Proceedings of the 2005 IEEE
Engineering in Medicine and Biology 27th Annual Conference
Shanghai, China, September 1-4, 2005

Towards an Integrated System for Estimating Multi-joint Movement from Diverse Sensor Data

Xiuchuan Pan, Emanuel Todorov and Weiwei Li

Abstract— **Motion capture has been an active research field for many years. Previous methods have mostly focused on converting segment orientations into joint angles. However such methods have a number of drawbacks: the estimates of joint angles are not optimal in a statistical sense; the common assumption that sensor placement is known and constant (relative to the limb) is in practice violated; existing methods are tailored to specific types of motion capture hardware, and cannot be easily adapted to new equipment or to situations where multiple sensor types are used simultaneously. This paper presents an approach that aims to resolve these problems, by combining modern statistical inference with domain-specific optimization. Our algorithm allows arbitrary collections of sensors to be used simultaneously (e.g. position and orientation markers, goniometers, gyroscopes), infers the sensor placements and segment sizes along with the time-varying joint angles (allowing the user to attach sensors quickly and inaccurately), and provides error bars for all quantities that it estimates.**

I. INTRODUCTION

The goal of this project is to develop an easy-to-use data analysis system for fast, accurate, and robust estimation of the multijoint movement trajectories of the human body. Such technology has a number of applications, including clinical gait analysis, physical medicine and rehabilitation, sports medicine and injury prevention, quantitative assessment of motor dysfunction, design and fitting of orthoses and prostheses, feedback control of neuromuscular stimulators, calibration of implantable sensors. Access to reliable multijoint estimation tools is also a prerequisite for continued progress in basic motor control research on both humans and other species.

Motion capture hardware has become widely available, and allows fast and reasonably accurate measurement of the position, orientation, bending, acceleration, etc. of various makers attached to the body. The available data analysis tools, however, lag behind these hardware advances; estimating the configuration of a multiarticulate body to which markers are non-rigidly attached remains a challenging problem. In particular:

1) existing methods assume rigid marker attachment and provide no estimate of the errors resulting from unavoidable soft tissue deformation and miscalibration;
2) placing markers at predetermined locations and measuring limb sizes for each subject requires prolonged setup sessions;
3) the reliance on hardware-specific estimation methods makes it difficult to utilize new sensor modalities or placements, and to use multiple motion capture devices simultaneously;
4) the redundancy in the sensor data due to the body structure is rarely exploited to handle missing data, marker misidentification, and noise in general;
5) most existing methods are tailored to the needs of the computer animation industry and do not even attempt to meet the accuracy requirements for research and clinical tools;
6) investigators who need such tools are usually faced with the daunting task of developing their own.

Our approach aims to resolve these problems. It is based on a general probabilistic formulation, which allows us to apply a combination of modern statistical estimation, numerical optimization, and software engineering techniques. The multiple core methodologies needed to develop such a solution are already available, albeit in different literatures, and the time is ripe to bring them together. Our ultimate objective is to develop software that is available for public use. Here we present our initial results in that direction.

II. THE GENERAL APPROACH

We formulate the problem of multi-joint movement estimation as a general problem of probabilistic inference. At the heart of that formulation is a *generative model*, which specifies how sensor measurements are obtained.

Let the vector $\mathbf{x}$ describe the instantaneous state of the body (including the position and orientation of the root segment, the angles of all joints that are expressed in Euler coordinates, quaternions or other more sensible representations for ball joints, and the time derivatives of all of the above), as well as all parameters that define segment sizes, axes of rotation, marker placements relative to the limbs, gains and offsets of goniometers, and everything else that affects the measurement process. Let $\mathbf{z}$ denote the vector of sensor readings – which is simply the list of all real numbers that the available sensors are generating, regardless of their type or hardware specifications.

In general, a generative model specifies the probability distribution of all measurable quantities conditional on the values of all unmeasurable quantities. In our case, multivariate Gaussian distribution is assumed:

$$p(\mathbf{z}|\mathbf{x}) = N(\mathbf{f}(\mathbf{x}); V)$$

This work is sponsered by NIH grant R01-NS045915.

X. Pan and E. Todorov are with the department of cognitive science, university of california san diego, la jolla, ca 92093, usa xpan etodorov@cogsci.ucsd.edu

W. Li is with the department of mechanical engineering, university of california, san diego, la jolla, ca 92093, usa wwli@mechanics.ucsd.edu

0-7803-8740-6/05/$20.00 ©2005 IEEE.

where $\mathbf{f}(\mathbf{x})$ is the expected value of the sensor measurements given $\mathbf{x}$. The sensor noise covariance matrix V is determined by the noise characteristics of the hardware. The function $\mathbf{f}$ involves the forward kinematics of the body, as well as models of how each marker is attached to the body and what kind of data it generates. We assume the a prior Gaussian probability distribution $\widetilde{p}(\mathbf{x})$, with mean $\widetilde{\mathbf{x}}$ and covariance matrix $\widetilde{P}$. Thus $\widetilde{p}(\mathbf{x}) = N\left(\widetilde{\mathbf{x}}, \widetilde{P}\right)$.

Using Bayes rule, we obtain the posterior probability

$$p(\mathbf{x}|\mathbf{z}) \sim p(\mathbf{z}|\mathbf{x})\,\widetilde{p}(\mathbf{x})$$

and compute the estimate $\overline{\mathbf{x}}$ which maximizes that posterior. This is known as maximum a posteriori estimation (MAP). Maximizing $p(\mathbf{x}|\mathbf{z})$ is equivalent to minimizing $-\log p(\mathbf{x}|\mathbf{z})$, which, using the formula for a multivariate Gaussian, is proportional to

$$E(\mathbf{x}) \equiv (\mathbf{z}-\mathbf{f}(\mathbf{x}))^{\mathsf{T}} V^{-1} (\mathbf{z}-\mathbf{f}(\mathbf{x})) + (\mathbf{x}-\widetilde{\mathbf{x}})^{\mathsf{T}} \widetilde{P}^{-1} (\mathbf{x}-\widetilde{\mathbf{x}})$$

The Extended Kalman Filter (EKF) deals with the nonlinearity by linearizing $\mathbf{f}(\mathbf{x})$ around the prior mean $\widetilde{\mathbf{x}}$, and solving the resulting quadratic minimization problem. In contrast, we perform full-blown optimization of $E(\mathbf{x})$ in each time step, using the Levenberg-Marquardt method, to increase accuracy and to avoid the instabilities associated with EKF. Once the optimal $\overline{\mathbf{x}}$ is found, we proceed as in the EKF method: the function $\mathbf{f}(\mathbf{x})$ is linearized around $\overline{\mathbf{x}}$, and the estimation error covariance P is updated accordingly.

Regarding the kinematics of human movement, many studies have shown that it is smooth, in the sense that high-order derivatives of position are small. Particularly successful are the minimum-jerk and minimum-torque-change models. We capture this using a stochastic model of dynamics, which simply says that acceleration does not change on average. The changes of acceleration are modeled as zero-mean noise, whose variance is inverse proportional to the ”weight” of the smoothness assumption.

One more caveat has to do with the distinction between rapidly varying quantities (such as joint angles), and quantities that are constant (segment sizes) or nearly constant (marker locations relative to the segments). Let us call the former θ and the latter $\mathbf{a}$, so that $\mathbf{x} = [\theta; \mathbf{a}]$. A key feature of our approach is that $\mathbf{a}$ is treated as a random variable, which has a prior distribution (reflecting average segment sizes and intended marker placements) but its posterior mean and standard errors are computed from the sensor data, just like the time-varying θ. We have considered two approaches. One is to use an EM-like algorithm, that computes a sequence of θ's given $\mathbf{a}$, then recomputes $\mathbf{a}$ given the best estimates of θ, and iterates until convergence. It turned out that if the prior mean of $\mathbf{a}$ is far from the truth, this method has poor convergence. Therefore we have settled on a different method, that estimates both θ and $\mathbf{a}$ as if they are time varying parameters, but makes a much stronger smoothness assumption about $\mathbf{a}$.

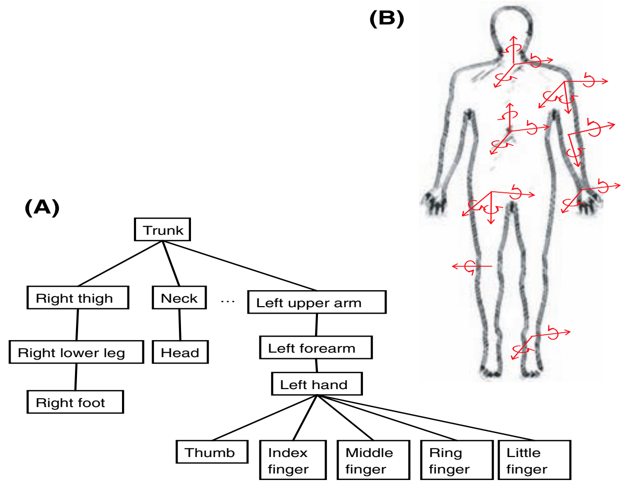

Fig. 1. Illustration of the kinematic tree model.

III. KINEMATIC MODELING

The human body or part of the body is modeled as a kinematic tree whose root is the human trunk or as close to the trunk as possible. Other parts are attached to the root and its descendants to form a tree. The end segments, e.g. fingers, are the leaves of the tree. Fig. 1 shows a simplified human tree structure. Instead of using one single representation for all joints, this paper uses four kinds of joint parameters. Exponential map, or rotation axis with length, is used for ball joints, such as shoulder, neck and hip joints. Euler angles are used for ordered rotation, such as elbow. Two degree of freedom rotation axis with length is used for wrists and ankles. One rotation angle is used for knee and some finger joints. In general, the method is optimized for the purpose of statistical inference rather than visualization and subsequent analysis. If the user wishes to obtain, say, an Euler angle representation for shoulder rotations, that should be computed after a ball-joint configuration has been estimated.

A local right-handed coordinate frame is attached to each segment. The center of the frame is located at the joint center where this segment connects to its parent segment. If this segment has only one child segment, axis X is chosen to pass through the joint center where it connects to its child. Axis Y is set along the rotation axis that is perpendicular to the axis X. If there are two rotation axes perpendicular to axis X, the user has the freedom to choose, but it is better to set axis Y parallel to axis Y of its parent segment when all joint angles are zeros. If the segment has no child segment or more than one child segment, the user is free to choose axis X, but it is still better to let the axis X of both parent and child in the same direction when human body is in rest condition. Note that these are not the requirements for the general method, but simply the convention we currently use while testing the method.

IV. Algorithm

A. State space model

Traditional methods are to estimate/measure segment orientation and joint position in a global coordinate frame, then to convert the segment orientation and joint position into joint angles using inverse kinematics. Instead of doing this in two steps, this paper presents one algorithm that estimates joint angles directly using forward kinematics and optimal estimation.

The smoothness assumption implies that all joint acceleration changes are small and distributed according to a zero-mean Gaussian. The marker positions and orientation are not fixed because of flexible attachment and soft tissue of human body. So the state of the system is:

$$X_{k+1} = A * X_k + \omega \tag{1}$$

$$\omega \sim N(0, Q) \tag{2}$$

in which state variable $X = [\theta_{1..k}\ \dot{\theta}_{1..k}\ \ddot{\theta}_{1..k}\ p_{1..l}\ a_{1..m}]^T$. $\theta_{1..k}$ is the joint angles, $p_{1..l}$ is the position of joint centers in the local frame of their parent segment. Note p_1 is the position of the root in the global frame. $a_{1..m}$ is the sensor attachment parameters such as position and orientation in the local frame of their corresponding segments, except those sensors on leaf segments. The transfer matrix A is a $(3k + l + m) \times (3k + l + m)$ square matrix:

$$A = \begin{bmatrix} I_{k\times k} & \Delta t I_{k\times k} & & \\ & I_{k\times k} & \Delta t I_{k\times k} & \\ & & I_{k\times k} & \\ & & & I_{(l+m)\times(l+m)} \end{bmatrix} \tag{3}$$

Q is the covariance matrix of system noise,

$$Q = diag(0_{2k} \quad \Omega_\theta I_k \quad \Omega_p I_l \quad \Omega_a I_m) \tag{4}$$

in which Ω_θ is the variance of joint angle acceleration noise, Ω_p is the variance of joint center displacement, Ω_a is the variance of marker attachment parameters.

Measurements can be sensor position, orientation, acceleration, angle velocity etc. in the global coordinate frame or in the sensor local frame. It is very hard to write the universal analytical format of the measurement function for all sensors. Instead, measurements are modeled using a chain of forward kinematics.

$$\begin{aligned} z_k &= f(x_k) + v_k \\ v &\sim N(0, V) \end{aligned} \tag{5}$$

in which v_k is the measurement noise at time k, whose covariance matrix is V.

B. Time propagation

Suppose at time k, the state estimate is $\hat{X}_k$, with covariance matrix $P_{k/k}$, the predicted state and measurement at time $k+1$ is:

$$\tilde{X}_{k+1} = A * \hat{X}_k \tag{6}$$

$$P_{k+1/k} = A * P_{k/k} * A' + Q \tag{7}$$

$$\tilde{z}_{k+1} = f(\tilde{X}_{k+1}) \tag{8}$$

A chain of equations are used here to calculate the sensor measurement, in which quaternion is used internally. Suppose position/orientation sensor i is attached on segment k, at the position p_{si} and orientation q_{si} in the local coordinate frame of segment k. Segment k connects to its parent segment $l (l < k)$ at joint j_k, whose position in local frame of segment l is $p_{jk} = (x_k, y_k, z_k)^T$.

The global coordinate of sensor i is

$$p_i = p_k + q_k * p_{si} * \bar{q_k} \tag{9}$$

$$q_i = q_k * q_{si} \tag{10}$$

in which p_k is the position of joint k in global frame, q_k is the orientation of segment k in global frame, and

$$p_k = p_l + q_l * p_{jk} * \bar{q_l} \tag{11}$$

$$q_k = q_l * q_{l,k} \tag{12}$$

in which p_l is the position of segment l in global frame, q_l is orientation of segment l in global frame, $q_{l,k}$ is the rotation from segment l to segment k. $q_{l,k}$ can be calculated from joint angles, depending on their representation.

C. Measurement update

Different from traditional methods, this paper does not use a Kalman filter to do measurement update. Instead, full-blown optimization of the non-quadratic cost function is used to find the optimal estimate of posture and sensor attachment parameters. Given the measurement z_{k+1} at time $k+1$, the optimal estimate of state X is

$$\begin{aligned} \hat{X}_{k+1} = arg \min_{X_{k+1}} ((X_{k+1} - \tilde{X}_{k+1})' P^{-1}_{k+1/k} (X_{k+1} - \tilde{X}_{k+1}) \\ + (z_{k+1} - \tilde{z}_{k+1})' V^{-1} (z_{k+1} - \tilde{z}_{k+1})) \end{aligned} \tag{13}$$

The posterior covariance of state estimate is

$$P^{-1}_{k+1/k+1} = P^{-1}_{k+1/k} + J' V^{-1} J \tag{14}$$

in which J is the Jacobian matrix of measurement function $f(X)$ to state X at $\hat{X}_{k+1}$. P^{-1} is also called information matrix.

V. Simulations and experimental data

The simulation of a 7-degree-of-freedom arm movement is shown in Fig. 2, with position-and-orientation sensors attached to body trunk, the upper arm, forearm, and hand. The simulation result, Fig. 2, compares the actual and estimated joint angles. The initial inaccuracy is due to the fact that the calibration parameters are wrong, and the method needs a certain amount of data to estimate them. Once that happens, joint angle estimation is near-perfect.

The experimental data from 7-degree-of-freedom right arm movement are also shown from Fig. 3 to 6. Fig. 3 shows the sensor attachment position on the arm. It is found that the algorithm corrects the initial inaccurate guess of sensor position, and sensor sliding on the skin because of loose attachment. Fig. 4 shows the estimated joint angles in the movement, Fig. 5 shows the joint angle velocity, and Fig. 6 shows the joint angle acceleration.

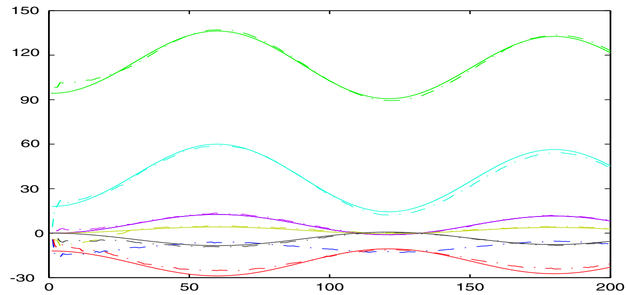

Fig. 2. 7 joint angles (solid curves) and their estimated values(dashed curves) varying over time

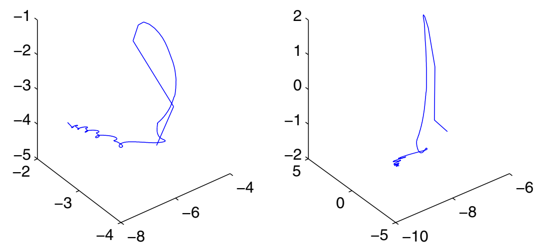

Fig. 3. Sensor slide on arm skin. Left: sensor on upperarm; Right: sensor on forearm

VI. Conclusion

The main idea of this paper is to consider human bodies as flexible bodies, which means that joints can move in the local frame of their parent segment; the soft tissues to which sensors are attached can deform, which brings sensors away from their original position; and sensors can slide on skin. To handle such effects, our algorithm tries to estimate everything that could affect the sensor readings.

This method gives an unbiased estimate of all joint angles, velocities and accelerations, joint positions as well as sensor attachment parameters from noisy measurement. Simulations

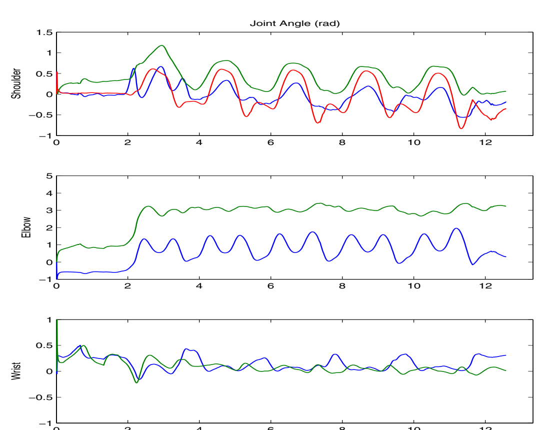

Fig. 4. Joint angles. From top to bottom: Shoulder, elbow and wrist. Blue line is the first rotation angle, green is the second, red is the last.

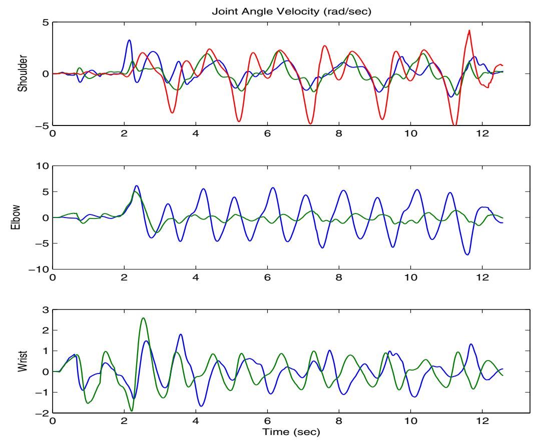

Fig. 5. Joint angles velocity. Same rotation order as in last figure.

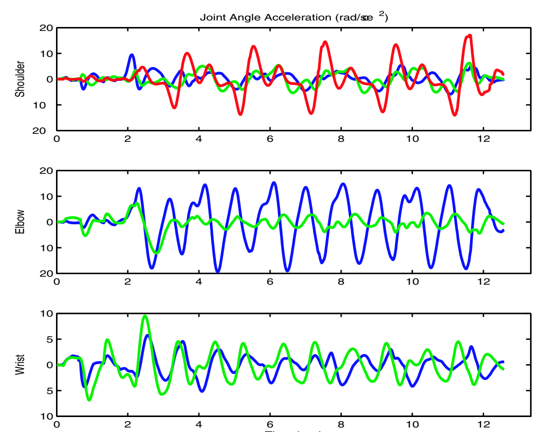

Fig. 6. Joint angles acceleration. Same rotation order as in last two figures.

show that the algorithm works well with sensor noise, sensor sliding on skin, soft tissue deformation and joint position changes. In addition to joint angles, this algorithm also estimates anatomical parameters: the joint positions and segment sizes.

One drawback is that, in the current Matlab implementation, the algorithm does not work in real time. This is often acceptable in offline data analysis, but our aim is to provide an implementation that works in real time. To that end, we have implemented an efficient C++ library for computing multijoint kinematics and their derivatives. Preliminary tests suggest that the optimization will be possible in real time.

References

[1] A. Ude, C. Atkeson, M. Riley, Programming full-body movements for humanoid robots by observation, *Robotics and Autonomous Systems* 47, 2004, pp 93-108.

[2] F. Grassia, Practical Parameterization of Rotations Using the Exponential Map, *Journal of Graphics Tools*, 3(3):29-48 1998.

[3] L. Herda, P. Fua, R. Plankers, R. Boulic and D. Thalmann, Skelton-Based Motion Capture for Robust Reconstruction of Human Motion, *Proceedings of Computer Animation*, 2000.