma_{n_{y_{i}}}\}$ is the hyper-parameter set trained for the $i$th dimension Gaussian process regression. To model observation function $g$ in equation (2b), $D$-dim GPDM hyper-parameter sets $\beta = \{\beta_{j}\}^{D}$ are learned and obtained.

To estimate the latent dynamics (2a), we use the Gaussian process model to learn the regression relationship between $\Delta x_{i}(k) = x_{i}(k) - x_{i}(k-1)$ and $s(k-1) = [x^{T}(k-1) u_{i}(k-1)]^{T}$, $i = 1, 2, \ldots, d$. Therefore, the second term in equation (3) is factorized as

$$P(X|U, \alpha) = \prod_{i=1}^{d} P(x_{i}|U, \alpha_{i})$$

$$= \prod_{i=1}^{d} \mathcal{N}(\Delta x_{i} | 0, K_{x_{i}} + \sigma_{n_{x_{i}}}^{2} I),$$

where $x_{i} = \{x_{i}(k)\}^{N}$, $\Delta x_{i} = \{\delta x_{i}(k)\}^{N}$, $K_{x_{i}}[p, q] = k_{x_{i}}(s_{p}, s_{q})$ is the kernel function for the $i$th dimension of the predictive function (2a) and $\sigma_{n_{x_{i}}}^{2} = \sigma_{x_{i}}^{2} e^{-\frac{1}{2}(s_{p} - s_{q})^{T} W_{1} (s_{p} - s_{q})}$. Terms $\alpha_{i} = \{\sigma_{n_{x_{i}}}, W_{x_{i}}, \sigma_{n_{y_{i}}}\}$ are the hyper-parameters learned for the $i$th dimension of the predictive function mapping $f$. Up to $d$-dim GPDM regression hyper-parameter sets $\alpha = \{\alpha_{i}\}^{d}$ are learned.

The third term in equation (3) is expressed as an identically independent Gaussian distribution with preset observation noise variance $\sigma_{n_{x_{i}}}^{2} I$, namely, $P(X|\hat{X}) = \prod_{i=1}^{N} \mathcal{N}(x(k)|\hat{x}(k), \sigma_{n_{x_{i}}}^{2} I)$. This term expresses the confidence of the label of the latent coordinates. With each term in equation (3) specified, the learning algorithm takes the training data $Y$ and $U$ and initial values $X, \alpha_{ini}$ and $\beta_{ini}$ to optimize equation (3) with respect to $X, \alpha$, and $\beta$. The optimization process is implemented with a scale conjugate gradient algorithm (Ko and Fox, 2011).

Constraints $z$ include the geometric and dynamics relationships and commonly exist in the physical human-robot interactions. These constraints can be integrated and incorporated during the stage to obtain the learning models such as the approach discussed in Chen et al. (2013). In this work, we will use the EKF to fuse various sensing information for the pose estimation application and the integration of constraints $z$ into the EKF design will be presented and discussed in Section 4.

2.3. Learning model initialization

Because of the non-convexity of the above optimizing process, the initialization of $X$ is critical to avoid local minimums and obtain the correct results. We propose a novel latent coordinates labeling approach, termed as axial linear embedding (ALE). The construction of the ALE reduction algorithm is inspired by the observation that the limb motion heavily depends on the trunk motion and the limb-robot interactions. This observation comes from the fact...
that the two endpoints of each limb are connected to the trunk (limb-shoulder connection) and the robot (hand- or foot-robot interaction). For example, when riding a bicycle, two hands are always holding the handlebar and two feet are in contact with the pedals. Therefore, the limb segment poses are mainly determined by the trunk and the bicycle motion.

The ALE algorithm is illustrated in Algorithm 1. We define an equilibrium point \( q^e \), at which the pHRI is at either stable or comfortable locations. For example, for riding a bicycle, \( q^e \) is defined when rider sits on the seat with two arms at a comfortable, natural position on the handlebars and the bicycle at vertical position and zero steering angle. At \( q^e \), the limb joint angles are at \( y^e \). To learn the model effectively, we subtract all the limb joint angles by \( y^e \) so that the center of the joint angles space is at zero. We abuse the notation slightly and still use \( y \) to denote the de-centered joint angles without causing any confusion. Then, we perturb the human’s motion around \( q^e \) by only moving along \( j \)th coordinate direction of \( q^e \), \( j = 1, 2, \ldots, n_r, n_r = d \) is the dimension of the latent space, and record the limb joint angles set \( \{ y_i \}_q \). We call this set of perturbed motion the template of \( q^e \), denoted as \( T_r \) is \( \text{span}(q^e) \). Using the PCA method, we factorize the first principal component of \( T_r \) to obtain \( \{ x_i \}_q = \text{PCA}(\{ y_i \}_q) \). After \( n_r \) experiment runs, \( \{ x_i \}_q \) are obtained for the \( d \)-dimensional latent space. This process is shown in lines 1 to 3 of Algorithm 1.

For a training set \( \{ y(k) \}_1^N \), ALE finds the latent label \( x(k) \) for \( y(k) \) by first finding \( M \) closest points around \( y(k) \) that are in the basic movements sets \( \bigcup_{i=1}^{N} \{ y_i \}_q \) (line 4), approximating \( y(k) \) as a linear regression of these \( M \) points (line 5), and then keeping this approximation relation in latent space (line 6). This projecting approach is similar to LLE, which preserves the reconstruction relation of a point relative to its neighbors in high dimension into the latent space. However, our approach preserves the high dimensional space reconstruction relationship of a point relative to neighboring points on the axes of the latent space.

Note that the equilibrium point \( q^e \) of the template \( T_r \) indeed corresponds to the origin of the latent space \( x \) by the above construction. The ALE constructs the latent space with the following properties: (1) origin \( 0 \in \mathbb{R}^d \) in the latent space maps to the equilibrium \( y^e \in \mathbb{R}^d \) in the limb joint angles space; (2) points in one latent space axis \( x_i, i = 1, \ldots, d \), map to the limb poses corresponding to one motion primitive in template \( T_r \); and (3) any limb movements can be decomposed onto motion primitives \( \bigcup_{i=1}^{N} \{ y_i \}_q \), namely, an arbitrary point in the limb joint angles space is approximated by a linear combination of motion primitives. By intentionally choosing motion primitives, the latent axes preserves the physical meaning, namely, the coordinates along motion primitive directions.

### 2.4. Mapping from the latent space to the physical space

Once the label \( \hat{x} = \{ \hat{x}_i \} \) is obtained from ALE, we apply GPDM to identify the latent dynamics model (2a) as discussed in Section 2.2. High-dimensional joint angles \( x(k) \) are estimated from \( x(k) \). We denote the training data sets for prediction (2a) and observation (2b) as \( T_p = \{ [x(i-1)]^N, [x(i)]^N \} \) and \( T_o = \{ [y(i-1)]^N, [y(i)]^N \} \), respectively. The latent coordinates are given from the model as

\[
P(\delta x(k)|s(k-1), T_p) \propto \mathcal{N}(\delta x(k)|GP_{s}(s(k-1), T_p)GP_{s}(s(k-1), T_p))
\]

(4)

where \( GP_{s}(s(k-1), T_p) = k^T_{s}(K + \sigma^2_{s}I)^{-1}(\delta x(i)) \) and \( GP_{s}(s(k-1), T_o) = k(s(k-1), s(k-1)) - k^T_{s}(K + \sigma^2_{s}I)^{-1}k_{s}, \) and \( \delta x(i) \) denotes a column vector obtained by stacking \( \delta x(i) \). \( K[p,q] = \sigma^2_{s}e^{-\frac{1}{2}(p-s_p)^T\omega^{-1}(p-s_p)} \) and \( \omega^{-1} \) is from the training data sets. \( k_{s} \) is realized with \( \omega^{-1} = \sigma^2_{s}e^{-\frac{1}{2}(p(k)-s_p)^T\omega^{-1}(p(k)-s_p)} \). The mean value of this distribution is added by \( y^e \) to obtain the estimates of the joint angles.

In the EKF design in Section 4, we need to calculate \( \frac{\partial \delta x(k)}{\partial x(k-1)} \) to obtain the Jacobian matrix of (2a). Noting that in equation (4) only \( k_{s} \) in \( GP_{s}(s(k-1), T_p) \) is function of \( x(k-1) \), we obtain

\[
\frac{\partial \delta x(k)}{\partial x(k-1)} = \frac{\partial GP_{s}(s(k-1), T_p)}{\partial k_{s}} \frac{\partial k_{s}}{\partial x(k-1)}
\]

(5)

Similarly, the joint angles are obtained from equation (2b) as

\[
P(y(k)|x(k), T_o) \propto \mathcal{N}(y(k)|GP_{s}(x(k), T_o), GP_{s}(x(k), T_o))
\]

(6)
where \( GP_\mu(x(k), T_o) = k^T(K + \sigma_n^2 I)^{-1} y \) and \( GP_\Sigma(x(k), T_o) = k(x(k), x(k)) - k^T(K + \sigma_n^2 I)^{-1} k \). The Jacobian of (2b) is the obtained
\[
\frac{\partial y(k)}{\partial x(k)} = \frac{\partial GP_\mu(x(k), T_o)}{\partial k} \frac{\partial k}{\partial x(k)} \tag{7}
\]

3. Rider-bikebot systems

In this section, we use rider-bikebot interactions as an example to illustrate the physical-learning modeling framework.

3.1. Bikebot system configuration

Figure 3 shows the bikebot system configuration. The bikebot is a modified bicycle with augmented steering, pedaling and balancing actuation to understand and study human sensorimotor balancing skills through unstable rider-bicycle interactions (Zhang et al., 2014b). The bikebot is equipped with various sensors, such as IMU and seat force sensors, as shown in the figure. When the actuators are not powered, the bikebot functions the same as a regular bicycle. Figure 4(a) shows the indoor riding experiments and Figure 4(e) shows the outdoor riding experiments.

Figure 5 illustrates the kinematic schematic of the rider-bikebot interactions. The rider's upper-body is considered as an inverted pendulum in three-dimensional (3D) space with its length, mass, and mass moment of inertia denoted as \( h, m_h, \) and \( J_h \), respectively. A ground-fixed inertial...
frame $\mathcal{I}$ ($X, Y, Z$) is defined with the $Z$-axis downwards. A moving frame $\mathcal{R}$ ($x, y, z$) is defined with the $x$-axis along wheel-ground contact points $C_1$ and $C_2$, the $z$-axis along the $Z$-axis, and the origin at $C_2$.

The bikebot roll and yaw angles and steering angle are denoted as $\phi_y$, $\psi$, and $\gamma$, respectively. The gyroscopes on the bikebot frame is tilted by angle $\gamma$ with respect to the $x$-axis. Let $\mathcal{I}_b$ and $\mathcal{I}_h$ denote the rider and bikebot gyroscope frames, respectively. The orientation of the trunk is defined by three Euler angles with the $X$-$Y$-$Z$ ordered rotation from frames $\mathcal{R}$ to $\mathcal{I}_b$: roll angle $\phi_y$ around the $x$-axis, angle $\theta$ around the $y$-axis, and finally self-spinning angle $\phi$ around the $x$-axis (Zhang et al., 2013). The zero-lateral velocity nonholonomic constraint of $C_2$ is considered and the bikebot’s velocity is denoted as $v_{rx}$.

We mainly focus on and present the modeling results for the upper-limb motion, and the lower-limb motion can be similarly obtained. The rider’s upper-limb poses are specified from the trunk orientation. We define the right and the left upper-arm poses by the $Y$-$Z$-$X$ rotation from the trunk frame (i.e., 3 degrees of freedom (3-DOF) shoulder joint) with joint angle sets ($y_1, y_2, y_3$) (left) and ($y_6, y_7, y_8$) (right), respectively. The elbow joints are assumed to be a 2-DOF joint and the right and the left fore-arms are obtained by $Y$-$X$ rotations with joint angle pairs of ($y_4, y_5$) (left) and ($y_9, y_{10}$) (right), respectively. The wrist joint is assumed fixed to the handlebar. Thus, the upper-limb is modeled as a 10-DOF multi-link. Similarly, the joint angles for the lower-limb are denoted as $y_{11}$ to $y_{20}$.

### 3.2. The physical-learning model

#### 3.2.1. Physical model for the bikebot and the rider trunk

We define the generalized coordinate for the trunk as $\mathbf{q}_t = [\phi_y \theta \phi]^T$. Considering $v_{rx}$ and $\psi$ as time-varying model parameters, the bikebot’s generalized coordinate is $\mathbf{q}_b = \phi_y$. The bikebot is considered as a rigid body with mass $m_b$ and mass moment of inertia $I_b$ around the $X$-axis. The bicycle mass center $G$ and the seat position are located at $[l_s \ 0 - h_s]^T$ and $[l_s \ 0 - h_s]^T$, respectively, in $\mathcal{R}$. The bicycle wheelbase is denoted as $l$ and the caster angle as $\eta$. From Yi et al. (2006), the bikebot yaw rate is calculated as

$$\dot{\psi} = \frac{v_{rx} \tan \gamma \ c_y}{l c_{\psi_y}} = \frac{v_{rx} \ c_y}{l c_{\psi_y}} u_s$$

where $u_s = \tan \gamma$ is the steering control input and notation $c_x := \cos x$ ($s_x := \sin x$) is used for angle $x$ in the above equation and throughout the rest of the paper. From equation (8), it is straightforward to approximate and obtain $\dot{\psi} \approx \frac{v_{rx} s_\gamma}{u_s}$.

Following the similar development in Zhang et al. (2013) and Wang and Yi (2015), we take a constrained Lagrange
where trunk roll and pitch motions and the bikebot steering angle and motion during bikebot riding are primarily influenced dimensional latent space. The rider’s upper-limb orientation We capture the 10-DOF upper-limb motion in a low-dimensional latent space, as illustrated in Figure 6. With the state dynamics (10a) and (10b), and observations (11) and (13), we define the state variable ξ = [φh, ϕh, θ, x, y, z]T ∈ R7 and the EKF design is illustrated in Algorithm 2. In the algorithm, matrices Q(k) and R(k) are the covariances of the prediction and observation errors at the kth step. Instead of using constant covariances, we follow the same treatment as Ko and Fox (2009) to update Q(k) and R(k) by equation (4). Particularly since the last d dimensions of ξ are given by the latent variables, only a block matrix of Q(k) is updated as shown in line 3 in the algorithm. Because the observation equations are functions of y, as shown in line 7, a Gaussian process prediction covariance M(k) of y(k) is obtained to calculate the gain K(k). We implement a Bayesian filter to obtain joint angle prediction ŷ.

4. EKF-based rider-bikebot pose estimation

In this section, we present an application of the physical-learning model for rider-bikebot pose estimation. Comparing with the previous studies in Zhang et al. (2014a, 2013), one main advantage of using the physical-learning model is that there is no need for wearable sensors attached to the upper-limb and the bikebot and their derivatives are taken as the input to the latent dynamics. The second interaction is from the geometric constraint and the human anatomical properties that will be presented in the next section.

\[ \mathbf{u}_t = \begin{bmatrix} -\tau_h & \tau_h & \tau_o \end{bmatrix}^T \]

where \( \tau_h \) and \( \tau_o \) are the torques applied by the rider to the trunk in the roll and the pitch directions, respectively.

3.2.2. Learning model for the upper-limb movement.

We capture the 10-DOF upper-limb motion in a low-dimensional latent space. The rider’s upper-limb orientation and motion during bikebot riding are primarily influenced by three motion templates of the bikebot-trunk system: trunk roll and pitch motions and the bikebot steering angle motion and therefore, \( D = 10 \) and \( d = 3 \). For the same reason, we build the three motion primitives by perturbing trunk roll (φh) and pitch (θ) motions and bikebot steering (γ) motion. For the input \( \mathbf{u}_h \) in equation (2), we have

\[ \mathbf{u}_h = \begin{bmatrix} γ(t) & \dot{γ}(t) & \dot{φ}_h(t) & \phi(t) & \dot{ϕ}(t) & \dot{θ}(t) & \dot{θ}(t) \end{bmatrix}^T \]

These are two main interactions between the physical and learning model. First, the orientation angles of the rider trunk and the bikebot and their derivatives are taken as the input to the latent dynamics. The second interaction is from the geometric constraint and the human anatomical properties that will be presented in the next section.

\[ \mathbf{Q}(t) = \begin{bmatrix} C_\gamma & 0 & S_\gamma & 0 & 0 & 0 & 0 \\ 0 & C_\gamma & 0 & S_\gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & C_\gamma & 0 & S_\gamma \\ 0 & 0 & 0 & 0 & 0 & C_\gamma & 0 \\ S_\gamma c_\gamma & c_\gamma c_\gamma & -c_\gamma s_\gamma & 0 & 0 & 0 & 0 \\ c_\gamma s_\gamma & s_\gamma c_\gamma & c_\gamma s_\gamma & 0 & 0 & 0 & 0 \\ S_\gamma s_\gamma & c_\gamma c_\gamma & c_\gamma s_\gamma & 0 & 0 & 0 & 0 \\ c_\gamma s_\gamma & c_\gamma c_\gamma & c_\gamma s_\gamma & 0 & 0 & 0 & 0 \end{bmatrix} \]

\[ = : f_2(q_{st}; \omega_r, \omega_t) \]  \( \text{for} \quad \dot{\omega}_t = \begin{bmatrix} \dot{\phi}_h \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} c_\gamma & 0 & S_\gamma \\ 0 & c_\gamma & 0 \\ 1 & -c_\gamma & 0 \end{bmatrix} \omega_r + \begin{bmatrix} 0 & S_\gamma & c_\gamma \\ 0 & c_\gamma & 0 \\ S_\gamma c_\gamma & c_\gamma S_\gamma & 0 \\ c_\gamma S_\gamma & c_\gamma S_\gamma & 0 \\ S_\gamma c_\gamma & c_\gamma S_\gamma & 0 \\ c_\gamma S_\gamma & c_\gamma S_\gamma & 0 \end{bmatrix} \omega_r \]

The geometric constraints are from the human anatomical property and the rider-bikebot interactions. When riding the bikebot, the rider’s hip sits on the seat and the hands always hold the handlebar. Therefore, the trunk, the upper-limb, and the bikebot frame and steering mechanism form a closed linkage structure. With the known upper-limb poses, we have one vector equation for the left upper-limb formulated as equation (12). In equation (12), \( l_{ul}, h_{ul}, l_u \) and \( l_f \) are the shoulder width and height, upper-arm and forearm lengths, respectively. \( r_i(\gamma) \) is the position vector from the bikebot seat to the left handlebar position in \( \mathcal{R} \), and \( \mathcal{R}(\gamma) \) represents the 3D rotational matrix around the i-axis with angle \( \gamma, i = x, y, z \). We obtain the similar constraint for the right upper-limb \( z_{2r} = 0 \) and thus, we have constraints

\[ \begin{bmatrix} z_2(\psi_h, \theta, \phi, y) \\ z_{2r} \end{bmatrix} = 0 \in \mathbb{R}^6 \]

Where to include the lower-limb pose estimation, we augment the latent variable \( x \in \mathbb{R}^6 \) with the additional three elements \( x_4-x_6 \) and output variable \( y \in \mathbb{R}^{20} \) with \( y_{11}-y_{20} \). The input to
the lower-limb latent dynamics is the pedal crank angle and angular rate. The geometric constraint for the lower-limb is constructed as the linkage formed by the lower-limb, the seat and the pedal (Chen et al., 2013). We omit the detailed discussion here.

5. Experiments

5.1. Experimental setup

Human subjects are recruited to conduct the indoor and outdoor bikebot riding experiments. Five healthy and experienced bicycle riders (four male and one female, ages 27 ± 3, heights 176 ± 5 cm, and weights 68 ± 10 kg) were recruited to conduct both the indoor and outdoor experiments. The duration for each riding experiment run was around 2 minutes. The subjects were first asked to get familiar with the riding bikebot before experimental data were taken. All subjects signed their informed consent using a protocol approved by the Institutional Review Board (IRB) at Rutgers University. Before experiments, all subjects’ biomechanic parameters and wearable sensor locations were measured. In experiments, we used three load cells mounted under the seat to roughly measure the trunk mass. The heights of the center of the mass position are assumed to be proportional to trunk length and their values are also estimated through OpenSim, a musculoskeletal modeling and dynamic simulation software package (available at http://opensim.stanford.edu). The model parameters (limb length, trunk mass, etc.) for each human subject are used in his/her pose estimation designs.

For the bikebot system, steering angle γ and velocity v_{rx} are measured by encoders. One IMU (model 605 from Motion Sense Inc.) was mounted on the bikebot and another on the human trunk. We only use the tri-axial gyroscope measurements from these IMUs. A vision-based motion capture system (8 Bonita cameras from Vicon Inc.) is used to provide ground truth in indoor experiments. For outdoor experiments, a camera is mounted on the bikebot handlebar and a set of gyroscope/marker pairs are mounted on the upper- and lower-limb to provide the ground truth through vision-inertial fusion algorithms. A real-time embedded system (CompactRIO 9082 from National Instruments Inc.) is used to sample and process all sensor measurements.
at the frequency of 100 Hz. The gyro-balancer actuation is used only in the outdoor experiments to generate perturbation torques to excite the rider’s responses.

5.2. Experimental results

We conduct indoor bikebot riding experiments to demonstrate the performance of the EKF-based pose estimation scheme. To construct isolated latent space coordinate axes in the ALE algorithm, we used the human movement data sets collected by riding a stationary bikebot and assumed that the riders use similar upper-limb and trunk movements and strategies when riding both the moving and stationary bikebots. We fixed the bikebot roll angles at a set of values and the subjects were asked to perturb their trunk motions in roll and pitch directions and move their steering angles independently in separate experiment runs. We collected the motion data in these experiments to construct the ALE reduction axes. By doing so, we avoid the challenges of perturbing the subjects and exciting the isolated body segment motions while riding and balancing a moving bikebot. The joint-angle prediction performance presented later in this section confirms the feasibility of using this method to construct the ALE axes.

We collect camera-based ground truth measurements of two rounds of bikebot riding experiment to obtain the learning model and the physical model parameters. The training data set contains a total of 400 pairs of input and output points. For testing and validation, we use separate experiments to compute the EKF estimation and compare them with the ground truth. Figure 7 shows a one-minute pose estimation of the bikebot and the upper-limb. We chose a group of five joint angles ($y_1$-$y_5$) for the left upper-limb as a representative group of all 20 limb joint angles (the right upper-limb is similar and the lower-limb results are more regular than those of the upper-limbs due to the periodic pedaling motion). The EKF estimates follow closely with the ground truth for the bikebot and human body segment orientations. Figure 8 further shows the estimation errors of $y_1$, $y_2$ and $y_4$ by both the EKF scheme and only the machine-learning-based prediction model. Clearly, the use of the integrated physical-learning model improves the estimation performance.

To demonstrate the performance for all ten upper-limb joint angles, Table 1 illustrates the estimation errors obtained by the EKF scheme with three model reduction approaches, including ALE, PCA and LLE. Each element in the table is the mean value with one standard deviation of the root mean square (RMS) errors obtained by six rounds of upper-limb motion over a 20-second period. The input data of the model are obtained from vision-based motion capture measurements. For most of the upper-limb joint angles (except $y_3$), the mean values of the RMS errors from the ALE are smaller than the corresponding results from the PCA method. The sum of the RMS errors of all the upper-limb joint angles (i.e., $y_1$-$y_{10}$) from the ALE algorithm (sum RMS: 36.6) is also less than those of the other two approaches (sum RMS: PCA: 40.9 and LLE: 57.2). From the viewpoint of statistical significance, the t-test of the mean value of the RMS errors over the ten joint angles confirms significantly smaller errors under the ALE than those under the PCA predictions ($p = 0.01$) although the t-test of the standard deviation of the RMS errors does not show a significant difference between these two methods ($p = 0.43$). For the purpose of comparison, we also list the estimation performance reported in Zhang et al. (2014a) by using the wearable IMUs attached to each of the upper-limb segment in the same experiment. The performance of the ALE-GPDM model-based prediction is comparable to the results obtained using the wearable sensors. The performance of the outdoor estimation results is similar to those of the indoor experiments and we omit the details.

Figure 9 shows the latent dynamics trajectory in the latent space. We plot both the latent dynamics for the upper-limb motion (Figure 9(a)) and the lower-limb motion (Figure 9(b)) during the indoor experiments. In these plots, we used a constant value $\sigma_z = 0.1$ for the preset observation noise variance. Because of the similar anatomical structures, like the upper-limb, a three-dimensional ($d = 3$) latent space is chosen for the lower-limb pedaling motion. Due to the periodic pedaling movement, the trajectory of the latent dynamics ($x_4$-$x_5$) of the lower-limb is periodic, while the upper-limb trajectory in the latent space ($x_1$-$x_3$) is non-periodic.
Therefore, the pose estimation scheme using the learning model can predict both the periodic and non-periodic human movements. A learning model such as GPDM has also been used and reported to capture other non-periodic human motion such as dancing and playing golf, etc. (Wang et al., 2008).

The rider-bikebot interaction is obtained through a set of geometric constraints and these interaction constraints are...
shared by different human subjects. Built on this observation, we use the latent dynamic model built from one subject to estimate the joint angles of the other subjects during bikebot riding experiments. Tables 2 and 3 list the mean and one-standard deviation of the RMS errors of the estimated joint angles for the upper-limb ($y_1$ to $y_{10}$) and the lower-limb ($y_{11}$ to $y_{20}$), respectively, for five human subjects in indoor and outdoor experiments. From these results, we clearly see that the latent dynamic model captures the rider-bikebot interactions and predicts other riders’ riding motion, although the model is built on one rider.

We further demonstrate the estimation performance for bikebot riding under perturbation. In experiments, the human rider was not informed when and how the perturbation torques would be generated and applied by the gyro-balancer. Under the perturbed external torque, the rider applies recovering strategies (e.g. trunk movements and steering) to keep the bikebot stable. Figure 10 shows the experimental results of the applied perturbation torque and the rider’s reactions. The perturbation starts at around 10 seconds and clearly the rider uses steering to react and keep balancing the platform. Figure 10(a) shows the perturbed...
Table 2. RMS errors (in deg) of the estimated upper-limb angles over a 1-minute period in the indoor and outdoor experiments.

<table>
<thead>
<tr>
<th>Indoor</th>
<th>(\psi_h)</th>
<th>(\theta)</th>
<th>(\phi)</th>
<th>(\psi_b)</th>
<th>(y_1)</th>
<th>(y_2)</th>
<th>(y_3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>6.4 ± 1.8</td>
<td>1.5 ± 0.2</td>
<td>7.6 ± 3.1</td>
<td>0.6 ± 0.2</td>
<td>2.6 ± 0.3</td>
<td>4.2 ± 0.9</td>
<td>5.9 ± 0.9</td>
</tr>
<tr>
<td>Outdoor</td>
<td>7.7 ± 1.0</td>
<td>2.0 ± 0.7</td>
<td>9.6 ± 0.4</td>
<td>0.8 ± 0.2</td>
<td>2.9 ± 0.3</td>
<td>4.9 ± 0.7</td>
<td>6.4 ± 0.9</td>
</tr>
<tr>
<td>(y_4)</td>
<td>(y_5)</td>
<td>(y_6)</td>
<td>(y_7)</td>
<td>(y_8)</td>
<td>(y_9)</td>
<td>(y_{10})</td>
<td></td>
</tr>
<tr>
<td>Indoor</td>
<td>5.3 ± 0.8</td>
<td>5.1 ± 0.7</td>
<td>2.8 ± 0.5</td>
<td>4.4 ± 1.2</td>
<td>3.9 ± 0.2</td>
<td>5.6 ± 0.1</td>
<td>3.0 ± 0.4</td>
</tr>
<tr>
<td>Outdoor</td>
<td>6.0 ± 2.0</td>
<td>5.2 ± 0.5</td>
<td>2.9 ± 0.4</td>
<td>4.7 ± 0.8</td>
<td>4.5 ± 1.3</td>
<td>6.3 ± 1.1</td>
<td>3.1 ± 0.5</td>
</tr>
</tbody>
</table>

We conduct additional experiments to further demonstrate the application of the GPDM with a comparison of the GP model to human motion with a greater variance and fewer constraints. Figures 11(a) and 11(b) illustrate new riding experiments that were conducted recently to extend the upper-limb motion with large flexibility. In the new experiments, instead of firmly grasping the steering handlebar, the subject was asked to ride and steer the bicycle with their hands holding a pair of elastic strings fixed to the handlebar, see Figure 11(b). The subject moved his limb arbitrarily for multiple riding experiment runs inside a laboratory, see Figure 11(a).

With this setup, the rider’s hands can flexibly move along the \(xx'\) and \(zz'\) directions during riding as shown in Figure 11(b). This gives greater variances and fewer constraints than the regular riding style. The less constrained hand positions can be observed by the dynamically changing the handlebar steering angle \(\gamma\) and the actual hand steering angle \(\gamma'\) calculated by the marker positions on the human hands. Figure 12(a) shows the differences between \(\gamma\) and \(\gamma'\) and these differences generate the large variance in upper-limb joint angles.

5.3. Discussion

From Figure 7, we notice that the upper-limb motion during bikebot riding is restricted to within a range of 30 degrees. These restrictions are coming from the fact that the human hands are fixed on the steering handlebar while the other ends of the upper-limb are connected to the human trunk. With this observation, it might be possible to obtain the upper-limb poses through a simple Gaussian process (GP) model to map from the trunk and steering angle data set, or even by only using geometric constraints \(z_2\) and dynamic constraints \(z_1\). We compare the prediction results of the upper-limb poses by using the GPDM and a simple GP model. Table 5 shows the comparison results for the upper-limb joint angles estimation under the GP model, and the only constraint-based approach. Comparing with similar indoor prediction results (by the GPDM shown in Table 2), it is clear that the GPDM and the GP approaches result in a similar performance (sums of RMS errors are both 42.8.). Indeed, the t-test of the mean value of the RMS errors over the ten joint angles show no significant difference between the GPDM and the GP model predictions \((p = 0.49)\). From Table 5, we also observe that with only the geometric and dynamic constraints, it is difficult to accurately obtain the pose estimation (i.e., large estimation errors) due to limb motion redundancy.
Table 4. RMS errors (in deg) of the estimated upper-limb angles over a 25-second perturbed period in the outdoor experiments.

<table>
<thead>
<tr>
<th>$\psi_{h}$</th>
<th>$\theta$</th>
<th>$\phi$</th>
<th>$\psi_{b}$</th>
<th>$y_1$</th>
<th>$y_2$</th>
<th>$y_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.7 ± 1.1</td>
<td>2.6 ± 0.7</td>
<td>5.1 ± 1.4</td>
<td>0.3 ± 0.1</td>
<td>3.7 ± 1.6</td>
<td>4.5 ± 1.9</td>
<td>6.3 ± 1.1</td>
</tr>
<tr>
<td>$y_4$</td>
<td>$y_5$</td>
<td>$y_6$</td>
<td>$y_7$</td>
<td>$y_8$</td>
<td>$y_9$</td>
<td>$y_{10}$</td>
</tr>
<tr>
<td>7.2 ± 2.0</td>
<td>4.5 ± 0.8</td>
<td>4.2 ± 1.2</td>
<td>4.9 ± 2.0</td>
<td>6.2 ± 2.1</td>
<td>6.0 ± 1.7</td>
<td>4.4 ± 1.0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$y_1$</th>
<th>$y_2$</th>
<th>$y_3$</th>
<th>$y_4$</th>
<th>$y_5$</th>
<th>$y_6$</th>
<th>$y_7$</th>
<th>$y_8$</th>
<th>$y_9$</th>
<th>$y_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>GP</td>
<td>2.8 ± 1.4</td>
<td>3.9 ± 1.7</td>
<td>5.3 ± 1.7</td>
<td>6.8 ± 3.0</td>
<td>4.4 ± 1.6</td>
<td>2.9 ± 0.7</td>
<td>2.7 ± 1.0</td>
<td>4.7 ± 1.4</td>
<td>6.2 ± 3.2</td>
</tr>
<tr>
<td>Constraint</td>
<td>11.7 ± 2.5</td>
<td>6.3 ± 2.7</td>
<td>13.3 ± 1.6</td>
<td>10.0 ± 1.9</td>
<td>9.3 ± 1.4</td>
<td>12.9 ± 3.2</td>
<td>7.0 ± 1.6</td>
<td>12.9 ± 4.6</td>
<td>8.3 ± 3.3</td>
</tr>
</tbody>
</table>

Table 5. RMS errors (in deg) of the estimated upper-limb angles over a 1-minute period (indoor experiments) by the GP model and the geometric and dynamic constraints only.

Fig. 11. (a) Riding a bicycle with two hands on elastic strings tied on the steering handlebar. (b) A close view of the hands on the elastic strings with flexibility along the $xx'$ and $zz'$ directions.

Gaussian noises. The use of the ALE reduction in GPDM generates more accurate results in almost all joint angles than those obtained under the PCA reduction method (sums of RMS errors are 77.8 and 103.2, respectively). The test results of the mean value and standard deviation of the RMS errors over the ten joint angles are significantly smaller under the GPDM-ALE than under the GPDM-PCA predictions ($p = 0.01$ and $p = 0.05$, respectively). We also see the significantly superior performance of the GPDM ALE approach compared to the GP model (sums of RMS errors are 77.8 and 213, respectively). The GP model and the only constraint-based approach cannot accurately predict most of the upper-limb joint angles. For example, Figures 12(b) to 12(f) show the joint angle estimations $y_1$-$y_5$ of one subject riding experiment under the GPDM-ALE, the GP model and the only constraint-based approach. We clearly see that the GP model and the only constraint-based scheme cannot capture the joint motion accurately in these experiments, particularly for joint angles $y_3$-$y_5$.

During the latent variable initialization process, we chose the preset observation noise value $\sigma_\epsilon = 0.1$ of the variance of $P(\mathbf{X}\mid \mathbf{X})$ during training. To illustrate the influence of the value $\sigma_\epsilon$ on the estimation results, Figure 13 shows the RMS errors of the ten predicted joint angles of the upper-limb motion during bikebot riding. From the results in Figure 13, we clearly see that the prediction performance is almost kept at a constant level until the $\sigma_\epsilon$ values increase to 100. Figure 14 further illustrates the comparison of the trajectories of the latent space variables $(x_1, x_2, x_3)$ for the upper-limb motion initially obtained ALE (Figure 14(a)) and with labeling uncertainty at $\sigma_\epsilon = 0.1$ (Figure 9(a)) and $\sigma_\epsilon = 100$ (Figure 14(b)). These plots clearly show that the larger values of $\sigma_\epsilon$ result in smoother latent variable trajectories. The learning model with large values of $\sigma_\epsilon$ possibly tends to generate worse prediction results, as shown in Figure 13 for $\sigma_\epsilon = 100$. Noting that the values of the latent space variables $(x_1, x_2, x_3)$ generated by the ALE algorithm are all within $(−2, 2)$, we therefore chose $\sigma_\epsilon = 0.1$ in our implementation.

The EKF estimation convergence of $\psi_{h}$ and $\phi_{h}$ partially comes from constraint $z_1 = 0$ in equation (11) obtained from systems dynamics. To demonstrate the impact of the dynamic and geometric constraints on the states estimation,
Fig. 12. Upper-limb pose estimation comparison among various modeling approaches for new experiments: (a) handlebar steering angle $\gamma$ and human hands steering angle $\gamma'$, (b) $\gamma_1$, (c) $\gamma_2$, (d) $\gamma_3$, (e) $\gamma_4$, (f) $\gamma_5$.

Table 6. RMS errors (in deg) of the estimated upper-limb angles by various models for bicycle riding with elastic strings.

<table>
<thead>
<tr>
<th></th>
<th>$\gamma_1$</th>
<th>$\gamma_2$</th>
<th>$\gamma_3$</th>
<th>$\gamma_4$</th>
<th>$\gamma_5$</th>
<th>$\gamma_6$</th>
<th>$\gamma_7$</th>
<th>$\gamma_8$</th>
<th>$\gamma_9$</th>
<th>$\gamma_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>GPDM/ALE</td>
<td>11.1 ± 3.1</td>
<td>3.7 ± 0.3</td>
<td>14.3 ± 7.1</td>
<td>8.5 ± 2.3</td>
<td>4.7 ± 0.9</td>
<td>7.4 ± 0.7</td>
<td>4.3 ± 1.0</td>
<td>7.5 ± 1.4</td>
<td>9.5 ± 0.6</td>
<td>6.8 ± 1.4</td>
</tr>
<tr>
<td>GPDM/PCA</td>
<td>15.2 ± 6.3</td>
<td>5.9 ± 1.5</td>
<td>17.8 ± 7.0</td>
<td>8.2 ± 1.5</td>
<td>3.1 ± 0.6</td>
<td>13.8 ± 1.4</td>
<td>6.7 ± 3.4</td>
<td>14.6 ± 2.3</td>
<td>11.7 ± 1.8</td>
<td>6.2 ± 0.5</td>
</tr>
<tr>
<td>GP</td>
<td>17.1 ± 5.6</td>
<td>8.2 ± 2.7</td>
<td>44.3 ± 32.0</td>
<td>33.3 ± 8.6</td>
<td>10.9 ± 3.7</td>
<td>22.0 ± 10.9</td>
<td>10.4 ± 2.9</td>
<td>11.5 ± 3.0</td>
<td>31.0 ± 6.9</td>
<td>24.3 ± 15.2</td>
</tr>
<tr>
<td>Constraint</td>
<td>12.8 ± 3.3</td>
<td>38.7 ± 2.0</td>
<td>22.6 ± 5.4</td>
<td>32.0 ± 1.6</td>
<td>8.1 ± 2.5</td>
<td>15.2 ± 2.5</td>
<td>13.5 ± 0.9</td>
<td>34.2 ± 7.3</td>
<td>34.9 ± 3.0</td>
<td>11.6 ± 2.3</td>
</tr>
</tbody>
</table>

we relax either of two constraints and compare the estimation results with those under full constraints. Figure 15 shows the mean and standard deviation of the pose estimation with and without constraints (11) or (13). We observe that the dynamic constraint (11) mainly influences the accuracy of $\phi_b$ estimation and the geometric constraint (13) mainly affects the estimation accuracy of the trunk and the upper-limb joint angles. This observation can be explained by the sensitivity calculation of these constraints with respect to each state variable. Moreover, the observability matrix of the EKF state dynamics with the geometric constraint always has a full rank (we omit the calculation here) and therefore, the EKF design satisfies the necessary convergence condition.

To validate the bikebot-trunk dynamics, Figure 16(a) shows the comparison of the values of torque $\tau_h$ calculated by equation (9) with these measured by the seat force sensor in a straight-line riding experiment. The measured torque values match the model predictions. One attractive

Fig. 13. Prediction error of the model trained with different latent space observation noise uncertainty levels $\sigma_\hat{x}$. 

The International Journal of Robotics Research 35(12)

Fig. 14. Trajectories of the latent space variables $(x_1, x_2, x_3)$ for the upper-limb motion using different values of $\sigma_x$: (a) directly from the ALE algorithm, (b) with a labeling uncertainty $\sigma_x = 100$.

Fig. 15. The pose estimation errors with and without constraints.

The property of constraint $z_1 = 0$ in equation (11) compared to dynamic equations (9) is that there is no need to use force or torque measurements. To validate constraint (11), Figure 16(b) plots the means and one standard deviation of the values of $z_1$ over one-minute riding experiments by five human subjects. Clearly, the values of $z_1$ are around zero and the variations can be captured through the noise term in the EKF design.

To understand how the size of the training data set affects the estimation performance, Figure 17 shows the sum of the root mean square of the estimation errors of all ten upper-limb joints over the number of training data points. We notice that the estimation performance is consistent since the number of used training data points is greater than a threshold (e.g. around 200 data points). Moreover, the proposed EKF design is not sensitive to the choice of the initial value of the state variables. For example, we use two different means to initialize the state variables: one is to use the average value for each variable to initialize each variable, and the other is to use accurate measurements. The resulting EKF estimation performance (RMS errors) are listed in Table 7 for a one-minute riding experiment. No significant performance differences are observed between these two cases.

6. Conclusion

We have presented an integrated physical-machine learning modeling approach for physical human-robot interactions. In this framework, physical models were used to capture
Table 7. RMS errors (in deg) of each estimated angle over a 1-minute experiment run with different initial values.

<table>
<thead>
<tr>
<th>Init. val. φ1</th>
<th>φ2</th>
<th>φ3</th>
<th>φ4</th>
<th>φ5</th>
<th>φ6</th>
<th>φ7</th>
<th>φ8</th>
<th>φ9</th>
<th>φ10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approx. 5.3</td>
<td>1.7</td>
<td>5.3</td>
<td>0.5</td>
<td>2.4</td>
<td>3.6</td>
<td>5.1</td>
<td>5.3</td>
<td>4.4</td>
<td>2.0</td>
</tr>
<tr>
<td>Measure</td>
<td>5.0</td>
<td>1.5</td>
<td>4.9</td>
<td>0.4</td>
<td>2.3</td>
<td>3.0</td>
<td>5.1</td>
<td>4.6</td>
<td>4.2</td>
</tr>
</tbody>
</table>

Fig. 17. The RMS of all estimated ten upper-limb joint angles versus the training data points.

robot and human trunk dynamics while a machine-learning-based model was used to describe the highly redundant, high-DOF limb motion in low-dimensional latent space. Coupled interconnections and constraints were built between these two models. For dimensionality reduction, we have presented a novel axial linear embedding (ALE) method. The main advantage of the proposed modeling approach is to represent the pHRI on the low-dimensional manifold and to enable the application of existing estimation and control techniques to complex pHRI. The new ALE reduction algorithm also preserves the physical meaning of the latent variables of the learning model. We have illustrated and demonstrated the modeling framework through bikebot riding experiments with applications to estimating human and bikebot poses. We also compared the limb pose estimation performance of the proposed GPDM-ALE and the simple GP model. The comparison of the results demonstrated that the GPDM-ALE outperformed the GP model for physical human-bikebot interactions with a large variance and fewer constraints. The attractive properties of using the proposed model are that even without any wearable sensors attached to the limb segments, the model-based estimation scheme accurately predicted the limb motion.

Acknowledgements

The authors thank Professor K. Li of Rutgers University and Dr G. Yue of the Kessler Rehabilitation Institute for their helpful discussions. We are also grateful to M. Trkov, P. Wang, and other members of the Robotics, Automation and Mechatronics (RAM) Lab at Rutgers. The authors would also like to thank the two anonymous reviewers for their valuable comments and suggestions for improving the quality of the paper.

Funding

This work was partially supported by the US National Science Foundation (grant numbers CMMI-1334389 and CMMI-0954966) (J. Yi); the National Natural Science Foundation of China (grant numbers 61428304 and 61403307); award GZKF-201404 from the State Key Lab of Fluid Power and Mechatronic Systems Zhejiang University, China (J. Yi, Y. Zhang and T. Liu); and a fellowship from the Chinese Scholarship Council (K. Chen).

References


