Estimation of Human Center of Mass and Inertial Parameters via Approximate Bayesian Inference

Robert C, Grande[1] and Doik Kim[2]

Abstract— **Estimation of human body inertia parameters and center of mass is important for a variety of applications such as understanding the motion and stability of humans, personalizing rehabilitation programs, and humanoid robot teleoperation via motion imitation. While previous methods can estimate the inertial parameters or center of mass, these methods either require an expensive or specialized setup or can only estimate the center of mass, not the inertial parameters. In this paper, we introduce a novel method for estimating inertial parameters and center of mass of a human operator using an inexpensive Wii Balance Board and Kinect camera system. It is shown that approaches using naïve methods such as least squares cannot be used to estimate the inertial parameters, and in order to overcome these deficiencies, we utilize approximate Bayesian inference algorithms to estimate the inertial parameters. The experiments show that the methods proposed in this paper not only outperform existing algorithms in terms of predicting center of mass but also successfully return realistic estimates of the inertial parameters.**

I. INTRODUCTION

Estimating the center of mass (CoM) and inertial parameters of a human is an important problem with many applications in stability and fall prediction [1], [2], personalized rehabilitation programs [3], designing impact protective equipment [4], [5], and analyzing and understanding the dynamics of motion of a human. In this work, the primary motivation for inertial parameters estimation is teleoperation of a humanoid robot via motion imitation of a human. Control of humanoid robots is difficult due to a high number of degrees of freedom as well as inherent instability, so it is an attractive idea to pursue motion imitation as a means for controlling the humanoid robot rather than generating control signals from first principles.

Previously, [6], [7] successfully demonstrated motion imitation for upper limbs on a humanoid robot and robotic arm, and [8], [9] successfully imitated simple human walking by parameterizing step size according to height and stride lengths. However, these works only consider limited motion types in which the robot is able to maintain balance in an unconstrained manner. For general motion imitation, the robot is not free to maintain balance in this way as it must imitate both upper and lower limb motions simultaneously. In particular, due to differences in inertial parameters and scale between the human operator and humanoid robot, a naïve copying of joint angles would result in instability, causing the robot to fall down. For instance, while a human with a light torso may bend over while maintaining stability, a robot with a heavy torso would not be able to imitate this motion directly without falling. Therefore, in order to successfully imitate human pose, knowledge of the relative inertial parameters is required. Throughout this work, inertial parameters will refer specifically to the mass of each limb as well as the location of the CoM of each limb.

Previous approaches to estimating inertial parameters requires expensive equipment such as MRIs [10] or force plates and motion capture systems [3]. Furthermore, approaches such as [3], [11] estimate parameters solely from observations and do not leverage past distributional knowledge from the medical literature.

This paper introduces a novel method for estimating the inertial parameters and center of mass of a human using static pose and center of mass data and approximate Bayesian inference algorithms, the Metropolis-Hasting (MH) and Particle Filter (PF) algorithms. Unlike previous approaches to estimate inertial parameters using expensive or specialized equipment, this approach only requires the use of an inexpensive Wii Balance Board and Kinect Sensor.

Using such inexpensive equipment requires that the dataset contain only slow moving or static poses in order for the measurements to be reliable [11]. However, in Section III, it is shown that solely using static pose and CoM data, such as in [11], it is impossible to estimate the inertial parameters. In order to overcome this deficiency, we pose the estimation problem in the Bayesian framework, by combining observational data from static poses as well as leveraging prior distributional knowledge from the medical literature. We utilize two approximate Bayesian inference algorithms for parameter estimation, and in Section V-B, show that our approach both outperforms other algorithms in terms of tracking the CoM, and also returns inertial parameter estimates similar to those in the literature.

II. PREVIOUS WORK

Past work for inertial parameter identification can be broadly classified into two fields. In the first, estimation is performed directly by weighing or observing physical properties of the limbs, for example, by examining cadaver limbs [5], using stereophotometric and anthropometric techniques [12], [4], or using an MRI to estimate density and volume [10], [13]. While these experiments directly measure the inertial parameters properties, they require specialized equipment, substantial time to perform experiments, or require postmortem measurements. For these reasons, such methods

[1] Massachusetts Institute of Technology, MISTI-Korea Intern, 77 Massachusetts Ave, Cambridge, MA, 02139, USA [2] Senior Researcher at Korean Institute of Science and Technology, Hwarangno 14-gil 5 Seongbuk-gu, Seoul, 136-791, Republic of Korea {robgrande415, diki2005}@gmail.com

are not accessible by a more general set of researchers to obtain inertial parameters for new subjects. One may use regression models [5] using height and weight to predict inertial parameters, but these sort of regression models have a wide variance due to variability between body types and are unsuitable for applications such as robot teleoperation, which requires a more accurate estimation of these properties.

In the second group of methods, the inertial parameters are estimated indirectly by observing a human perform some set of dynamic or static motions while in contact with a force plate that measures the center of pressure (CoP). For example, [3] uses a motion capture system and force plates to estimate the dynamic parameters using inverse dynamics and least squares. However, this method requires an expensive experimental setup and this method does not guarantee feasible assignments to variables, i.e. masses may be negative. The problem of negative parameters is addressed in [14] by approximating the skeleton as a large set of point masses, however, this requires optimization routine over substantially more variables with little physical intuition.

[11] uses a simple Kinect and Wii Balance Board setup to predict the center of mass of a human, however, as shown in Section III, it is fundamentally impossible to perform inertial parameter estimation using only static poses. Additionally, the experiments in this paper (Section V-B) show that our methods are less sensitive to measurement noise and outperform [11] in terms of predicting the center of mass. Other methods include [15], which estimates the inertial parameters using visual data and an optimization routine over a hueristic cost function. However, it is not proven that optimization over this cost function yields the correct answer, and is in fact shown to fail for more complicated systems.

In order to overcome these limitations, we formulate the problem in the Bayesian framework and estimate the parameters using approximate Bayesian inference algorithms, MH and the PF. Unlike previous approaches in the second group of literature that estimate the CoM using only experimental data, we leverage the past wealth of data on inertial parameters [12], [4], [16], [17] and use this in designing the prior distribution for Bayesian estimation.

Thus, in contrast to previous methods, the method proposed in this paper does not require expensive or specialized equipment and is guaranteed to return feasible estimates of the mass and center of mass. Our method utilizes past distributional knowledge from the literature and is flexible in that the experimenter may design a prior distribution using any study or distributional form, i.e. it is not limited to simple distributions such as the Gaussian or Uniform distributions.

III. HUMAN MODEL

The human operator is modeled as a set of links corresponding to the following body parts: upper arm, lower arm and hand combination, upper torso, lower torso, upper leg, lower leg and foot combination, and neck. While the model may be made more complex, such as including separate links for hands and feet, these joints correspond closely to those controllable by a humanoid robot, so this parsimonious model suffices. Additionally, body parts with small masses and ranges of motions have negligible effect on the CoM beyond the resolution of the Wii Balance Board.

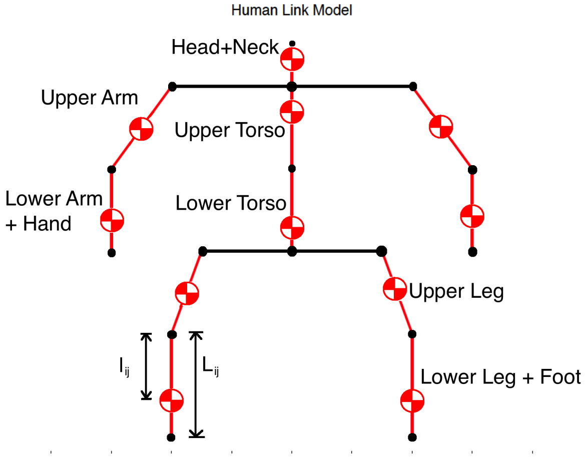

Fig. 1. Chain approximation model of a human.

Joints $\mathscr{J}$ are indexed $i = 1,\ldots,N$ with corresponding position in the horizontal frame $r_i \in \mathbb{R}^2$. Note, while in reality, $r_I \in \mathbb{R}^3$, the measurements of CoM from the Wii are only in $\mathbb{R}^2$, so the vertical component of the position is discarded. Given a limb with proximal joint i and distal joint j, each limb is modeled as a thin rod link labeled as (i,j). In this context, proximal refers to the limb joint closest to the torso, and distal refers to the joint furthest from the torso. This link has mass m_{ij}, length $L_{ij} = \|r_j - r_i\|$, and location of center of mass located along the length of the link, distance $l_{ij} = \rho_{ij} L_{ij}$ from the proximal joint. See Figure 1 for a visual representation. The set of all links used to model the human are given by the set $\mathscr{L}$, and we use the notation $(i,j) \in \mathscr{L}$ to denote that a link exists in our model that connects joint i to joint j.

In this setup, the Kinect sensor measures the joint positions in the horizontal frame, $r_i \in \mathbb{R}^2$, and the wii balance measures the CoP of the human in the horizontal frame. Similar to [11], [18], this work assumes the human operator is static or moving slowly during training. This assumption guarantees that the CoP measured by the Wii balance board is approximately the same as the CoM.

Using this representation, the CoM can be predicted as

$$\text{CoM} = \frac{1}{M} \sum_{(i,j)\in\mathscr{L}} m_{ij}(1-\rho_{ij}) r_i + m_{ij}\rho_{ij} r_j \tag{1}$$

where M is the total mass of the human. Alternatively,

$$\text{CoM} = \frac{1}{M} \sum_{(i,j)\in\mathscr{L}} m_{ij} r_i + m_{ij}\rho_{ij}(r_j - r_i) \tag{2}$$

Consider a naïve method of solving for m_{ij}, ρ_{ij} using least squares. In this case, we rewrite (2) as,

$$\text{CoM} = [r_1, r_2 - r_1, \ldots r_i, r_j - r_i, \ldots]\theta \tag{3}$$

where

$$\theta = [\frac{m_{12}}{M}, \frac{m_{12}}{M}\rho_{12}, \ldots, \frac{m_{ij}}{M}, \frac{m_{ij}}{M}\rho_{ij}, \ldots]^T$$

Then, given K measurements, $k=1,\ldots,K$, using superscript to denote measurement number, we concatenate (3) to receive

$$\begin{bmatrix} \mathrm{CoM}^1 \\ \vdots \\ \mathrm{CoM}^k \\ \vdots \\ \mathrm{CoM}^K \end{bmatrix} = \begin{bmatrix} r_1^1, r_2^1 - r_1^1, \ldots r_i^1, r_j^1 - r_i^1, \ldots \\ \vdots \\ r_1^k, r_2^k - r_1^k, \ldots r_i^k, r_j^k - r_i^k, \ldots \\ \vdots \\ r_1^K, r_2^K - r_1^K, \ldots r_i^K, r_j^K - r_i^K, \ldots \end{bmatrix} \theta \tag{4}$$

For convenience let Q stand for the matrix in (4). In the case that a joint is shared by two or more links, i.e. $\exists i,j,k$ s.t. $\{(i,k),(k,j)\} \in \mathscr{L}$, as is the case in the human model, the matrix Q has redundant columns and therefore is ill-conditioned. Regardless of the number of observations, one cannot solve for θ using least squares. This is due to an inherent observability problem when using only static observations. That is to say, when observing the center of mass during only static positions, it is impossible to delineate both how much a link weighs and where the center of mass lies along the link simultaneously. There exists a null space in which a change in the mass of a link coupled with a movement of the center of mass will result in identical center of mass observations. The formulations in [11], [18] use only static position measurements and therefore suffer from the same problem of observability.

IV. BAYESIAN INFERENCE ALGORITHMS

In order to account for the problem of observability mentioned in the previous section, we adopt a Bayesian probabilistic approach. Bayesian inference is a method for inferring the most probable parameters given the observations as well as prior distributional knowledge of the parameters. In this case, Bayesian inference fills in the missing information in the null space by supplementing the estimation process with prior distributional knowledge from the medical literature. In this sense, Bayesian inference acts as a sort of regularization, which solves the problem of ill-posedness of Q. This section first reviews the fundamentals of Bayesian inference and then introduces the algorithms used for parameter estimation: MH and the PF.

A. Bayesian Inference

Bayesian inference is a method for determining the posterior probability distribution of a set of (unobservable) parameters given a set of observations as well as a prior distribution over the parameters. More formally, consider a set of parameters of interest, θ, a prior probability density function (pdf) over those parameters $p(\theta)$, a set of measurements y, and measurement likelihood pdf $p(y \mid \theta)$, parameterized by θ. Baye's rule states that the posterior distribution of the parameters $p(\theta \mid y)$ given the observations is

$$p(\theta \mid y) = \frac{1}{Z} p(y \mid \theta) p(\theta) \tag{5}$$

where $Z = \int_\Theta p(y \mid \theta) p_\theta(\theta)$ and $\Theta = \{\theta \mid p(y \mid \theta) p_\theta(\theta) > 0\}$. Z is a normalization factor to ensure that $p(\theta \mid y)$ integrates to 1 and is *constant* for a fixed set of data. The value of θ with highest posterior probability, i.e. $\theta_{MAP} = \arg\max_\theta p(\theta \mid y)$, is called the the maximum a posteriori (MAP) estimate. In this work, after the MAP estimate has been approximately solved, the center of mass is calculated by plugging the MAP estimate into θ in (2).

In general, it is impracticable or infeasible to calculate the normalization factor for all but special distributions. Therefore, a variety of approximation techniques must be used. This paper considers two approximation methods, one batch and one online. The first approximation method, Metropolis Hasting (MH), is a member of the Markov Chain Monte Carlo (MCMC) framework [19] and works for batch data. The MCMC algorithm generates samples from the posterior distribution $p(\theta \mid y)$, and one can then record the samples with the highest value of $p(\theta \mid y) \propto p(y \mid \theta) p(\theta)$ to approximately estimate the MAP solution. Since samples with high probability are drawn more frequently than those will low probability, MCMC generates good approximate MAP solutions in a relatively small number of samples.

In general, MCMC works well for solving approximate MAP problems in which the dimensionality of the parameters is high or in which generating samples from the distribution is difficult due to a complicated distribution shape. In this work, the dimensionality of the parameters is quite large at 22, 11 masses and 11 center of masses, and the prior distribution has many constraints.Therefore, MCMC is a natural fit for this estimation problem. However, while a batch solution may work for many cases, one might also wish to have an online method for parameter estimation to permit some sort of active exploration of parameters. For example, [20], [21] provide feedback to the user during the training phase to move joints that have larger uncertainty.

Thus, the second approximation method uses an online Particle Filter (PF) method [22]. Using a PF, the probability distribution function is approximated using a set of weighted basis probability functions, or particles. After each new observation, the weight of each particle is adjusted according to the measurement likelihood function in order to reflect the new information. In this way, the posterior distribution can be approximated using a reduced basis set and simple multiplicative updates at each observation.

Both algorithms are first presented in their general form, and then the specific form of the likelihood and prior pdfs used in the experiments are presented in Section V-A.

B. Metropolis-Hasting

MCMC is a set of algorithms in the statistics literature that are able to generate samples from a distribution for which direct sampling is difficult. In the work, we draw samples from the posterior distribution $p(\theta \mid y)$ and the specific algorithm used is the Metropolis-Hasting (MH) algorithm. The MH algorithm does not require the exact probability function $p(\theta \mid y)$ but simply a proportional function, in this case $p(y \mid \theta) p(\theta)$. The MH algorithm works by generating a random walk using some distribution which may be directly sampled, such as a uniform or Gaussian distribution, and then accepting or rejecting samples according to the relative likelihoods of the current sample and the proposed next sample. For a more detailed explanation of the MH algorithm and its applications, see [19].

The algorithm is initialized with an arbitrary parameter sample, θ_0. In this work, θ_0 is drawn from the prior distribution, $\theta_0 \sim p(\theta)$. Then, at every iteration, the MH algorithm proposes a new sample by perturbing the current sample $\theta' \sim f(\theta_i)$ according to some distribution that is simple to sample. In this work, at every iteration, the algorithm adds white Gaussian noise to the current sample $\theta' \sim \mathcal{N}(\theta t, \sigma_{MH}^2)$. With some probability, the new sample is then accepted or rejected using the relative posterior probabilities of the samples, i.e. with probability $\alpha = \min\left(1, \frac{p(\theta'|y)}{p(\theta_t|y)}\right)$ the new sample is accepted and $\theta_{t+1} = \theta'$, or the sample is rejected so $\theta_{t+1} = \theta_t$. By accepting proposed samples with this modified probability, it can be shown that the resulting distribution, as the number of samples drawn tends to infinity, is the true posterior distribution [19]. This work focuses on estimating the MAP solution, so in addition, the sample with the highest posterior probability is saved.

Lastly, while in theory the MH algorithm will return samples from the target distribution as the number of iterations tends to infinity, in practice the MH may become stuck in local optima. Therefore, random restarts are needed to ensure proper mixing and prevent convergence to local optima. Determining when to perform random restarts is somewhat domain and user dependent, however, in this work, we perform a restart when the sample hasn't changed for over 1000 iterations. The algorithm is formalized in Algorithm 1.

Algorithm 1 Approximate MAP via Metropolis Hasting

```
Input: Prior distribution $p(\theta)$, likelihood function $p(y \mid \theta)$, random walk distribution $\mathcal{N}(0, \sigma_{MH}^2)$
Initialize sample $\theta_0 \sim p(\theta)$, $\hat{\theta}_{MAP} = \theta_0$
while Samples are needed do
    Generate new candidate $\theta' = \theta_t + \nu$, $\nu \sim \mathcal{N}(0, \sigma_{MH}^2)$
    Calculate acceptance probability $\alpha = \min\left(1, \frac{p(y|\theta')p(\theta')}{p(y|\theta_t)p(\theta_t)}\right)$
    Generate $b \sim \text{Uniform}(0,1)$.
    if $b < \alpha$ then
        Accept new sample, $\theta_{t+1} = \theta'$.
    else
        Reject new sample, $\theta_{t+1} = \theta_t$.
    if $p(y \mid \theta_{t+1})p(\theta_{t+1}) > p(y \mid \hat{\theta}_{MAP})p(\hat{\theta}_{MAP})$ then
        $\hat{\theta}_{MAP} = \theta_{t+1}$
    if $\theta_t$ remains unchanged for 1000 iterations then
        Random restart: $\theta_{t+1} \sim p(\theta)$
```

C. Particle Filter

While the MH algorithm works well for finding the approximate MAP solution, it is constrained to batch data. In order to generate a MAP solution or give feedback during training, the particle filter (PF) is used as an online method for finding the approximate MAP solution. For a detailed introduction to the PF, refer to [22].

The PF approximates the posterior distribution using a set of basis functions, such as the Dirac delta function or a Gaussian distribution with narrow bandwidth. In this work, the posterior distribution is approximated using Dirac delta functions as

$$\hat{p}(\theta \mid y) = \sum_i w_i \delta(\theta - \theta_i) \tag{6}$$

where $\delta(\theta - \theta_i)$ is the dirac delta, which is zero everywhere except for $\delta(0) = \infty$, and integrates to one, $\int_\Theta \delta = 1$. In order to ensure that $\hat{p}(\theta \mid y)$ is a valid probability distribution, i.e. integrates to one and is nonnegative, the weights have the properties $\sum_i w_i = 1$ and $\forall i,\ w_i \geq 0$.

The PF is initialized by drawing particles $\theta_i \sim p(\theta)$ from the prior distribution. The weights are initialized to $w_i' = p(\theta_i)$ and then normalized as $w_i = w_i' / \sum_i w_i'$. Given a new observation y^t, the weights of the particles must be updated to reflect the modified posterior distribution. In particular, one may write the updated posterior probability as

$$p(\theta \mid y^1, \ldots, y^t) \propto p(y^1, \ldots, y^t \mid \theta) p(\theta) \tag{7}$$

and it follows that due to the conditional indendence of the observations conditioned on the parameters θ, (7) can be rewritten as,

$$p(\theta \mid y^1, \ldots, y^t) \propto \prod_{i=1}^{t} p(y^i \mid \theta) p(\theta) \tag{8}$$

Or written recursively as,

$$p(\theta \mid y^1, \ldots, y^t) \propto p(\theta \mid y^1, \ldots, y^{t-1}) p(y^t \mid \theta) \tag{9}$$

Therefore, the weights are updated as $w_i^t = w_i^{t-1} p(y^t \mid \theta)$ and then again normalized. Finding the MAP solution at iteration t requires only searching over the weights w_i and returning the corresponding $\hat{\theta}_{MAP} = \theta_j$, where $j = \arg\max_i w_i$.

In practice, after many observations, many of the weights become negligibly close to zero, as the posterior probability $p(\theta_i \mid y^1, \ldots, y^t)$ goes to zero. Therefore, in order to maintain coverage over the regions with high probability, particles are deleted and resampled close to particles with higher weight. Many techniques exist for resampling, however, our approach is detailed in Algorithm 2, line 13. Essentially, every iteration, the algorithm deletes 10% of the particles with the lowest weight and resamples. For resampling, the algorithm choses an existing particle θ_j with probability proportional to the weight of the particle w_j or choses to generate a new particle from the prior $\theta_k \sim p(\theta)$ with probability α_{PF}. If an existing particle is chosen, a new particle $\theta_k \sim \mathcal{N}(\theta_j, \sigma_{PF}^2)$ is created close to the old particle. The weight is initialized as $w_k' = p(\theta)p(y_1, \ldots, y_t \mid \theta)$. This process is formalized in Algorithm 2.

V. EXPERIMENTS AND RESULTS

This section describes the distributions used in the MH and PF algorithms as well as experiments used for validation.

Algorithm 2 Approximate MAP via Particle Filter

```
Input: Prior distribution $p(\theta)$, likelihood function $p(y \mid \theta)$, particle generation distribution $\mathcal{N}(0, \sigma_{PF}^2)$, number of particles $K$
Initialize particles $k = 1, \dots, K$, $\theta_k \sim p(\theta)$.
Calculate probability $w'_k = p(\theta)$
Calculate normalized weights $w_k = \frac{w'_k}{\sum_i w'_k}$
Calculate approximate MAP $j = \arg\max_i w_i$, $\hat{\theta}_{MAP} = \theta_j$

while Observations are available do
    Observe new datapoint $y_t$
    Update probabilities $\forall k, w'_k = w'_k p(y_t \mid \theta_k)$.
    Calculate normalized weights $\forall k$, $w_k = \frac{w'_k}{\sum_i w'_k}$.
    Estimate MAP $j = \arg\max_i w_i$; $\hat{\theta}_{MAP} = \theta_j$
    for Each particle $\theta_k$ with weight in the lowest 10% percentile do
        Delete particle $\theta_k$
        Sample from multinomial distribution Multi$(w_1, \dots, w_K, \alpha_{PF})$
        if Particle $\theta_j$ selected then
            Sample $\theta_k \sim \mathcal{N}(\theta_j, \sigma_{PF}^2)$
        else
            Sample $\theta_k \sim p(\theta)$
        Set $w'_k = p(\theta) p(y_1, \dots, y_t \mid \theta)$
```

A. Distributions and Parameters

In order to use Bayesian algorithms, one must designate a prior which generally captures the structure of the true prior distribution. As more data becomes available, the prior has less effect on the MAP solution, so for data rich applications, the exact formulation of the prior is not as important. Since the weight and dimensions of each human operator will be different, we non-dimensionalize the distribution by considering the distribution over mass fraction (i.e. $\frac{m_{ij}}{\sum m_{ij}}$) and the distance of the center of mass relative to the link length, ρ_{ij}.

In the experiments section, the prior over the masses and center of masses is modeled as a Gaussian with truncated tails and hard constraints that the body masses are symmetric and that the mass sums to one. The mean and standard deviation over the masses m_{ij} and center of masses ρ_{ij} are derived primarily from [17], [4] and are listed in Table I. The tails are truncated such that no body part has negative mass and such that the center of mass is contained within the link modeling the body part. Additionally, we constrain the distribution such that left-right symmetry is maintained and such that the masses sum to one. Mathematically,

$$p(\theta) \propto \begin{cases} 0 & : \sum m_{ij} \neq 1 \\ 0 & : \exists m_{ij} < 0 \\ 0 & : \exists \rho_{ij} < 0 \\ 0 & : \exists \rho_{ij} > 1 \\ 0 & : \text{Not symmetric} \\ \mathcal{N}(\mu_\theta, \Sigma_\theta) & : \text{otherwise} \end{cases} \tag{10}$$

TABLE I

PARAMETERS FOR PRIOR DISTRIBUTION

	m (mean, variance)	ρ (mean, variance)
Head+Neck	$(0.0775, 0.05^2)$	$(0.5344, 0.05^2)$
Upper Torso	$(0.1284, 0.05^2)$	$(0.5529, 0.05^2)$
Lower Torso	$(0.7233, 0.05^2)$	$(0.4903, 0.05^2)$
Upper Arm	$(0.0401, 0.05^2)$	$(0.5503, 0.05^2)$
Lower Arm+Hand	$(0.0291, 0.05^2)$	$(0.7115, 0.05^2)$
Upper Leg	$(0.0987, 0.05^2)$	$(0.4482, 0.05^2)$
Lower Leg+Foot	$(0.0364, 0.05^2)$	$(0.5797, 0.05^2)$

where the mean values and variances are given in Table I. The measurement likelihood function is modeled as a Gaussian with mean given by the right hand side of (2), with variance $\omega^2 = 100^2 \text{mm}^2$,

$$p(y \mid \theta, r) = \mathcal{N}\Big(\sum_{(i,j) \in \mathscr{L}} m_{ij} r_i + m_{ij} \rho_{ij} (r_j - r_i), 100^2 \Big) \tag{11}$$

In the experiments, the following parameters were used for the MH and PF algorithms: $\sigma_{PF} = 0.005$, $\alpha_{PF} = 0.01$, $K = 100$, and $\sigma_{MH} = 0.01$.

B. Experiments

In order to validate the methods for parameter identification and center of mass prediction, five subjects were tested. Each subject performed in two trials. In each trial, the subject moved slowly and demonstrated a variety of different poses for approximately two minutes. The first of these trials was used as a training dataset by the algorithms to estimate the mass distribution, and the second data trial was used as a validation, or test, dataset. In addition to testing the MH and PF algorithms, we also compare our methods to the SECS model of [11], which does not estimate the mass distribution but estimates regression coefficients that can be used to predict the center of mass from angle positions.

We compare the predicted center of mass to that measured by the Wii balance board on the second *testing dataset*, which is different from the first dataset used to train the algorithms. For testing CoM prediction, the MH and PF MAP estimates are plugged into θ of (2). There is no way to directly measure the inertial parameters of the human, so this paper simply compares the output of these algorithms to the prior distribution. Further work will focus on other validation methods.

A Kinect camera estimates the pose of the human using the built-in skeleton detection algorithm, with sampling frequency approximately 30 Hz. The wii balance board records the center of pressure, which is assumed to be close to the center of mass, in the horizontal place at approximately 60 Hz. Both Kinect and balance board measurements are first filtered using a median filter and then a sliding average filter. The median filter considers the nearest 7 measurements and is used to filter outliers that sporadically occur when the Kinect skeleton detection algorithm loses track of the skeleton. The sliding window filter has the purpose of both filtering noise as well as accounting for different sensor sampling rates. In particular, the sliding average filter interpolates

TABLE II

CoM Prediction MAE (mm)

	Subject 1	Subject 2	Subject 3	Subject 4	Subject 5
MH	28.87	27.88	30.59	16.56	22.13
PF	27.68 1	27.43	30.69	18.79	23.44
SESC	60.06	65.92	43.49	42.52	59.9

the observation value by taking a weighted moving average of all observations within a 0.1 s window.

Table II compares the mean absolute error (MAE) predicting the center of mass for each person. Figure 2 shows example CoM trajectories and predictions for the test trials. Lastly, table III shows the inertial parameter estimates compared with the initial prior estimates for one subject.

In general, both the the PF and MH algorithms perform well in terms of predicting the CoM location. Both algorithms outperform the SESC algorithm by approximately a factor a two for all subjects. CoM predictions are generally within 30 mm of the Wii Balance Board prediction, so the prediction capability is quite reliable. On the other hand, the SESC algorithm appears to overfit the parameters and perform poorly, especially in the *y* dimension. All algorithms had more difficulty in predicting the *y* position of the CoM, most likely due to the fact that the Kinect sensor is not as accurate in measuring depth.

The MH and PF algorithms return reasonable estimate for the inertial parameters, although there is some variability between the results returned by the algorithms. It is likely that a more thorough training phase is needed in which the subject performs a greater variety of movements focusing on specific limbs. For example, if a subject lifts his full arm up, the algorithm cannot distinguish the difference between the lower arm and the upper arm, because both are moving simultaneously. On the other hand, if one moves only the lower arm, keeping the upper arm fixed, this would help the algorithm find the true parameters.

VI. DISCUSSION

This paper presented a novel method for estimating the mass distribution of a human using a position and center of mass measurements of a human performing slow moving or static poses. Unlike previous methods, our setup does not require expensive or specialized equipment and only requires a Kinect and Wii Balance Board. Using such inexpensive equipment requires that the dataset contain only slow moving or static poses in order for the measurements to be reliable. However, in Section III, it was shown that using only static position measurements, it is impossible to determine the mass distribution due to an observability problem. Therefore, the problem was formulated in the Bayesian framework using prior distributions derived from the medical literature, and two approximate Bayesian inference algorithms, Metropolis-Hasting (MH) and the Particle Filter (PF), were used to estimate the mass distribution. In Section V-B, it was shown that the MH and PF estimates of mass distribution were close

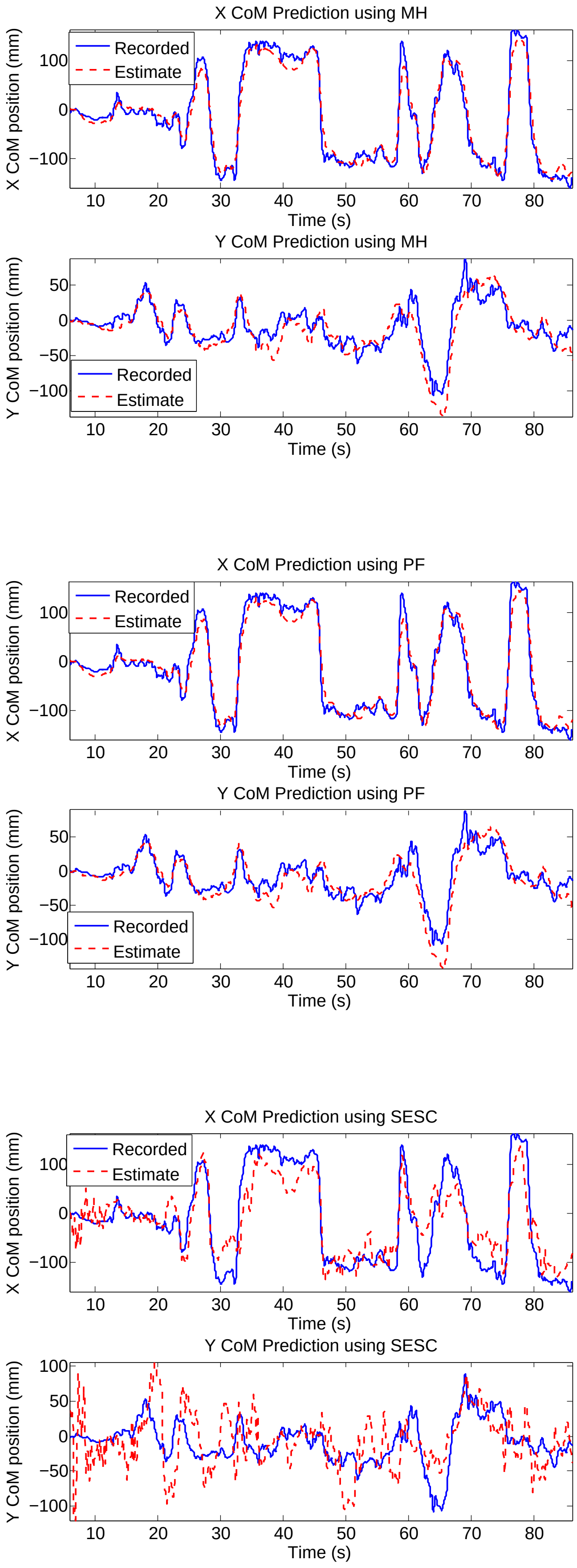

Fig. 2. CoM prediction using (Top) MH, (Middle) PF, (Bottom) SESC

TABLE III

MASS DISTRIBUTION FOR SUBJECT 1-5. PARAMETERS LISTED IN TUPLES (m FRACTION, ρ)

	Head+Neck	Upper Torso	Lower Torso	Upper Arm	Lower Arm+Hand	Upper Leg	Lower Leg+Foot
Prior Mean	(0.0775,0.5433)	(0.1284,0.5529)	(0.2733,0.4903)	(0.0401,0.5503)	(0.0291,0.7115)	(0.0987,0.4482)	(0.0364,0.5797)
Subject 1							
MH Estimate	(0.0494,0.5408)	(0.1193,0.5216)	(0.4394,0.5523)	(0.0149,0.5704)	(0.0107,0.7185)	(0.0891,0.3708)	(0.0812,0.6379)
PF Estimate	(0.0349,0.5473)	(0.1436,0.5288)	(0.4051,0.5231)	(0.0218,0.5505)	(0.0095,0.7140)	(0.1050,0.4064)	(0.0720,0.6088)
Subject 2							
MH Estimate	(0.0140,0.5119)	(0.1142,0.5441)	(0.3929,0.5773)	(0.1077,0.5245)	(0.0099,0.7074)	(0.1120,0.4275)	(0.0099,0.6768)
PF Estimate	(0.0105,0.5382)	(0.1446,0.5503)	(0.3340,0.5151)	(0.1015,0.5759)	(0.0098,0.7066)	(0.1003,0.4364)	(0.0439,0.6333)
Subject 3							
MH Estimate	(0.0134,0.4719)	(0.3422,0.4982)	(0.3965,0.4678)	(0.0104,0.5116)	(0.0104,0.6882)	(0.0240,0.4320)	(0.0791,0.6881)
PF Estimate	(0.0447,0.5175)	(0.2877,0.5700)	(0.3630,0.4720)	(0.0097,0.5552)	(0.0098,0.7222)	(0.0713,0.4346)	(0.0616,0.6032)
Subject 4							
MH Estimate	(0.0134,0.4697)	(0.3422,0.5451)	(0.3965,0.6114)	(0.0102,0.5240)	(0.0102,0.6835)	(0.0460,0.4088)	(0.0843,0.6842)
PF Estimate	(0.1046,0.5346)	(0.1953,0.5393)	(0.3384,0.4792)	(0.0103,0.5522)	(0.0101,0.7028)	(0.0876,0.4313)	(0.0729,0.5962)
Subject 5							
MH Estimate	(0.1943,0.5992)	(0.0255,0.4678)	(0.4212,0.5487)	(0.0106,0.5193)	(0.0106,0.6784)	(0.0360,0.4140)	(0.1223,0.6899)
PF Estimate	(0.1219,0.5746)	(0.1332,0.5386)	(0.3436,0.5075)	(0.0098,0.5506)	(0.0173,0.7232)	(0.0581,0.4261)	(0.1154,0.6448)

to those in the medical literature and that using these mass distribution estimates, one can successfully predict the center of mass of a human from observed joint angles. Additionally, both MH and PF algorithms outperformed the Statically Equivalent Serial Chain (SESC) Model [11] in terms of prediction error of the center of mass for new test data.

Further work will focus on using these human inertial parameter estimates to allow a humanoid robot with different inertial parameters to imitate the motion of a human operator. Additionally, future work will focus on methods for generating feedback for users during the training phase to generate poses which result in the most uncertainty reduction.

ACKNOWLEDGMENT

This work was supported by the KIST Institutional Program(Project No. 2E24800).

REFERENCES

[1] E. E. Stone and M. Skubic, "Unobtrusive, continuous, in-home gait measurement using the microsoft kinect," 2013.

[2] A. Dutta and A. Banerjee, "Low-cost visual postural feedback with wii balance board and microsoft kinect-a feasibility study," in *Point-of-Care Healthcare Technologies (PHT), 2013 IEEE*. IEEE, 2013, pp. 291–294.

[3] G. Venture, K. Ayusawa, and Y. Nakamura, "Motion capture based identification of the human body inertial parameters," in *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE*. IEEE, 2008, pp. 4575–4578.

[4] H. Armstrong, "Anthropometry and mass distribution for human analogues. volume 1. military male aviators," *Aerospace Medical Research Lab Wright-Patterson AFB Ohio USA, Tech. Rep*, 1988.

[5] R. Chandler, C. Clauser, J. McConville, H. Reynolds, and J. Young, "Investigation of inertial properties of the human body," DTIC Document, Tech. Rep., 1975.

[6] N. Miller, O. C. Jenkins, M. Kallmann, and M. J. Mataric, "Motion capture from inertial sensing for untethered humanoid teleoperation," in *Humanoid Robots, 2004 4th IEEE/RAS International Conference on*, vol. 2. IEEE, 2004, pp. 547–565.

[7] H. Ishiguro and T. Minato, "Development of androids for studying on human-robot interaction," in *INTERNATIONAL SYMPOSIUM ON ROBOTICS*, vol. 36. unknown, 2005, p. 5.

[8] S.-K. Kim, S. Hong, D. Kim, Y. Oh, B.-J. You, and S.-R. Oh, "Online footprint imitation of a humanoid robot by walking motion parameterization," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 2692–2697.

[9] S. Hong, Y. Oh, D. Kim, and B. J. You, "A walking pattern generation method with feedback and feedforward control for humanoid robots," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 1078–1083.

[10] C.-K. Cheng, H.-H. Chen, C.-S. Chen, C.-L. Lee, and C.-Y. Chen, "Segment inertial properties of chinese adults determined from magnetic resonance imaging," *Clinical biomechanics*, vol. 15, no. 8, pp. 559–566, 2000.

[11] A. González, M. Hayashibe, and P. Fraisse, "Estimation of the center of mass with kinect and wii balance board," in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 1023–1028.

[12] J. T. McConville, C. E. Clauser, T. D. Churchill, J. Cuzzi, and I. Kaleps, "Anthropometric relationships of body and body segment moments of inertia," DTIC Document, Tech. Rep., 1980.

[13] D. Pearsall, J. Reid, and R. Ross, "Inertial properties of the human trunk of males determined from magnetic resonance imaging," *Annals of biomedical engineering*, vol. 22, no. 6, pp. 692–706, 1994.

[14] K. Ayusawa, G. Venture, and Y. Nakamura, "Real-time implementation of physically consistent identification of human body segments," in *Robotics and Automation (ICRA), 2011 IEEE International Conference on*. IEEE, 2011, pp. 6282–6287.

[15] J. Zhao, Y. Wei, S. Xia, and Z. Wang, "Estimating human body segment parameters using motion capture data," in *Universal Communication Symposium (IUCS), 2010 4th International*. IEEE, 2010, pp. 243–249.

[16] R. K. Jensen, "Body segment mass, radius and radius of gyration proportions of children," *Journal of Biomechanics*, vol. 19, no. 5, pp. 359–368, 1986.

[17] ——, "Changes in segment inertia proportions between 4 and 20 years," *Journal of Biomechanics*, vol. 22, no. 6, pp. 529–536, 1989.

[18] S. Cotton, M. Vanoncini, P. Fraisse, N. Ramdani, E. Demircan, A. Murray, and T. Keller, "Estimation of the centre of mass from motion capture and force plate recordings: a study on the elderly," *Applied Bionics and Biomechanics*, vol. 8, no. 1, pp. 67–84, 2011.

[19] S. Chib and E. Greenberg, "Understanding the metropolis-hastings algorithm," *The American Statistician*, vol. 49, no. 4, pp. 327–335, 1995.

[20] K. Ayusawa, Y. Nakamura, and G. Venture, "Optimal estimation of human body segments dynamics using realtime visual feedback," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 1627–1632.

[21] A. González, M. Hayashibe, and P. Fraisse, "Online identification and visualization of the statically equivalent serial chain via constrained kalman filter," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 5323–5328.

[22] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later."