Dynamical Simulation Priors for Human Motion Tracking

Marek Vondrak,Leonid Sigal, *Member, IEEE,* and Odest Chadwicke Jenkins, *Member, IEEE*

Abstract—We propose a simulation-based dynamical motion prior for tracking human motion from video in presence of physical ground-person interactions. Most tracking approaches to date have focused on efficient inference algorithms and/or learning of prior kinematic motion models; however, few can explicitly account for *physical plausibility* of recovered motion. Here, we aim to recover physically plausible motion of a single articulated human subject. Towards this end, we propose a full-body 3D physical simulation-based prior that explicitly incorporates a model of human dynamics into the Bayesian filtering framework. We consider the motion of the subject to be generated by a feedback "control loop" in which Newtonian physics approximates the rigid-body motion dynamics of the human and the environment through the application and integration of interaction forces, motor forces and gravity. *Interaction forces* prevent physically impossible hypotheses, enable more appropriate reactions to the environment (*e.g.*, ground contacts) and are produced from detected human-environment collisions. *Motor forces* actuate the body, ensure that proposed pose transitions are physically feasible and are generated using a motion controller. For efficient inference in the resulting high-dimensional state space, we utilize an exemplar-based control strategy that reduces the effective search space of motor forces. As a result, we are able to recover physically-plausible motion of human subjects from monocular and multi-view video. We show, both quantitatively and qualitatively, that our approach performs favorably with respect to Bayesian filtering methods with standard motion priors.

Index Terms—articulated tracking, human pose tracking, human motion, physical simulation, physics-based priors, Bayesian filtering, particle filtering.

1 Introduction

We consider the problem of physically plausible human motion tracking from video. Although the area of articulated pose tracking has seen many advances, the general problem of tracking the 3D motion of a person in typical environments from monocular image observations remains a challenge. High dimensionality of human pose, variability in imaging conditions, appearance and clothing are but some of the issues that need to be addressed. Most prior approaches to tracking have concentrated on developing search methods and motion priors that allow efficient inference in this high-dimensional pose space. However, physical realism of such motion priors and plausibility of the recovered motion remains an open problem. As a result, many existing methods suffer from visually distinct and physically implausible artifacts, including foot skate, out-of-plane rotations and jitter. With these concerns in mind, **we propose a method for incorporating full body physical simulation for prediction within the probabilistic tracking framework of Bayesian filtering**.

Dynamical simulation has a large body of existing work in animation [4], [16], [19], [21], [31], [33], [39], [60], [63] and robotics [10], [28], [47], [59] and is now a commodity technology. Simulation allows to computationally account for various physical and biomechanical factors that affect human motion, *e.g.*, a person's mass, gravity, interaction with the ground plane, friction, self-collisions and physical disturbances. Our goal is to build a tracking system that can take advantage of predictions based on such simulations so that the search for poses can be biased towards physically valid interpretations, resulting in more accurate and realistic performance.

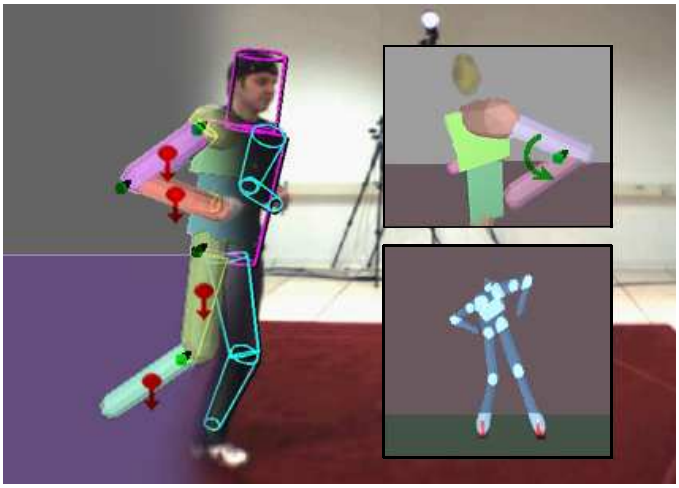

Fig. 1. **Incorporating physics-based dynamic simulation with joint actuation and dynamic interaction into Bayesian filtering**. The figure's motion is determined by its dynamics, actuation forces at joints (top) and surface interaction at contacts (bottom).

Traditionally, physical dynamics has been approximated in motion priors only indirectly by enforcing temporal coherence

- *M. Vondrak and O. C. Jenkins are with Department of Computer Science, Brown University, Providence, RI, .*
  E-mail: { marek, cjenkins }@cs.brown.edu
- *L. Sigal is with Disney Research Pittsburgh.*
  E-mail: lsigal@disneyresearch.com

[13] or by learning statistical models from reference human motion capture data, as in [52]. Motion capture data can be thought of as a snapshot of the dynamics that occurred at the time when the motion was captured and within a given environment where it was performed (*e.g.*, typically on a non-compliant level surface/ground). Such motion capture-based priors are limited to a specific motion, or class of motions, subject to the environment at the time of collection and, consequently, have difficulty generalizing to new environments (*e.g.*, non-level ground or stairs). In addition, because the underlying physical phenomena are encoded only indirectly through statistical relations over body kinematics, pose samples produced by such priors are not guaranteed to be physically valid. As such, kinematic priors can not ensure that it is physically possible to move the body from the current pose to the proposed pose.

To explicitly address realism of the poses generated by a kinematic model we suggest to employ physical simulation. Predictions made by our physics-based prior can then be seen as results of a post-filtering process built on top of the more traditional kinematic prior model. This post-filtering process takes a kinematic pose produced by the kinematic model ("desired pose") as an input and outputs a physically feasible pose, close to the desired pose, by running physical simulation of the motion of the human body, guiding the body's current pose towards the desired pose. A motion controller is responsible for this guidance, through application of motor forces exerted on the body, subject to biomechanical constraints and constraints in the environment. We model the human body as an actuated articulated structure composed of three-dimensional rigid body segments connected by joints whose motion is determined by the mass properties, gravity, interactions among the segments or with the environment and actuation of the joints by the motion controller. In order to facilitate a better understanding of this model and promote use of physical simulation for tracking, we have made the source code of our controller and the simulation-based prior available on our project website http://brown-robotics.org/wp/projects/current/dynamical_tracking/.

Our simulation-based approach has a number of benefits compared to the pure kinematic approach: (1) predicted poses are implicitly biased towards physically plausible interpretations and (2) reasonable predictions can be made even when there is not enough training data available due to the direct incorporation of laws of dynamics. We chose to model motor forces by using a motion controller, because doing so allows us to avoid an explicit inference over motor force trajectories. Consequently, in tracking, we only need to infer the dynamic state of the articulated body and information required for motion control, keeping the dimensionality of the state space manageable.

As a more generic but less tractable alternative, one could re-parameterize the body motion completely in terms of motor forces and avoid the use of motion controller and the kinematic motion prior. However, this parameterization would require inference and priors over the force trajectories. Such inference would be particularly problematic due to the high-dimensional, discontinuous, and nonlinear nature of the space of motor forces. Priors over force trajectories are notoriously hard to characterize, in part, because obtaining such trajectories, in general, requires specialized equipment (*e.g.*, force plates and motion capture data, exception being [6]). Due to sensitivity of measurements the obtained trajectories are also numerically prone to errors [22], [36]. In addition, these trajectories are intimately dependent on the terrain, speed of execution, muscle tone, and even age of the subject [12]. Consequently, characterizing and studying such force trajectories is a source of active research in biomechanics [55].

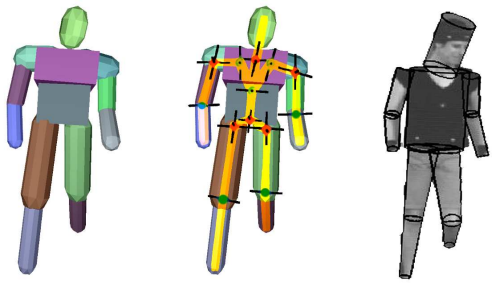

Fig. 2. **Figure model**. 31 degree-of-freedom (DOF) human model with collision geometries of the figure segments (left), the joints and skeletal structure (middle) and the visual representation in an image projection (right). Most joints have 3 DOFs, except for the knee and elbow joints (1 DOF), spine joint and the clavicle joints (2 DOFs) and the root joint (6 DOFs).

We present results demonstrating efficacy of our physics-based prior for tracking. We compare the performance of our method with other commonly used kinematic priors, yielding favorable performance under the effects of dynamic human-environment interactions occurring in monocular and multi-view video. We qualitatively and quantitatively demonstrate that the performance of our physics-based prior produces better tracking accuracy than standard smooth or kinematic exemplar-based priors and is able to better generalize to certain new environments and physical interactions.

2 Related Work

There has been a vast amount of work in computer vision in the past 10-15 years on articulated human motion tracking. We focus on a subset of relevant approaches here and refer reader to [17], [29], [34] for more complete review of the literature.

Most approaches to human tracking to date [2], [13], [44] have concentrated on development of efficient inference methods that are able to handle the high-dimensionality of the human pose. Generative methods typically propose to either learn a low-dimensional embedding of the high-dimensional kinematic data and then attempt to solve the tracking problem in this more manageable low-dimensional space [52], or, alternatively, propose the use of prior models to reduce effective search space in the original high-dimensional space [13]. More recent discriminative methods attempt to map directly from image features to the 3D articulated pose from either monocular imagery [40], [45] or features obtained from multiple views [18].

Producing smooth and accurate tracking results remains a challenge, especially from monocular imagery. In particular, many of these efforts do not address physical plausibility of estimates and often result in recovered motions that violate the constraints imposed on the body by the world or environment (producing out-of-plane rotations and foot skate). Such artifacts can be attributed to the general lack of physically plausible priors [7], [8] which can account for static and/or dynamic balance and ground-person-object interactions.

Recently, priors that directly constrain kinematics with geometric constraints imposed by the environment have been introduced [37], [38]. While shown to be effective, these prior models can only constrain the location of the body segments with respect to the environment. For example, such models can encode a constraint that feet should not penetrate the ground plane [37] or that feet or hands must be in some fixed configuration (as dictated by the environment) with respect to one another [38]. Such geometric priors are not able, however, to allow dynamically plausible environmental interactions, *e.g.*, encode that feet must be in contact with the ground in such a way as to support the resulting motion, *etc*.

The computer graphics and robotics community, on the other hand, has been very successful in developing realistic physical models of human motion. These models for the most part have only been developed and tested in the context of synthesis (*i.e.*, animation [16], [19], [31], [33], [39], [60], [62], [61], [57]) and humanoid robotics [10], [28], [47], [59].

The key benefit of physics-based models in graphics and robotics has been shown to be the ability to use these models to plausibly re-target the original kinematic motions to other environments [33], dynamic interactions with the environment [57], skeletal dimensions/proportions [9] and temporal executions [27]. We conjecture that the use of similar models in tracking would allow equally effective generalizations, beyond the scope of pure kinematic prior models. To this end, we propose a full body physics-based dynamical model as a motion prior, for tracking, that accounts for physically plausible environmental interactions.

Earliest work on integrating physical models with vision-based tracking can be attributed to influential work by Metaxas *et al.* [30] and Wren *et al.* [56]. In both [30] and [56] a Lagrangian formulation of the dynamics was employed, within the context of a Kalman filter, for tracking of simple (no contact) upper body motions using segmented 3D marker [30] or stereo [56] observations. In contrast, we incorporate full body human dynamical simulation into a Particle Filter, suited for multi-modal posteriors that commonly arise from ambiguities in monocular imagery. More recently, Brubaker *et al.* [7], [8] introduced a low-dimensional biomechanically-inspired model that accounts for human lower-body walking dynamics. The low-dimensional nature of the model [7], [8] facilitated tractable inference; however, the model, while powerful, is inherently limited to modeling dynamics of walking motions in 2D and resorts to conditional kinematics to allow tracking of walking motions in 3D and allow turning.

In this work, we introduce a more general full-body model that can potentially model a large variety of human motions. However, the high-dimensionality of our model makes direct inference using standard techniques (*e.g.*, particle filtering) challenging. Consequently, we make use of an exemplar-based prior for the dynamics to limit the effective search space and allow tractable inference in this high-dimensional space. Exemplar-based methods similar to ours have been successfully used for articulated pose estimation in [40], [43], [52], dynamically adaptive animation [63], and humanoid robot imitation [20].

Our exemplar-based prior, discussed in the previous paragraph, can be thought of as an incremental trajectory controller [21], where joint angle trajectories are defined on a frame-by-frame basis from a database of motion capture data. As such, our method also relates to a rich literature on controller design, [42]. While the use of our controller is dictated by simplicity and convenience, other controllers can also be used in this context to produce physically plausible motion. For example, one can use a set of key-poses with proportional derivative (PD) control [19], [61], [62], a learned low-dimensional controller [39], or a combination of controllers [15] that individually deal with properties of the desired motion, *e.g.*, balance (using Zero Point Moment [11] or otherwise [48]). Note that in such cases one would typically need to infer full parameters of the controller [7], [8].

2.1 Background: Human Tracking

Tracking, including human motion tracking, is most often formulated probabilistically using a Bayesian filter formulation [14]. In computer vision literature such filters are usually implemented using a Particle Filter (PF). In PF the *posterior*, $p(\mathbf{x}_f|\mathbf{y}_{1:f})$, where $\mathbf{x}_f$ is the state of the body at frame f and $\mathbf{y}_{1:f}$ is the set of observations up to and including the frame f, is approximated using a set of (typically) weighted samples/particles and is computed recursively,

$$\underbrace{p(\mathbf{x}_f|\mathbf{y}_{1:f})}_{\text{Posterior at } f} \propto \underbrace{p(\mathbf{y}_f|\mathbf{x}_f)}_{\text{Likelihood}} \underbrace{\int \overbrace{p(\mathbf{x}_f|\mathbf{x}_{f-1})}^{\text{Temporal Prior}} \overbrace{p(\mathbf{x}_{f-1}|\mathbf{y}_{1:f-1})}^{\text{Posterior at } f-1} \mathrm{d}\mathbf{x}_{f-1}}_{\text{Predictive Density}}.$$

In this formulation, $p(\mathbf{x}_{f-1}|\mathbf{y}_{1:f-1})$ is the posterior from the previous frame and $p(\mathbf{y}_f|\mathbf{x}_f)$ is the *likelihood* that measures how well a hypothesis at frame f explains the observations. **The $p(\mathbf{x}_f|\mathbf{x}_{f-1})$ is often referred to as the *temporal prior*, or *motion model*, and is the main focus of this paper**.

The temporal prior is usually modeled as a first or second order linear dynamical system with Gaussian noise [2], [13], [44]. For example, in [2], [13] the non-informative smooth prior,

$$p(\mathbf{x}_f|\mathbf{x}_{f-1}) = \mathcal{N}(\mathbf{x}_{f-1}, \Sigma), \tag{1}$$

which facilitates continuity in the recovered motions, was used; alternatively, constant velocity temporal priors of the form:

$$p(\mathbf{x}_f|\mathbf{x}_{f-1}, \gamma_{f-1}) = \mathcal{N}(\mathbf{x}_{f-1} + \gamma_{f-1}, \Sigma), \tag{2}$$

where γ_{f-1} is scaled velocity learned or inferred (*e.g.*, $\gamma_{f-1} = \mathbf{x}_{f-1} - \mathbf{x}_{f-2}$), have also been proposed [44] and shown to have favorable properties when it comes to monocular imagery. However, human motion, in general, is non-linear and non-stationary.

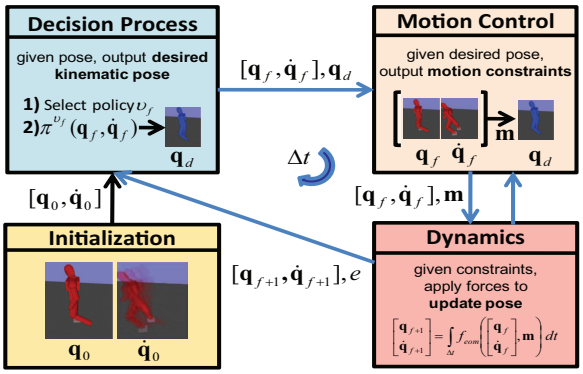

Fig. 3. **Motion Model: Control Loop.** Each iteration advances the figure state $[\mathbf{q}, \dot{\mathbf{q}}, \upsilon]$ by time Δt by (1) generating a desired kinematic pose $\mathbf{q}_d$ for the figure to follow (decision process), (2) constructing corresponding motion constraints $\mathbf{m}$ for generating motor forces (motion control) and (3) applying forces on the figure, as determined by $\mathbf{m}$ (dynamics). Event feedback information e records collisions with the environment and affects selection of the control policy for the next iteration.

For this reason, more recently, it became customary to either model human motion as a mixture of linear models in the original high-dimensional state space [32] or by learning an explicit non-linear low-dimensional embedding of kinematics [23], [26], [46], [50], [51], [52] (or mixture of low-dimensional linear embeddings [25]). The latter class of models often takes a form of Gaussian Processes Latent Variable Models (GPLVMs) [50], [51], [52]. Consequently, to learn effective low-dimensional latent representations, GPLVM-based models are often restricted to particular classes of motion (*e.g.*, walking [51], golf swing [52]). While these models have shown to be effective kinematic priors, and are able to be trained from small datasets [52], they are inherently unable to explicitly model the physical aspects of human motion (*e.g.*, consistency with gravity, balance, *etc*.) Furthermore, it is hard for such models to generalize to different environments. For example, if the kinematic prior model is trained on the data of a person walking on level ground, it may generalize to other people walking on level ground with different styles or speeds [51], but it would not be able to generalize to a person walking up the stairs, as discussed in the results.

3 Tracking with Dynamical Simulation

Dynamical Newtonian simulation is explicitly suited as a *temporal prior* (motion model) to address physical realism in predictions. Temporal prior $p(\mathbf{x}_f|\mathbf{x}_{f-1})$ encodes temporal relationship between states and implicitly approximates underlying processes that govern the motion represented by these states. We assume that true human motion is determined by dynamics and a feedback-based thought process that tasks the actuation of the body (through muscles) such that desired motion would be performed. Our physics-based motion model idealizes this concept by assuming that the thought process can be abstracted by a discrete feedback control loop illustrated in Figure 3. We use this loop to draw samples from the prior.

We run our simulations in a world model consisting of a known static environment, G, and a loop-free articulated structure (*figure*) that represents the subject (Figure 2). We assume "physical properties" (mass, inertial properties and collision geometries) are known for each rigid body segment. Our predictions are produced by a filtering process that takes next pose proposals from a kinematic prior as an input. Proposed predictions are corrected through *simulation* of the articulated body towards the proposed poses. The corrected poses are, by definition, physically valid and transitions between them physically feasible. We abstract kinematic priors by control policy functions. Control policy functions map current poses to next intended poses and implicitly encode intentions (or objectives) of the subject. We permit a collection of (possibly motion-specific) control policies over a variety of objectives and allow inference over which policy to use at any given time. For example, the system can incorporate two different policies for actuated motions (actions), one for walking and another one for jogging, or, it can provide one policy to account for voluntary motions and another for involuntary (passive) body responses. We switch control policies probabilistically, (optionally) conditioned on simulation event feedback, e.

Pose inference in our framework takes the form of a particle filter (see Algorithm 1) with the motion (lines $3-9$, Section 3.2), likelihood (line 12, Section 3.3) and noise (lines 5 and 10, Section 3.4) models explained next.

Algorithm 1 Update particle set for the next frame

Input: Weighted particle set $\{\mathbf{x}_f^{(i)}, w_f^{(i)}\}_{i=1}^N$ for frame f and geometry of the scene G (note $\mathbf{x}_f^{(i)} = [\mathbf{q}_f^{(i)}, \dot{\mathbf{q}}_f^{(i)}, \upsilon_f^{(i)}]$)

Output: Weighted particle set $\{\mathbf{x}_{f+1}^{(i)}, w_{f+1}^{(i)}\}_{i=1}^N$ for $f+1$

1: $\{\tilde{\mathbf{x}}_f^{(i)}, \frac{1}{N}\}_{i=1}^N :=$ resample$(\{\mathbf{x}_f^{(i)}, w_f^{(i)}\}_{i=1}^N)$
2: **for** $i := 1$ to N **do**
3: **if** $\tilde{\upsilon}_f^{(i)} = A$ **then**
4: $\mathbf{q}_d := \pi^A([\tilde{\mathbf{q}}_f^{(i)}, \tilde{\dot{\mathbf{q}}}_f^{(i)}])$ // predict desired pose
5: $\mathbf{q}_d \mathrel{+}= \eta_d$ // add noise: $\eta_d \sim \mathcal{N}(0, \Sigma_d)$
6: **else**
7: $\mathbf{q}_d := \emptyset$
8: **end if**
9: $[\mathbf{q}_{f+1}^{(i)}, \dot{\mathbf{q}}_{f+1}^{(i)}, e] :=$ simulate$([\tilde{\mathbf{q}}_f^{(i)}, \tilde{\dot{\mathbf{q}}}_f^{(i)}], \mathbf{q}_d, G)$
10: $[\mathbf{q}_{f+1}^{(i)}, \dot{\mathbf{q}}_{f+1}^{(i)}] \mathrel{+}= \eta_s$ // add noise: $\eta_s \sim \mathcal{N}(0, \Sigma_s)$
11: $\upsilon_{f+1}^{(i)} \sim p(\upsilon_{f+1}^{(i)}|\tilde{\upsilon}_f^{(i)}, e)$ // sample next policy
12: $w_{f+1}^{(i)} := p(\mathbf{y}_{f+1}|[\mathbf{q}_{f+1}^{(i)}, \dot{\mathbf{q}}_{f+1}^{(i)}])$ // image likelihood
13: **end for**

3.1 Body Model and State Space

Figure state captures information about the pose and control policy and is represented by a vector $\mathbf{x} = [\mathbf{q}, \dot{\mathbf{q}}, \upsilon]$, where $\mathbf{q} \in \mathbb{R}^{31}$ is kinematic pose of the body, $\dot{\mathbf{q}} \in \mathbb{R}^{31}$ is the time derivative of the kinematic pose (velocity), and υ is a discrete identifier designating the control policy currently in use.

Our *figure* (body) consists of 13 rigid body segments and has a total of 31 degrees of freedom (DOFs), as illustrated in Figure 2. Segments are linked to parent segments by

either 1-DOF (hinge), 2-DOF (saddle) or 3-DOF (ball and socket) rotational joints to ensure that only relevant rotations about specific joint axes are possible. The root segment is "connected" to the world space origin by a 6-DOF global "joint" whose DOF values define the global orientation and position of the figure (body). The values of rotational joint DOFs are encoded using Euler angles. Collision geometries attached to segments affect physical aspects of the motion, while additional segment shapes define visual appearance of the segments and have an impact on the evaluation of the likelihood discussed later.

Joint DOF values concatenated along the kinematic tree define the *kinematic pose*, $\mathbf{q}$, of the body. Joint DOF velocities, $\dot{\mathbf{q}}$, defined as the time derivatives, together with the kinematic pose $\mathbf{q}$ determine the *dynamic pose* $[\mathbf{q}, \dot{\mathbf{q}}]$. The pose is considered *invalid* if it causes self-penetration of body parts and/or penetration with the environment (detected by the simulator's collision detection library), or if the joint DOF values are outside of the valid ranges that are learned from the training motion capture data. These constraints on the kinematic pose allow us to reject invalid samples early in the filtering process.

Control policy υ identifies the policy function to be used for generating next pose proposals. The policy function $\pi^{\upsilon} : [\mathbf{q}, \dot{\mathbf{q}}] \mapsto \mathbf{q}_d$ determines the next desired (intended) kinematic pose $\mathbf{q}_d$ that the subject attempts to reach from $[\mathbf{q}, \dot{\mathbf{q}}]$ during simulation and is typically obtained by sampling from an associated kinematic motion prior. We implement two control policies, an *active* motion policy (A) for actuated motions, where $\mathbf{q}_d$ is obtained from kernel regression on training motion capture data, and a *passive* motion policy (P) for unactuated motions, where no particular desired pose is proposed and consequently no motor forces are applied during simulation. Consequently, $\upsilon \in \{A, P\}$ is binary (see Section 3.2.1 and Figure 4 for illustration).

3.2 Motion Model and Control Loop

Sampling from our physics-based motion prior is realized by executing the control loop. For every state hypothesis $\mathbf{x} = [\mathbf{q}, \dot{\mathbf{q}}, \upsilon]$ at one frame[1], one loop iteration is taken to produce a hypothesis at the next frame, as illustrated in Figure 3 and Algorithm 1.

The update procedure uses the current control policy function π^{υ} to propose the next desired kinematic pose $\mathbf{q}_d = \pi^{\upsilon}([\mathbf{q}, \dot{\mathbf{q}}])$ for the figure to approach (*decision process*, see Section 3.2.1). *Dynamics simulation* (see Section 3.2.3) filters this pose to be physically valid by performing *constrained rigid body simulation* of the motion of the subject, guiding its current pose $[\mathbf{q}, \dot{\mathbf{q}}]$ towards $\mathbf{q}_d$, subject to biomechanical and environment constraints, dictated by the scene geometry G. The guidance is realized through application of appropriate motor forces, generated implicitly by the simulator from a set of motion control constraints $\mathbf{m}$ set up by the *motion controller* (see Section 3.2.2) from $[\mathbf{q}, \dot{\mathbf{q}}]$ and $\mathbf{q}_d$. In case no desired kinematic pose was proposed by the policy, $\mathbf{m} = \emptyset$, no motor forces are generated and the subject is let move passively.

1. Where unnecessary, for clarity of notation, we omit sub-scripts for frame identity and super-scripts for hypothesis identity.

	$p(\upsilon_{f+1} = A \mid \upsilon_f, e = 0)$					
	0.95	0.9	0.75	0.5	0.25	0.0
Error (in *mm*)	32.1	30.2	35.0	37.1	38.3	70.8

TABLE 1
Tracking errors as a function of $p(\upsilon_{f+1} = A \mid \upsilon_f, e = 0)$ (L1 walking). For details on the error metric see Section 4.

As a simpler alternative to constraints, the motion controller could generate motor forces directly by *e.g.*, a *proportional-derivative servo* [57].

In order to optionally allow the control process to react to "external events" that took place during simulation, we record event feedback information, e, from the simulator and use it in the decision process to help choose the control policy for the next time step (see Section 3.2.1). We currently restrict ourself to modeling reactions to unanticipated heavy impacts (*e.g.*, as in Figure 13) that are unlikely to be represented well in the training motion capture set. Hence, our feedback information consists of only a binary indicator variable recording whether the body has collided with the environment, detected when a relative velocity at a body contact exceeds a threshold of 1 m/s during simulation.

3.2.1 Decision Process

The decision process in the control loop is responsible for (1) applying the current control policy to propose a next intended kinematic pose $\mathbf{q}_d$ to be corrected by simulation and (2) determining which policy the current policy should switch to after the simulation completes, utilizing the event feedback information from the simulation. We switch policies by a stochastic process in which the new policy υ_{f+1} is sampled from simple $p(\upsilon_{f+1} | \upsilon_f, e)$ distributions that do not take pose or velocity information into account. In practice, we assume that $p(\upsilon_{f+1} = A | \upsilon_f = A, e = 0) = p(\upsilon_{f+1} = A | \upsilon_f = P, e = 0)$ and estimate the value of $p(\upsilon_{f+1} = A | \upsilon_f, e = 0) = 1 - p(\upsilon_{f+1} = P | \upsilon_f, e = 0) = 0.9$ using cross validation. The behavior of the tracker as a function of $p(\upsilon_{f+1} = A | \upsilon_f = A, e = 0)$ is illustrated in Table 1. The value of $p(\upsilon_{f+1} = A | \upsilon_f = A, e = 1) = p(\upsilon_{f+1} = A | \upsilon_f = P, e = 1)$ is set by hand as in our data impacts of desired magnitude happen very infrequently and hence learning (even using cross-validation) is inconclusive. Motivated by [63] we let $p(\upsilon_{f+1} = A | \upsilon_f, e = 1) = 0$.

Passive motion (P). This policy applies no motor forces, as if the figure was unconscious. As a result, no $\mathbf{q}_d$ is generated and no actuation takes place when the policy is in effect. Its purpose is to account for unmodeled dynamics in the active motion policy and it should typically be activated for short periods of time or when the body is in the free fall.

Active motion (A). Our active motion-capture based policy generates desired kinematic poses so that the proposed motion would look similar to training motion capture. We take an exemplar based approach resembling [20], [40], [63] and extend it to work with dynamic poses. To that end, we first form a database of observed input-output pairs (from training motion capture data) between a dynamic pose at frame f and a kinematic pose at frame $f+1$, $\{[\mathbf{q}^*_f, \dot{\mathbf{q}}^*_f], \mathbf{q}^*_{f+1}\}_{f=1}^{n}$. For

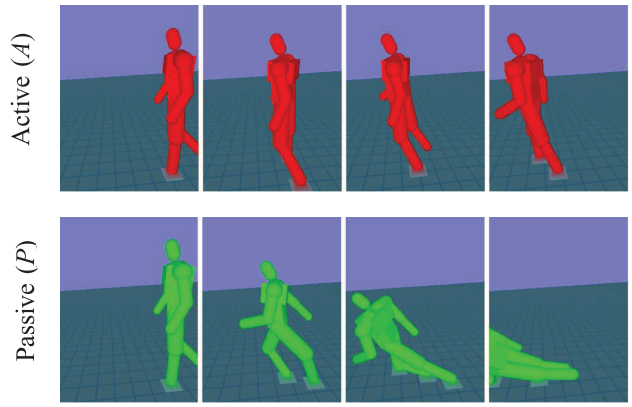

Fig. 4. **Control Policies.** Predictions made by the control loop from a given initial dynamic pose (top and bottom left) for a time duration of 2 seconds. The top row shows poses generated by the active motion policy, the bottom row shows the poses generated by the passive policy.

pose invariance to global position and heading, corresponding degrees of freedom are removed from $\mathbf{q}_f^*$ and $\dot{\mathbf{q}}_f^*$.

Given this database, that can span training data from multiple subjects and activities, and a new query dynamic pose $[\mathbf{q}, \dot{\mathbf{q}}]$, our objective is to determine the next intended kinematic pose $\mathbf{q}_d$. We formulate this objective as in [40] using a k nearest neighbors (k-NN) kernel regression method, where a set of similar prototypes/exemplars to the query point $[\mathbf{q}, \dot{\mathbf{q}}]$ is first found in the database and then the $\mathbf{q}_d$ is obtained by weighted averaging over their corresponding outputs; the weights are set proportional to the similarity of the prototype/exemplar to the query point. This inference can be formally written as,

$$\mathbf{q}_d = \frac{1}{Z_K} \sum_{[\mathbf{q}_f^*, \dot{\mathbf{q}}_f^*] \in neighborhood[\mathbf{q}, \dot{\mathbf{q}}]} K(d_f([\mathbf{q}_f^*, \dot{\mathbf{q}}_f^*], [\mathbf{q}, \dot{\mathbf{q}}])) \cdot \mathbf{q}_{f+1}^*, \quad (3)$$

where Z_K is a normalizing constant, $d_f([\mathbf{q}_f^*, \dot{\mathbf{q}}_f^*], [\mathbf{q}, \dot{\mathbf{q}}])$ is the similarity measure and K is the *kernel* function that determines the weight falloff as a function of distance from the query point.

We use a similarity measure that is a linear combination of positional and velocity information,

$$d_f([\mathbf{q}_f^*, \dot{\mathbf{q}}_f^*], [\mathbf{q}, \dot{\mathbf{q}}]) = w \cdot d_M(\mathbf{q}, \mathbf{q}_f^*) + (1 - w) \cdot d_M(\dot{\mathbf{q}}, \dot{\mathbf{q}}_f^*), \quad (4)$$

where $d_M(\cdot)$ denotes a Mahalanobis distance between $\mathbf{q}$ and $\mathbf{q}_f^*$, and $\dot{\mathbf{q}}$ and $\dot{\mathbf{q}}_f^*$, respectively with covariance matrices learned from the training data, $\{\mathbf{q}_f^*\}_{f=1}^n$ and $\{\dot{\mathbf{q}}_f^*\}_{f=1}^n$; the value of $w = 0.9$ accounts for the relative weighting of the two terms and is determined empirically using cross-validation. For the kernel function, we use a simple Gaussian, $K = \mathcal{N}(0, \sigma)$, with empirically determined variance σ^2.

The method discussed above can be interpreted as a form of a more traditional kinematic prior learned from a database of motion-capture exemplars. While we opted for a simple and robust approach with kernel regression, other regression methods can be used in this context; for example, Gaussian Processes Regression [35] or Conditional Mixture of Experts [5], [45]. The former is closely related to kernel regression, but in addition produces the measure of uncertainty for the prediction; the latter allows for multi-modal predictions. Because we are conditioning on both the kinematics and velocity information, the multi-modality does not seem to be as abundant as with pure kinematic models [5], [45]. Furthermore, as with traditional kinematic motion priors, it is reasonable to assume that the underlying degrees of freedom are much lower than those encoded by the full kinematic state. With that in mind, low-dimensional motion priors (*e.g.*, Latent Variable Gaussian Process Latent Variable Models [50], [51], [52] or Mixture of Factor Analyzers [25]) are likely to facilitate more efficient inference methods. The use of such latent variable models in this context remains future work.

3.2.2 Motion Control

The motion controller conceptually approximates muscle actuation of the subject to move the body from the current pose $[\mathbf{q}, \dot{\mathbf{q}}]$ towards the intended kinematic pose $\mathbf{q}_d$ when the state is updated by dynamics. Because $\mathbf{q}_d$ is generated using a statistical model, kernel regression, it is not guaranteed to be free of self and world penetrations. Motion control, together with physical simulation, is responsible for resolving these penetrations and producing a new pose for the body model that is close to $\mathbf{q}_d$ and can be physically reached from the current pose. We formulate motion control as a set of soft Lagrange multiplier-based constraints [4] on $\mathbf{q}$ and $\dot{\mathbf{q}}$ that implicitly yield actuation forces. Each constraint is defined as an equality or inequality with a softness constant determining what portion of the constraint force should actually be applied to the constrained bodies. Magnitudes of actuation forces can be bounded to account for biomechanical properties of the human motion, like muscle power limits or joint resistance.

Unlike traditional constraint-based controllers [21] that directly constrain and track both linear and angular DOFs of the figure, our objective is to constrain only the angular quantities so that the trajectory traced by the root segment would result from interactions with the environment[2]. However, control that tracks joint angles alone is problematic. One of the problematic cases is illustrated in Figure 5 (right) and (bottom). Consider the case where the desired kinematic pose $\mathbf{q}_d$ is infeasible (*e.g.*, causing penetration with the environment). Leaving the linear DOFs unconstrained, in this case, often leads to unexpected toe contacts/impacts with an environment during simulation which can affect the motion adversely. For example, impacts at the end of the walking cycle (see the impact of the right foot in the middle frame of Figure 5 (bottom)) will force the figure to step back instead of forward.

To address the above mentioned problems, we propose to use a hybrid constraint-based controller (see Figure 6) that can track both desired joint angle trajectories as well as trajectories of selected points on surface geometry of the body that we generally refer to as markers. We use this controller for

2. We constrain the orientation of the root segment in order to implement simple balancing. Although this form of balancing is not physically correct, as the figure's orientation can change regardless of the support from the rest of the body, this strategy allows us to make longer-time predictions required by some of our experiments.

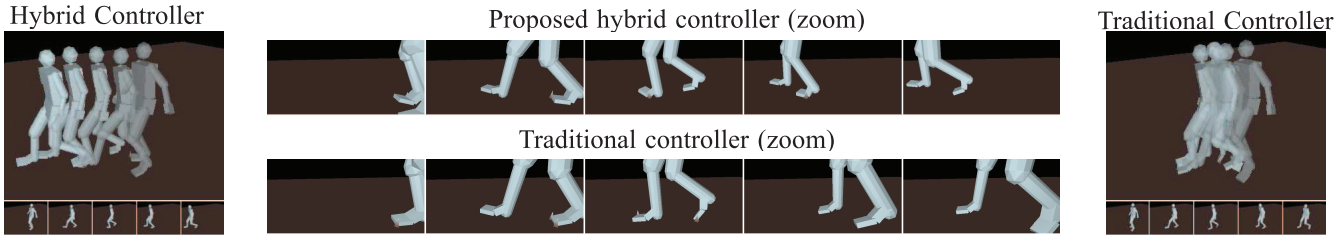

Fig. 5. **Locomotion.** Comparisons of motions generated by a traditional DOF tracking controller (right) and our hybrid controller (left) when the controllers follow the same trajectories specified by the motion capture data. Impacts between the right foot and the ground plane (see the middle frame of the zoomed bottom animation) prevent the traditional controller from performing the motion correctly, resulting in the undesired step backwards, making the figure stay to the right of the camera view. Our hybrid controller on the other hand (see the zoomed top animation) is more robust to such unexpected collisions with the environment, allowing the body to faithfully follow the desired motion.

tracking desired positions of toes (computed with respect to the desired kinematic pose $\mathbf{q}_d$ using forward kinematics) that we adjust in order to avoid penetrations with the environment. Consequently, our markers are attached to the locations of toes and the controller's tracking objectives are

$$\dot{\mathbf{z}}^j = -c_\alpha \cdot (\mathbf{z}^j - \mathbf{z}_d^j) \quad (5)$$

$$\dot{\mathbf{q}}^k = -c_\beta \cdot (\mathbf{q}^k - \mathbf{q}_d^k), \quad (6)$$

where $\mathbf{z}^j$ are the locations of the tracked toe markers j attached to the figure body, $\mathbf{z}_d^j$ are their corresponding corrected desired locations, k are the tracked angular DOFs and $c_\alpha > 0, c_\beta > 0$ are controller parameters[3] determining how fast the controller should approach the desired values.

Ideally, we would like to submit these objectives as constraints for the simulation step. However, in our constraint model, these objectives can not be satisfied directly. The marker tracking objectives prescribe desired values for the marker velocities $\dot{\mathbf{z}}^j$ in the global frame and, consequently, could be satisfied by changing the velocity of the root segment, resulting in undesired motion. To avoid this problem, we add an additional step that uses inverse dynamics to reformulate the objectives. In this step we augment our objective set by

$$\dot{\mathbf{q}}^{root} = (\dot{\mathbf{q}}^{root})_f \quad (7)$$

$$\mathbf{q}^i \geq \mathbf{q}^i_{min} \quad , \quad \mathbf{q}^i \leq \mathbf{q}^i_{max}, \quad (8)$$

where *root* refers to the figure root segment's linear and angular DOFs, $(\dot{\mathbf{q}}^{root})_f$ are the root segment's current DOF values, i iterates over the remaining angular DOFs and $\mathbf{q}^i_{min}$ and $\mathbf{q}^i_{max}$ are the corresponding joint angle limits, learned from the training data. The objective (7) fixes the velocity of the root segment such that the actuation of the body due to the tracking objectives (5), (6) can not be realized by directly actuating the root. We use first-order inverse dynamics, implemented by the physics engine, to solve for the angular velocities $\dot{\mathbf{q}}_d$ of the figure consistent with the augmented objectives and follow these velocities during the actual simulation by requesting

$$\mathbf{m} = \{\dot{\mathbf{q}}^i = \dot{\mathbf{q}}^i_d\}. \quad (9)$$

3. We manually set $c_\alpha = 0.5$ and $c_\beta = 0.2$ so that the controller can replay training motions in simulation.

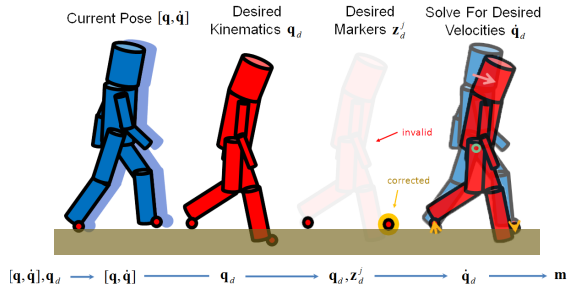

Fig. 6. **Motion Controller.** Input kinematic pose $\mathbf{q}$ determines the positions $\mathbf{z}^j$ of markers on the feet (1), the desired kinematic pose $\mathbf{q}_d$ their desired positions $\mathbf{z}_d^j$ (2). Desired positions are adjusted to prevent penetration with the ground (3) and constraints on the marker world space velocities $\dot{\mathbf{z}}^j$ and relative joint DOF velocities $\dot{\mathbf{q}}^k$ are formed. Finally, constraints are solved for desired velocities $\dot{\mathbf{q}}_d$ using first-order inverse dynamics (4) and the velocities are followed during simulation.

3.2.3 *Dynamical Simulation*

Dynamical simulation numerically integrates the dynamic pose $[\mathbf{q},\dot{\mathbf{q}}]$ forward in time for the time duration of one frame, Δt seconds, following Newtonian equations of motion and a set of active motion constraints[4]. Active constraints honored by the simulator are the explicit motion control constraints $\mathbf{m}$ provided by motion controller, soft position constraints from Eq. (8) implementing joint angle limits and implicit velocity or acceleration constraints enforcing body non-penetration and modeling friction. Because motion control constraints $\mathbf{m}$ are valid only with respect to a specific dynamic pose, the constraints have to be reformulated each time the state is internally updated by the simulator. As a result, motion controller can be called back throughout the simulation process, which is illustrated by the corresponding arrows in Figure 3. For simulation, we use Crisis physics engine [64]. The simulator's collision detection library is used to check for body penetrations.

4. Derivation of equations of motion for the articulated body and the general discussion of constraint-based control is beyond the scope of this paper and we refer reader to [53] for details.

3.3 Likelihood Model

The likelihood function measures how well a particular hypothesis explains image observations. We first define likelihoods for kinematic poses and then extend the approach to handle dynamic poses by considering velocity information.

Kinematic Poses. For a *kinematic pose* $\mathbf{q}_f$, we employ a relatively generic likelihood model $p(\mathbf{I}_f|\mathbf{q}_f)$ from [1] that tries to maximize the similarity between the projection of the model and the observed silhouette extracted from the image $\mathbf{I}_f$.

Dynamic Poses. For a *dynamic pose* $[\mathbf{q}_f, \dot{\mathbf{q}}_f]$, we need to consider information extracted from both the current and the next frame so that velocity information could be implicitly measured and compared against $\dot{\mathbf{q}}_f$. Towards this end, we set up the *coupled observation* $\mathbf{y}_f = [\mathbf{I}_f, \mathbf{I}_{f+1}]$ and define the likelihood of the *dynamic pose* with respect to the coupled observation as a weighted product of two kinematic likelihoods

$$p(\mathbf{y}_f|[\mathbf{q}_f, \dot{\mathbf{q}}_f]) \propto p(\mathbf{I}_f|\mathbf{q}_f)p(\mathbf{I}_{f+1}|\hat{\mathbf{q}}_{f+1}), \tag{10}$$

where $\hat{\mathbf{q}}_{f+1} = \mathbf{q}_f + \Delta t \cdot \dot{\mathbf{q}}_f$ is the estimate of the kinematic state/pose at the next frame. Alternatively, one can formulate a likelihood measure that explicitly computes the velocity information [7] (*e.g.*, using optical flow) and compares it to the corresponding velocity components of the state vector.

The remaining portions of our state $\mathbf{x}_f$, such as the control policy, are inherently unobservable and are assumed to have uniform probability with respect to the likelihood function[5], hence we define our likelihood function $p(\mathbf{y}_f|\mathbf{x}_f)$ as $p(\mathbf{y}_f|\mathbf{x}_f) = p(\mathbf{y}_f|[\mathbf{q}_f, \dot{\mathbf{q}}_f])$.

3.4 Noise Model

The motion model outlined in Section 3.2 is inherently deterministic. In practice, however, as with any Bayesian filtering method, one requires noise (or diffusion process) to account for disturbances and subtleties of a particular motion being tracked. In kinematic trackers, such as [2], [3], [13], [23], [26], [44], it is customary to perform deterministic prediction first and then directly add noise to the predicted state[6]. Adding noise to the state, in our case, would result in (to some extent) the loss of physical realism, because the recovered motion trajectory, in general, cannot be simulated by the physical model exactly. Alternatively, to ensure the realism of recovered motion, the noise can be added to the desired kinematic poses, $\mathbf{q}_d$, before the motion control and dynamic simulation takes place (see Algorithm 1, line 5). While, in principle, this is a desired alternative, it assumes that (1) our physical simulation is rich enough to generate the motion we are observing exactly and (2) that the likelihood model is strong enough to ensure that the physical state of the system (including contact state) can be inferred accurately at all times.

5. The resulting dual-counting of observations, only makes the unnormalized likelihood more peaked, and can formally be handled as in [7].

6. Most often, the deterministic motion model is an identity function $\mathbf{q}_f = \mathbf{q}_{f-1}$ and state diffusion is implemented by replacing the predicted $\mathbf{q}_f$ with a sample from $\mathcal{N}(\mathbf{q}_f, \Sigma)$, where the covariance Σ controls the amount of noise added.

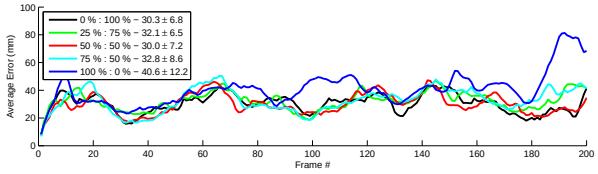

Fig. 7. **Tracking errors for different noise distributions (L1 walking).** Effects of different ratios of noise added to desired poses vs. to predicted poses on tracking errors. All cases produce similar tracking results, except for the case 100 % - 0 %, when all noise is added only to the desired poses and the tracker can not fix global translation errors until the end of the stride.

In practice, neither of the two assumptions holds exactly. In particular, often, because of the noise in the image observations, the moment when the foot hits the ground cannot be observed well using the implemented likelihood. As a result, foot sometimes hits the ground too early (or too late) requiring the model to either take a longer or shorter step to compensate and keep up with subsequent image observations, or to lag behind for the duration of the walk cycle (until support transfers). Both result in sub-optimal performance.

As a practical compromise, we take a hybrid approach, where we add a fraction Σ_d of required noise Σ to the desired pose (see Algorithm 1, line 5) and a fraction Σ_s to the predicted state (see Algorithm 1, line 10)[7]. As is illustrated in Figure 7 (see experiments for description of dataset and error metric) this strategy results in a plausible compromise between the physical realism and the ability to subdue the confusion that arises from lack of good image observations. It is worth noting that even though the resulting motion cannot be guaranteed to be physical simulation from our model, it is *very* strongly biased by the model towards physically plausible solutions, because the amount of noise is relatively small with respect to the deterministic prediction.

4 Experiments

We evaluated our method both in terms of its ability to track common human actions from monocular and multi-view video (see Table 2) as well as to predict human motion alone.

4.1 Data Sets and Performance Metrics

Datasets. In our experiments, we make use of two publicly available datasets [2] and [41], containing synchronized motion capture (MoCap) and video data from multiple cameras. We also collected our own custom monocular sequences that contain no associated MoCap. The use of the synchronized data, in the former datasets, allows us to (1) perform baseline experiments that quantitatively analyze performance, (2) obtain data for training the motion and noise models and to (3) get reasonable initial poses for the first frame of the sequence

7. We set the standard deviation in Σ to be proportional to the maximum expected difference in state between the deterministic prediction and the true observation as in [2], [13] and let $\Sigma = \rho\Sigma_d + (1-\rho)\Sigma_s$ for a factor ρ.

from which tracking is initiated. Each public dataset contains disjoint training and testing data, that we use accordingly.

L1. The first public dataset, used in [2], contains a single subject (L1) performing a walking motion with stopping, imaged with 4 grayscale cameras.

S1 - S3. The second public dataset, HUMANEVA-I [41], contains three subjects (S1 to S3) performing a variety of motions (*e.g.*, walking, jogging, boxing) imaged with 7 cameras (we, however, make use of the data from at most 3 color and 1 grayscale cameras for our experiments).

M. The custom sequences contain monocular footage of a subject (M) exhibiting more complex dynamical interactions with the environment like walking up the stairs and jumping of a ledge. The footage was taken by a low-quality stock digital camera for which no ground truth MoCap information is available. The images were captured at 640×480 resolution at 30 frames per second. Due to the low quality of the camera some of the frames were dropped, resulting in a variable frame rates as low as 15 frames per second. Rough camera calibration was extracted directly from sequences. Geometry of the environment (*e.g.*, ground plane, stairs) was built by hand based on the recovered calibration. For each sequence, initial poses and parameters of the body model (limb lengths) were tuned manually. In all cases, the MoCap for training L1 sequence was used to train motion priors for the M sequences.

Performance metrics. To quantitatively evaluate the performance on standard datasets we make use of the metric employed in [2] and [41], where pose error is computed as an average distance between a set of 15 markers defined at the key joints and end points of the limbs. Hence, in 3D this error has an intuitive interpretation of the average joint distance, in (*mm*), between the ground truth and recovered pose (*absolute error*). In our monocular experiments and specific motion model experiments, we use an adaptation of this error that measures the average joint distance with respect to the position of the pelvis (*relative error*) to avoid biases that may arise due to depth ambiguities and to avoid penalizing of other competing motion priors that do not model changes in 3D position and orientation of the body. For tracking experiments, we report the error of the expected pose.

Method comparison. For comparison with our Physics-based method (**Physics**), we implemented two alternative standard Bayesian filtering approaches, Particle Filter[8] (**PF**) and Annealed Particle Filter[8] with 5 levels of annealing (**APF 5**), each with two priors: a smooth prior (**Smooth**) and a kinematic motion capture exemplar prior (**Mocap**). The kinematic motion capture prior takes the form of

$$\begin{aligned} p(\mathbf{q}_f|\mathbf{q}_{f-1},\dot{\mathbf{q}}_{f-1}) &= \mathcal{N}(\mathbf{q}_d,\Sigma) \\ &= \mathcal{N}(\pi^A([\mathbf{q}_{f-1},\dot{\mathbf{q}}_{f-1}]),\Sigma). \end{aligned} \quad (11)$$

For uniformity with the implementation of our motion controller, the kinematic motion capture prior does not predict the motion of the root; rather it relies on the sampling covariance of the noise model for the positional degrees of freedom of the root segment. To make the comparison as fair as possible we always use the same number of particles, 250 for multi-view sequences, 1000 for monocular sequences[9], same likelihoods, same noise model and same interpenetration and joint limit constraints in all cases; joint limit constraints are learned from training data.

Subject	Action	Source	# Cameras	# Frames	Section
L1	Walking	[2]	4	200	4.3
S3	Jogging	[41]	4	200	4.4
L1	Walking	[2]	1	200	4.5
S3	Jogging	[41]	1	200	4.6
M	Stairs	Custom	1	150	4.7
M	Jumping	Custom	1	35	4.8

TABLE 2
Sequences and data sets used in tracking experiments.

4.2 Motion Prediction with Ground Interaction

We first evaluate the proposed motion model (see Section 3.2) alone. The key aspect of our model is the ability to perform accurate physically-plausible predictions of the future state based on the current state estimates. We demonstrate this ability through quantitative comparisons with predictions made by the standard temporal prior models based on stationary linear dynamics (described in Section 2.1) and exemplar-based predictions.

Figure 9 (right) shows performance of the *smooth* prior (No Prediction, see Eq. (1)), *constant velocity* prior (see Eq. (2)), *kinematic motion capture* prior (see Eq. (11)) and individual predictions based on the two control policies implemented within our physics-based prediction module. For all 5 methods we use 200 frames of motion capture data from the L1 sequence to predict poses from 0.05 to 0.5 seconds ahead. To make sure the experiment results are not biased by the effects of the noise models, we only use deterministic priors with $\Sigma = 0$. We then compare our predictions to the poses observed by motion capture data at corresponding times and report prediction errors.

For short temporal predictions all methods perform well; however, once the predictions are made further into the future, our *active* motion control policy, filtering predictions from the exemplar-based MoCap method, significantly outperforms the competitors. Overall, the active control policy achieves 29 % lower error over the constant velocity prior (averaged over the range of prediction times from 0.05 to 0.5 seconds).

Figure 9 (left) shows the effect of noise on the predictions. For a fixed prediction time of 0.25 seconds, a zero mean Gaussian noise is added to each of the initial ground truth dynamic poses before the prediction is made. The performance is then measured as a function of the noise variance. While performance of the constant velocity prior and passive motion prior degrade with noise, the performance of our active motion prediction stays low and flat.

8. We make use of the public implementation by Balan *et al.*[2] available from http://www.cs.brown.edu/people/alb/.

9. In APF, we use 250 particles for each annealing layer in multi-view sequences and 1000 particles for each layer in monocular sequences.

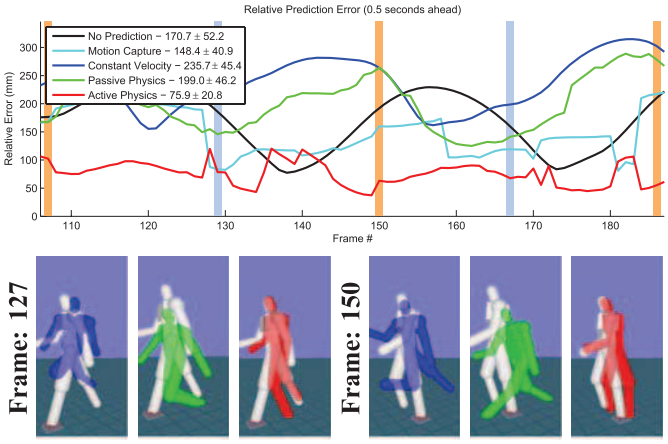

Fig. 8. **Prediction Error.** Errors in predictions (0.5 seconds ahead) are analyzed as a function of one walking cycle. Vertical bars illustrate different phases of walking motion: light blue – foot pushes on the ground, light orange – change in the direction of the arm swing.

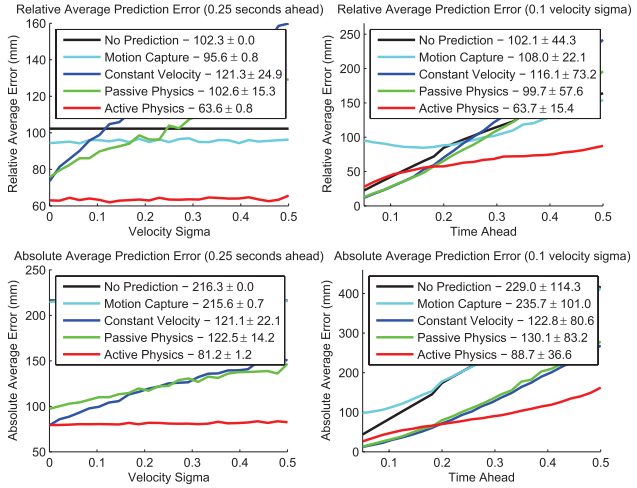

Fig. 9. **Average Prediction Error.** Illustrated, on the right, is the quantitative evaluation of 5 different dynamical priors for human motion: smooth prior (No Prediction), motion capture based prior without physics (Motion Capture), constant velocity prior and (separately) active and passive physics-based priors implemented here. On the left, performance in the presence of noise added to initial ground truth poses is explored. For completeness, top row shows the relative error and bottom absolute error measure. See text for further details.

Notice that in both plots in Figure 9 (right) the constant velocity prior performs similarly to the passive motion; intuitively, this performance makes sense because the constant velocity prior is an approximation to the passive motion dynamics that does not account for environment interactions. Because such interactions happen infrequently and we are averaging over 200 frames, the differences between the two methods are not readily observed, but are important at the key instants when they occur (see Figure 8). For example, predictions of passive motion model tend to be more accurate when foot strikes the ground. Consequently, when change in the direction in the arm swing occurs, inertia tends to allow the motion to continue in the initial direction, making passive motion model predictions less accurate.

	Test: L1 Train: L1	Test: L1 Train: S1-S3
Our method	30.3 (*mm*)	47.2 (*mm*)
Our method ([58] likelihood)	33.9 (*mm*)	52.4 (*mm*)
Xu and Li [58]	49.0 (*mm*)	55.3 (*mm*)
PF (smooth prior, [58] likelihood)	120.7 (*mm*)	
PF (smooth prior)	80.0 (*mm*)	
APF 5 (smooth prior, [58] likelihood)	63.5 (*mm*)	
APF 5 (smooth prior)	42.8 (*mm*)	

TABLE 3
Comparison with other methods (multi-view L1 walk).

4.3 Tracking with Multiple Views (L1 walking)

As our first tracking experiment, we analyze the tracking of the multi-view sequence of L1 (see supplemental video for qualitative analysis). The quantitative results are illustrated in Figure 11 (top left). Our method has 63 % lower error than PF and 27 % lower error than APF with smooth prior; 59 % lower error than PF and 18 % lower error than APF with kinematic motion capture prior. In all cases, our method also resulted in considerably lower variance.

We have also tested how performance of our method degrades with larger training sets that come from other subjects performing similar (walking) motions (see Physics S1-S3 L1). It can be seen that additional training data does not noticeably degrade the performance of our method (only by 0.6 *mm* on an average), which suggests that our approach is able to scale to larger datasets with multiple subjects. We also test whether or not our approach can generalize, by training on data of subjects from HUMANEVA-I dataset and running on a different subject, L1, from the dataset of [2] (Physics S1-S3). The results are encouraging in that we can still achieve reasonable performance that has lower error than PF with either of the two alternative priors, but performs marginally worse than APF with smooth prior. Comparison with APF using kinematic motion capture prior is not meaningful in this case because it is trained using subject specific motion data of L1. Our experiments tend to indicate that our approach can generalize within observed classes of motions given sufficient amount of training data (for generalizations to execution in different environments see Sections 4.7 and 4.8).

We also compare performance of the Bayesian tracking method with our physics-based prior to that of [58]. In [58] a more informative kinematic prior model was proposed (as compared to the smooth prior), that explicitly learns correlations between parts of the body in coordinated motion (*e.g.*, walking). This prior is then used in the context of a more efficient Rao-Blackwellised Particle Filter (RBPF). In [58], however, a weaker likelihood model was used (which we employed in an earlier variant of this work [54]), so we report performance with both types of likelihoods (see Table 3). It is

	Mean (std.) in (*mm*)
Our method	71.5 (19.7)
APF 5 (250 particles/layer)	132.5 (35.5)
APF 5 (500 particles/layer) [24]	111.82 (47.91)
APF 5 GPLVM (500 particles/layer) [24]	99.05 (21.90)
MHT GCMFA [24]	70.13 (21.34)

TABLE 4
Comparison with other methods (multi-view S3 jog).

also worth mentioning that in [58] 1000 particles were used, instead of 250 here; yet our method still performs favorably. Furthermore, from the table one can see that while the models that utilize smooth priors tend to be very sensitive to the quality of the likelihoods (gaining over 30 % performance increase with the better likelihoods utilized here), our model is much less sensitive to those aspects.

4.4 Tracking with Multiple Views (S3 jogging)

To illustrate that our method is not limited to motions of any particular type (*e.g.*, walking), we conducted a similar experiment to above but on the jogging sequence of subject S3 from HUMANEVA-I dataset. All the parameters of the tracker are set as above except that the prior was trained on jogging sequences of the subject S3 (disjoint of the test set). Performance on sample frames is illustrated in the supplemental video and quantitatively analyzed in Figure 11 (bottom left). The proposed model once again achieves lowest error against all competing methods.

We also compare our performance on this sequence to other methods published in literature (see Table 4). In particular, to the results reported in [24], where a Multiple Hypothesis Tracker with a kinematic Globally Coordinated Mixture of Factor Analyzers (MHT GCMFA) prior was presented and compared to an independent implementation of Annealed Particle Filter with 5 layers of annealing with (1) smooth prior (APF 5) and (2) kinematic Gaussian Processes Latent Variable Model prior (APF 5 GPLVM). In this case, the comparison may not be direct because we are not certain as to the exact form of the likelihood used in [24]; in addition [24] uses 500 particles per layer of APF (as opposed to 250 in our implementation). The difference in the number of particles/samples may explain slightly lower performance of our APF implementation (132.5 versus 111.82 (*mm*)). Quantitatively, performance of our method is on par with that of [24] and is more accurate than that of APF 5 GPLVM, despite the fact that we are using half as many samples/particles and a relatively simple kinematic proposal process for $\mathbf{q}_d$. In addition, we expect our method to produce more physically realistic motions.

4.5 Monocular Tracking (L1 walking)

The most significant benefit of our approach is that it can deal with monocular observations. Physical constraints encoded in our prior help to properly place hypotheses and avoid overfitting of monocular image evidence that lacks 3D information (see Figure 10 (Physics) and supplemental video); the results from PF and APF on the other hand suffer from these problems, resulting in physically implausible 3D hypotheses (see Figure 10 (APF 5) bottom) and lead to more severe problems with local optima (see Figure 10 (APF 5) top). Figure 10 (Physics) bottom, illustrates the physical plausibility of the recovered 3D poses using our approach. Quantitatively, our model has 74 % lower error than PF and 76 % lower error than APF with smooth prior; 76 % lower error than both PF and APF with kinematic motion capture prior, with considerably lower (roughly $\frac{1}{6}$) variance (see Figure 11 at top right).

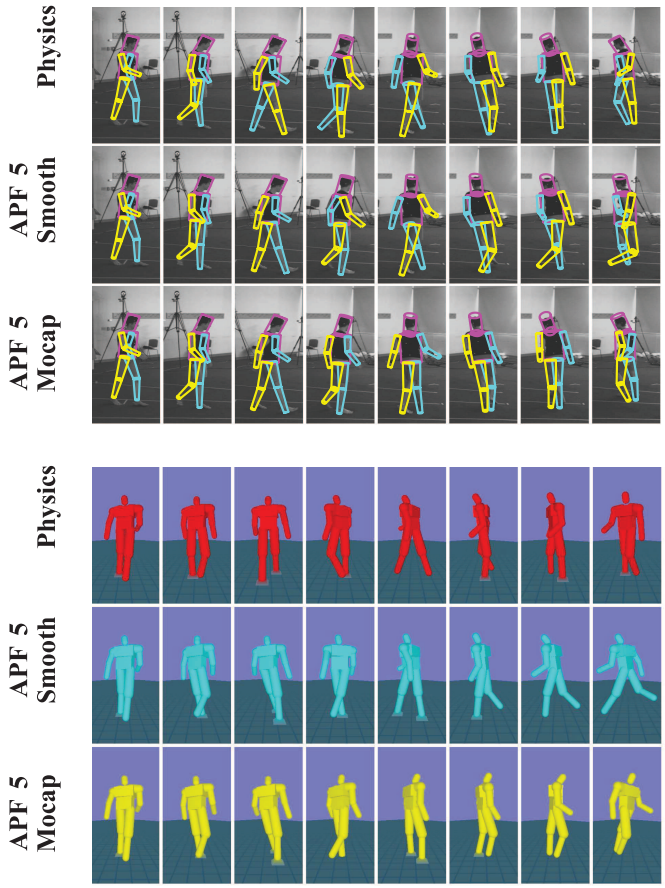

Fig. 10. **Monocular Tracking (walking).** Visualization of performance on a monocular walking sequence of subject L1. Illustrated is the performance of the proposed method (Physics) versus the Annealed Particle Filter (APF 5) with smooth and kinematic motion capture prior; in all cases with 1000 particles. The top row shows projections (into the view used for inference) of the resulting 3D poses at 20-frame increments; bottom shows the corresponding rendering of the model in 3D, from a different view, along with the ground contacts. Our method, unlike APF with either prior, does not suffer from out-of-plane rotations, has consistent ground contact pattern and can estimate correct heading of the subject that is consistent with the direction of motion. APF, on the other hand, produces poses that tend to drift along the ground plane and face in an opposite direction (APF 5 Mocap). For quantitative evaluation see Figure 11 (top right).

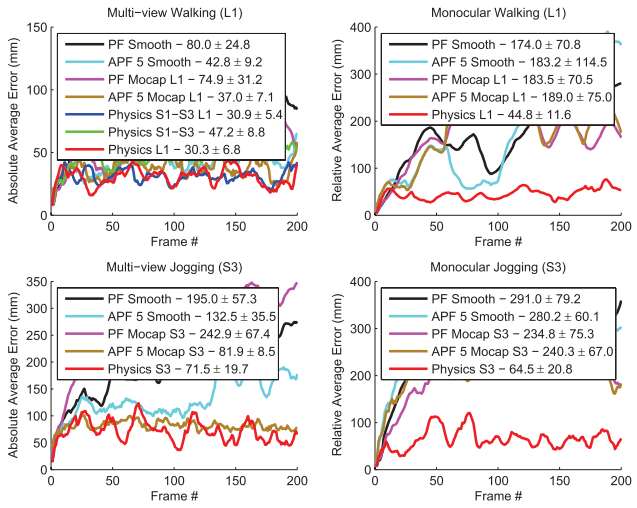

Fig. 11. **Quantitative Tracking Performance.** Performance of the proposed physics-based prior method (Physics) with different testing and training datasets versus standard Particle Filter (PF) and Annealed Particle Filter (APF 5) with 5 layers of annealing. Multi-view tracking performance with 4 cameras and 250 particles is shown on the left, monocular tracking performance with 1000 particles on the right.

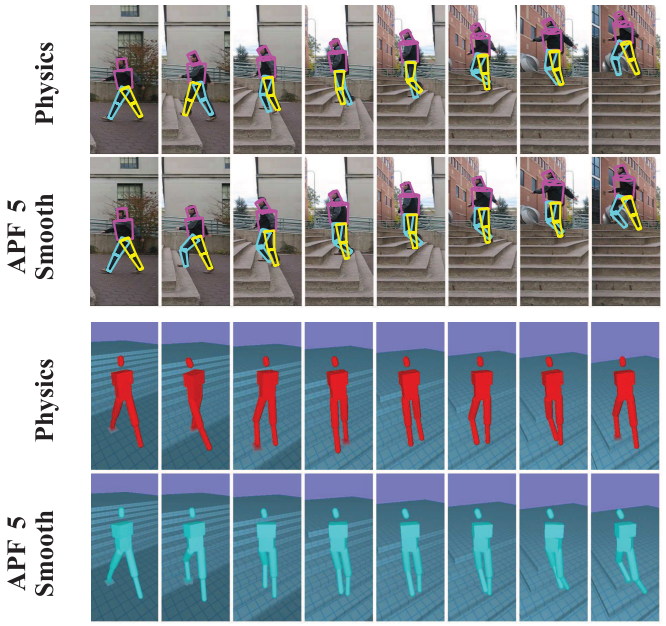

Fig. 12. **Monocular Tracking (stairs).** Ability of the proposed model to generalize to complex interactions with the environment is tested. While the prior is trained only using level-ground walking, the tracker still performs well on this more challenging terrain. Annealed particle filter with the kinematic motion capture prior (APF 5 Mocap) fails to track this sequence completely. See text for additional details and discussion.

4.6 Monocular Tracking (S3 jogging)

Similar results can be seen for the jogging sequence of subject S3 from HUMANEVA-I dataset. The results are quantitatively analyzed in Figure 11 (bottom right). See supplemental video for the qualitative results. While it may seem that tracking with single view performs better than with multiple views, this is not the case, because in the monocular sequence we report marker error with respect to the position of the pelvis (thereby ignoring the global translation of the body).

To be fair, we would like to also acknowledge that the lowest error of 18.9 (*mm*), with standard deviation of 7.5 (*mm*), on this sequence was reported by Urtasun *et al.* in [49]. The method of [49], however, utilizes a very different class of discriminative models. Discriminative models tend to produce quantitatively accurate performance, but result in poses that are extremely noisy over time (no temporal continuity); these models are also difficult to generalize to new motions and/or poses that, in part, result from environment interactions (something we address with our model in the next two experiments).

4.7 Monocular Tracking (M stairs)

A key benefit of the proposed model is its ability to generalize to complex interactions with the environment. In Figure 12 and supplemental video we illustrate performance of our tracker, trained using level-ground walking of L1, on a new subject walking up a set of stairs. Despite the fact that the prior is trained with clearly very different motion from the one observed, our method is able to successfully track the lower body as it interacts with the stairs in order to support the motion of the subject. To our knowledge, no other kinematic prior method is able to illustrate such generalization. Clearly, the knowledge of the environment together with the ability to reason about interactions of the feet with the stairs through physics-based predictions are responsible for the resulting performance.

4.8 Monocular Tracking (M jumping)

In the final experiment (Figure 13), we illustrate performance of the tracker on the fast motion of a subject jumping off a ledge. Because we know that this is mostly ballistic motion, we tuned the parameters of the decision process to always select passive motion control policy (for more efficient inference). The physics-based model is able to track the person reasonably well, even though the footage is of poor quality and the motion is extremely fast, producing large changes in body pose between frames and motion blur (see supplemental video). Of particular interest is the natural way body crouches as it hits the ground (unlike what happens with more traditional kinematic priors, *e.g.*, Figure 13 bottom right). Consequently, the pose changes cannot be predicted by simpler smooth (*e.g.*, a constant-velocity or no-motion) motion prior models, because the noise required to account for the fast motion is simply too large to allow efficient inference. Because this motion was not part of the motion capture training set, comparison with the motion capture based prior is not possible (or meaningful).

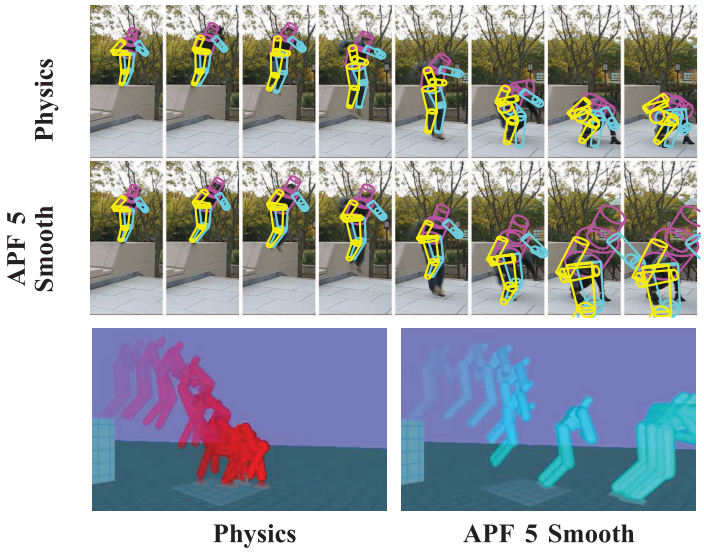

Fig. 13. **Monocular Tracking (jump).** Same as Figure 12, except no motion capture data is used for training and the model relies on passive motion control policy for predictions.

5 Conclusions

We have presented a method for incorporating full-body physics-based constrained simulation, as a temporal prior, into articulated Bayesian tracking. Consequently, we are able to account for non-linear non-stationary dynamics of the human body and interactions with the environment (*e.g.*, ground contact). To allow tractable inference, we also introduce two controllers: a hybrid constraint-based controller, which uses motion-capture data to actuate the body, and a passive motion controller. Using these tools, we illustrate that our approach can better model the dynamical process underlying human motion and achieve physically plausible tracking results using multi-view and monocular imagery. We show that the resulting tracking performance is more accurate than results obtained using standard Bayesian filtering methods such as Particle Filtering (PF) or Annealed Particle Filtering (APF) with kinematic priors. In order to promote use of physical models for tracking, we have made the source code of our controllers and the simulation-based prior available on our website. In the future, we plan to explore richer physical models and control policies, which may further loosen the current reliance of our method on motion-capture training data.

Acknowledgments

This work was supported in part by Office of Naval Research (ONR) Young Investigator Award N000140710141 and ONR Presidential Early Career Awards for Scientists and Engineers (PECASE) Award N000140810910. We wish to thank Michael J. Black for valuable contributions in the early stages of this project; Alexandru Balan for the PF code; David Fleet for insightful discussions; Matt Loper, Deqing Sun and reviewers for feedback on the paper itself; Morgan McGuire and German Gonzalez for discussions; Sarah Jenkins for proofreading.

References

[1] A. Balan, L. Sigal, M. J. Black, J. Davis and H. Haussecker. Detailed Human Shape and Pose from Images, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.
[2] A. Balan, L. Sigal and M. J. Black. A Quantitative Evaluation of Video-based 3D Person Tracking, *IEEE Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, pp. 349–356, October 2005.
[3] J. Bandouch, F. Engstler and M. Beetz. Accurate Human Motion Capture Using an Ergonomics-Based Anthropometric Human Model, *International Conference on Articulated Motion and Deformable Objects (AMDO)*, 2008.
[4] D. Baraff. Linear-time dynamics using Lagrange multipliers, *Computer Graphics Proceedings*, Annual Conference Series: pp. 137–146, 1996.
[5] L. Bo, C. Sminchisescu, A. Kanaujia and D. Metaxas. Fast Algorithms for Large Scale Conditional 3D Prediction, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.
[6] M. Brubaker, L. Sigal and D. J. Fleet. Estimating Contact Dynamics, *IEEE International Conference on Computer Vision (ICCV)*, 2009.
[7] M. Brubaker, D. J. Fleet and A. Hertzmann. Physics-based person tracking using simplified lower-body dynamics, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.
[8] M. Brubaker and D. J. Fleet. The Kneed Walker for Human Pose Tracking. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.
[9] J. Chai and J. Hodgins. Constraint-based Motion Optimization Using A Statistical Dynamic Model, *ACM Transactions on Graphics (SIGGRAPH)*, 26(3), 2007.
[10] N. Chakraborty, S. Berard, S. Akella and J. C. Trinkle. A Fully Implicit Time-Stepping Method for Multibody Systems with Intermittent Contact, Robotics: Science and Systems, 2007.
[11] Y. Choi, B.-J. You and S.-R. Oh On the stability of indirect ZMP controller for biped robot systems, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 2, pp. 1966-1971, 2004.
[12] P. DeVita and T. Hortobagyi. Age causes a redistribution of joint torques and powers during gait, *J Appl Physiol*, Vol. 88, pp. 1804–1811, 2000.
[13] J. Deutscher and I. Reid. Articulated body motion capture by stochastic search. *International Journal of Computer Vision (IJCV)*, Vol. 61, No. 2, pp. 185–205, 2004.
[14] A. Doucet, N. de Freitas and N. Gordon. Sequential Monte Carlo methods in practice, *Statistics for Engineering and Information Sciences*, Springer Verlag, 2001.
[15] P. Faloutsos, M. van de Panne and D. Terzopoulos. Composable controllers for physics-based character animation. *ACM Transactions on Computer Graphics (SIGGRAPH)*, 2001.
[16] A. C. Fang and N. S. Pollard. Efficient Synthesis of Physically Valid Human Motion, *ACM Transactions on Graphics*, 22(3), pp. 417-426, 2003.
[17] D. A. Forsyth, O. Arikan, L. Ikemoto, J. O'Brien and D. Ramanan. Computational Studies of Human Motion: Part 1, Tracking and Motion Synthesis, *ISBN: 1-933019-30-1*, 178pp, July 2006.
[18] K. Grauman, G. Shakhnarovich and T. Darrell. Inferring 3D Structure with a Statistical Image-Based Shape Model, *IEEE International Conference on Computer Vision (ICCV)*, 2003.
[19] J. Hodgins, W. Wooten, D. Brogan and J. O'Brien. Animating human athletics, *ACM SIGGRAPH*, pp. 71–78, 1995.
[20] O. C. Jenkins and M. J. Mataric. Performance-derived behavior vocabularies: Data-driven acqusition of skills from motion, *International Journal of Humanoid Robotics*, 1(2):237–288, 2004.
[21] E. Kokkevis. Practical Physics for Articulated Characters, *Game Developers Conference*, 2004.
[22] A. D. Kuo. A least-squares estimation approach to improving the precision of inverse dynamics computations, *Journal of Biomechanical Engineering*, Vol. 120, No. 1, pp. 148–159, 1998.
[23] C. Lee and A. Elgammal. Modeling view and posture manifolds for tracking, *IEEE International Conference on Computer Vision (ICCV)*, 2007.
[24] R. Li, T.-P. Tian, S. Sclaroff and M.-H. Yang. 3D Human Motion Tracking with a Coordinated Mixture of Factor Analyzers, *International Journal of Computer Vision (IJCV)*, 2010.
[25] R. Li, T. Tian and S. Sclaroff. Simultaneous learning of nonlinear manifold and dynamical models for high-dimensional time series, *IEEE International Conference on Computer Vision (ICCV)*, 2007.
[26] Z. Lu, M. Carreira-Perpinan and C. Sminchisescu. People tracking with the laplacian eigenmaps latent variable model, *Neural Information Processing Systems (NIPS)*, 2007.
[27] J. McCann, N. Pollard and S. Srinivasa. Physics-Based Motion Retiming, *ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, 2006.

[28] P. Michel, J. Chestnutt, S. Kagami, K. Nishiwaki, J. Kuffner and T. Kanade. Online Environment Reconstruction for Biped Navigation, *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 3089–3094, 2006.
[29] T. Moeslund, A. Hilton and V. Kruger. A Survey of Advances in Vision-Based Human Motion Capture and Analysis, *International Journal of Computer Vision and Image Understanding*, 104(2), pp. 90–126, 2006.
[30] D. Metaxas and D. Terzopoulos. Shape and nonrigid motion estimation through physics-based synthesis, *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 15(6), pp. 580–591, June, 1993.
[31] Y. Nakamura and K. Yamane Dynamics Computation of Structure-Varying Kinematic Chains and Its Application to Human Figures, *IEEE Transactions on Robotics and Automation*, Vol. 16, No. 2, pp. 124–134, 2000.
[32] V. Pavlovic, J. Rehg, T. J. Cham and K. Murphy. A dynamic bayesian network approach to figure tracking using learned dynamic models, *IEEE International Conference on Computer Vision (ICCV)*, pp. 94–101, 1999.
[33] Z. Popovic and A. Witkin. Physically Based Motion Transformation, *ACM SIGGRAPH*, 1999.
[34] R. Poppe. Vision-based human motion analysis: An overview, *International Journal of Computer Vision and Image Understanding*, 108(1-2), pp. 4–18, 2007.
[35] C. E. Rasmussen and C. Williams. Gaussian Processes for Machine Learning, *MIT Press*, 2006.
[36] R. Riemer, E. T. Hsiao-Wecksler. Improving joint torque calculations: Optimization-based inverse dynamics to reduce the effect of motion errors, *Journal of Biomechanics*, Vol. 41, Issue 7, pp. 1503–1509, 2008.
[37] B. Rosenhahn, C. Schmaltz, T. Brox and H.-P. Seidel. Staying Well Grounded in Markerless Motion Capture, *Pattern Recognition 2008*, DAGM.
[38] B. Rosenhahn, C. Schmaltz, T. Brox, J. Weickert, D. Cremers and H.-P. Seidel. Markerless Motion Capture of Man-Machine Interaction, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.
[39] A. Safonova, J. Hodgins and N. Pollard. Synthesizing Physically Realistic Human Motion in Low-Dimensional, Behavior-Specific Spaces, *ACM Transactions on Graphics (SIGGRAPH)*, 23(3), 2004.
[40] G. Shakhnarovich, P. Viola and T. Darrell. Fast Pose Estimation with Parameter Sensitive Hashing, *IEEE International Conference on Computer Vision (ICCV)*, Vol. 2, pp. 750–757, 2003.
[41] L. Sigal, M. J. Black HumanEva: Synchronized Video and Motion Capture Dataset for Evaluation of Articulated Human Motion. *Technical Report CS-06-08*, Brown U., 2006.
[42] B. Siciliano, O. Khatib. Springer Handbook of Robotics. *Springer*, Berlin, Heidelberg, 2008.
[43] H. Sidenbladh, M. J. Black and L. Sigal. Implicit probabilistic models of human motion for synthesis and tracking, *European Conference on Computer Vision (ECCV)*, Vol. 1, pp. 784–800, 2002.
[44] H. Sidenbladh and M. J. Black. Learning image statistics for Bayesian tracking. *IEEE International Conference on Computer Vision (ICCV)*, Vol. 2, pp. 709–716, 2001.
[45] C. Sminchisescu, A. Kanaujia, Z. Li and D. Metaxas. Discriminative Density Propagation for 3D Human Motion Estimation, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Vol. 1, pp. 390–397, 2005.
[46] C. Sminchisescu and A. Jepson. Generative modeling for continuous non-linearly embedded visual inference, *International Conference on Machine Learning*, pp. 759–766, 2004.
[47] Robot Modeling and Control (Hardcover) M. W. Spong, S. Hutchinson and M. Vidyasagar, *Wiley*, Hoboken, NJ, 2006.
[48] B. Stephens. Humanoid Push Recovery, *Humanoids*, 2007.
[49] R. Urtasun and T. Darrell Local Probabilistic Regression for Activity-Independent Human Pose Inference *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2008.
[50] R. Urtasun, D. J. Fleet, A. Geiger, J. Popovic, T. Darrell and N. D. Lawrence. Topologically-constrained latent variable models, *International Conference on Machine Learning*, 2008.
[51] R. Urtasun, D. Fleet and P. Fua. Gaussian Process Dynamical Models for 3D people tracking, *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2006.
[52] R. Urtasun, D. J. Fleet, A. Hertzmann, P. Fua. Priors for People Tracking from Small Training Sets, *IEEE International Conference on Computer Vision (ICCV)*, 2005.
[53] M. Vondrak, L. Sigal and O. C. Jenkins. Dynamics and Control of Multibody Systems, *Motion Control*, IN-TECH, Vienna, 2009.
[54] M. Vondrak, L. Sigal and O. C. Jenkins. Physical Simulation for Probabilistic Motion Tracking, *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.
[55] D. A. Winter. Biomechanics and Motor Control of Human Movement, *Wiley*, Hoboken, NJ, 2005.
[56] C. R. Wren and A. Pentland. Dynamic Models of Human Motion, *IEEE International Conference on Automatic Face and Gesture Recognition*, 1998.
[57] P. Wrotek, O. Jenkins and M. McGuire. Dynamo: Dynamic Data-driven Character Control with Adjustable Balance, *ACM SIGGRAPH Video Game Symposium*, 2006.
[58] X. Xu and B. Li. Learning Motion Correlation for Tracking Articulated Human Body with a Rao-Blackwellised Particle Filter, *IEEE International Conference on Computer Vision (ICCV)*, 2007.
[59] K. Yamane and Y. Nakamura. Robot Kinematics and Dynamics for Modeling the Human Body, *Intl. Symp. on Robotics Research*, 2007.
[60] K. Yamane and Y. Nakamura. Automatic Scheduling for Parallel Forward Dynamics Computation of Open Kinematic Chains, *Robotics: Science and Systems*, 2007.
[61] K. Yin, S. Coros, P. Beaudoin and M. van de Panne. Continuation Methods for Adapting Simulated Skills, *ACM SIGGRAPH*, 2008.
[62] K. Yin, K. Loken and M. van de Panne. SIMBICON: Simple Biped Locomotion Control, *ACM SIGGRAPH*, 2007.
[63] V. Zordan, A. Majkowska, B. Chiu, M. Fast. Dynamic Response for Motion Capture Animation, *ACM SIGGRAPH*, 2005.
[64] http://crisis.sourceforge.net/

Marek Vondrak was born in Prague, Czech Republic. He received his Sc.M. degree in computer science from Charles University, Prague and is currently pursuing a Ph.D. degree at Brown University, Providence, RI. Marek's research interests include recovery of articulated human motion from video, physical simulation, motion control of humanoids and character animation. His current major focus has concentrated on introducing techniques from computer graphics, robotics and animation to computer vision in order to build effective models of human motion for tracking.

Leonid Sigal is a Research Scientist at Disney Research Pittsburgh. He was a postdoctoral fellow in the Department of Computer Science at University of Toronto from 2007 to 2009. He received Ph.D. in Computer Science from Brown University in 2008; M.A. from Boston University in 1999 and M.S. from Brown University in 2003; B.Sc. degrees in Computer Science and Mathematics from Boston University in 1999. From 1999 to 2001, he worked as a senior vision engineer at Cognex Corporation. Leonid's research interests include computer vision, visual perception, machine learning, and character animation. He has (co)authored over 30 publications in refereed journals and conferences, including in PAMI, IJCV, CVPR, ICCV, ECCV, NIPS, and ACM SIGGRAPH. His work received the Best Paper Award at the Articulated Motion and Deformable Objects Conference in 2006 (with Prof. Michael J. Black). He is a member of the IEEE.

Odest Chadwicke Jenkins, Ph.D. , is an Associate Professor of Computer Science at Brown University. Prof. Jenkins earned his B.S. in Computer Science and Mathematics at Alma College (1996), M.S. in Computer Science at Georgia Tech (1998), and Ph.D. in Computer Science at the University of Southern California (2003). Prof. Jenkins was awarded a Sloan Research Fellow in 2009 and Popular Science Brilliant 10 Award in 2011. His research primarily addresses problems in robot learning and human-robot interaction, topics in computer vision, machine learning, and computer animation.