Interaction Capture And Synthesis

(Notes From The Paper)

by Ashwin Nanjappa

Web: http://www.comp.nus.edu.sg/~ashwinna/

October 31, 2006

1 Introduction

Contact changes the motion of both the character and the environment due to the physical interaction between the two. Animating human hands is challenging due to their highly articulated kinematics and the numerous contacts they exhibit during interaction.

This paper introduces a new technique called *Interaction Capture* which extends motion capture to include capture of contact phenomena. Using both motion and contact forces, an intermediate representation of movement is estimated called the *interaction trajectory*. It describes both the intended motion (a reference joint angle trajectory) and passive response to contact (compliance of the joints). From this representation we can synthesize new interactions with similar objects but of different shapes, sizes and properties using physically based simulation.

2 Related Work

3 Overview Of Interaction Capture

In this section, a simple example is used to describe interaction capture. An index finger moves in the vertical plane and strokes a flat surface. This interaction is captured and resynthesized for different situations.

3.1 Compliant Articulated Structure

The *finger* is modeled as a *kinematic chain* of 3 hinge joints.

The joint angles are collected in the vector θ.

The *compliance* C can be thought of as a collection of torsional springs that when displaced from a *reference configuration* θ_r, produce joint torques τ by the relation:

$$(1){:}\ \theta = \theta_r + C\tau$$

The *contact force* f at a point on the fingertip produces torques τ at each of the joints:

$$(2){:}\ \tau = J^T f$$

where J is the Jacobian matrix which depends on the contact point and finger configuration.

3.2 The Capture Session And Data

2 types of sensors are used to capture the interaction:

1. Motion is captured using *markers* attached to the finger.

2. Contact force is measured using a *6-axis force torque sensor* which is attached below the surface.

Both the measurements are kept in sync using the timing pulse of the motion capture system. Both the values are captured at a relatively high rate of 500 Hz.

Processing the motion data, the joint angle trajectory can be obtained (θ). Using this J can be calculated.

Using J and f (obtained from sensor data) in (2), joint torques τ can be calculated.

3.3 Joint Compliance Estimation

Previous methods used extra equipment to measure compliance.

This paper doesn't use the above. Instead, we observe that:

1. Compliance follows after contact.

2. Compliance changes are slow.

Therefore, the key insight in this paper is to "observe force changes in a small window of time immediately following contact" to derive the compliance values.

This is the reason why motion and force were captured at a high rate.

The *exact moment of contact* is easy to determine from the change in joint torque (it changes sharply on contact).

Low degree polynomials (linear or quadratic) are fitted to the joint angle trajectory in small windows just before and just after the moment of contact. These smoothen the data.

Compliance is then calculated from the ratio of derivative of joint angle difference and derivative of torque. This follows roughly from (1):

$$C = \frac{d(\theta_2 - \theta_1)}{d(\tau)}$$

3.4 Reference Trajectory

Using the above values of compliance, torques and joint angles we can obtain the *reference configuration* using (1):

$$\theta_r = \theta - C\tau$$

The reference configuration is "the configuration of the finger had there been no contact."

We will use the above calculated values of C and θ_r to retarget the interaction to other surfaces and conditions.

3.5 Interaction Synthesis

The compliance C and the reference trajectory θ_r are together called the *interaction trajectory*. This allows us to resynthesize new interaction.

The motion thus synthesized exhibits realistic dynamic behaviour, in which the character's dynamics come from the motion capture, the environment dynamics come from a dynamic simulation, and "the captured compliance provides a biologically plausible interface between the two."

4 Validation

4.1 Task Dependence

Compliance values are lesser for scratching compared to exploring the surface. This is valid since stiffness of joints is more during scratching.

4.2 Estimates From Perturbation

The compliance values calculated from our method is compared to actual values measured using perturbation device. They are found to comply.

4.3 Limitations

The estimated compliance values need to be monitored for invalid values (sometimes they can be negative). These errors are caused due to:

1. *Motion capture noise* can sometimes hide small joint motions.

2. Compliance estimation gets difficult when the joints are nearly straight (i.e., the kinematic structure is close to singularity in the direction of force measurement).

3. It is also possible that low degree polynomials are insufficient for trajectory interpolation.

5 Force Measurement Methods

We used 6-axis force torque sensors. There are other options too:

1. Single axis pressure sensors (Ex: force sensitive resistors (FSR))

2. Capacitance based sensors (Ex: pressure sensitive sphere)

6 Interaction Capture In 3D

In this section, we describe how to extend the results from the previous sections to 3D articulated structures with branches and compliant joints, like human hands.

6.1 Kinematics And Dynamics In 3D

The 4×4 matrix ${}^{j}_{i}E$ denotes the homogeneous coordinates of frame i with respect to another frame j.

${}^{i}x$ denotes the homogeneous coordinates of a 3-dimensional vector x in frame i.

Then, the homogeneous coordinates of this vector in frame j is given by:

$$ {}^{j}x = {}^{j}_{i}E \, {}^{i}x $$

***

The *spatial velocity* ϕ describes the relative motion of a body with respect to the fixed world frame. In coordinates of frame i, it is given by the 6-size column vector ${}^{i}\phi$:

$$ {}^{i}\phi = ({}^{i}\omega^{T}, {}^{i}v^{T})^{T} $$

where,

ω is the *angular velocity* of the point at the origin of frame i

v is the *linear velocity* of the point at the origin of frame i

Spatial force called *wrench* is given by:

$$w = (\tau^T, f^T)^T$$

where,

τ is the *rotational torque*

f is the *translational force*

Spatial velocities and wrenches are transformed using the 6×6 adjoint transformation matrix ${}^j_i\mathrm{Ad}$:

$${}^j_i\mathrm{Ad} = \begin{pmatrix} \Theta & 0 \\ [p]\Theta & \Theta \end{pmatrix}, \text{where } {}^j_iE = \begin{pmatrix} \Theta & p \\ 0 & 1 \end{pmatrix}$$

where,

Θ is a 3×3 rotation matrix

p is a 3×1 displacement

$[p]$ denotes the skew symmetric 3×3 matrix equivalent to the cross product $p \times$

Spatial velocities being *contravariant quantities* are transformed by:

$${}^j\phi = {}^j_i\mathrm{Ad}\, {}^i\phi$$

Wrenches (are column vectors) being *covariant quantities* are transformed by:

$${}^jw = {}^j_i\mathrm{Ad}^T\, {}^iw$$

***

We define $\Gamma(r)$, a 3×6 matrix for a point r by:

$$\Gamma(r) = \begin{pmatrix} [-r] & I \end{pmatrix}$$

Given spatial velocity ϕ for a point r we can find the linear velocity by:

$$\dot{r} = \Gamma(r)\phi$$

Given linear force f acting at a point r we can find the wrench by:

$$w = \Gamma(r)^T f$$

***

The spatial cross product of ϕ is the linear operator with coordinate matrix:

$$[\phi] = \begin{pmatrix} [\omega] & 0 \\ [v] & [\omega] \end{pmatrix}$$

With this information we can write the *Newton-Euler equation for a rigid body* in body coordinates as:

$$(3)\colon w = M\dot{\phi} - [\phi]^T M\phi$$

where, M is the *mass inertia matrix*

***

For an articulated structure made up of rigid components linked by joints, the *Jacobian* provides a linear transformation from joint angle velocities to spatial velocities of a given link.

Hence, the Jacobian matrix J has columns where each column contains the spatial velocity due to an angular velocity of 1 rad/s at a joint.

Since the spatial velocity at i is the sum of twists from root to i (the rest being excluded) we introduce a diagonal binary matrix S_i to select only those required joints (columns) in J.

Thus, the spatial velocity of body i is given by:

$$\phi_i = JS_i\dot{\theta}$$

The linear velocity of a point r_i on body i is given by:

$$\dot{r}_i = J_i\dot{\theta}$$

where,

$$(4)\colon J_i = \Gamma(r_i)JS_i$$

What this means is:

$$\text{Joint angle velocity } (\theta) \overset{JS}{\longrightarrow} \text{Spatial velocity } (\phi) \overset{\Gamma(r)}{\longrightarrow} \text{Linear velocity } (\dot{r})$$

Similar to above, the Jacobian transpose is used to get wrenches and joint torques:

$$\tau = J_i^T f_i$$

***

For small joint displacements, J_i provides a linear approximation to the displacement of r_i:

$$\Delta u_i = J_i \Delta\theta$$

6.2 Effective Endpoint Compliance

A small force Δf_i when applied to the compliant kinematic structure at a contact point r_i results in a displacement Δu_i at the contact point.

The force maps to torques, joint displacements and displacements through J and C as follows:

$$\Delta f_i \xrightarrow{J_i^T} \text{Internal torques}\,(\tau) \xrightarrow{C} \text{Joint displacements}\,(\theta) \xrightarrow{J_i} \text{Displacements}\,(\Delta u_i)$$

The last step ($\theta \xrightarrow{J_i} \Delta u_i$) is due to the linear approximation at the end §6.1.

This effective compliance relationship is written as:

$$(5a)\colon \Delta u_i = J_i \mathrm{C} \mathrm{J}_i^T \Delta f_i$$

With N contacts on multiple links in the kinematic structure, the effective compliance becomes *coupled* due to shared joints on the path between the contacts and the root.

This relationship is described by the matrix:

$$(5)\colon G = J C J^T$$

where, $J^T = (J_1^T, \ldots., J_N^T)$ a block row matrix.

Grouping linear forces and displacements into block column vectors Δf and Δu respectively, the local linear model of the compliance is written as:

$$(6)\colon \Delta u = G \Delta f$$

This is nothing but a generalization of ($5a$) derived above.

7 Interaction Synthesis

7.1 Algorithm

The *state of the system* at time k is given by $(C^k, \theta_r^k, \phi^k, q^k, \theta^k, f_i^k, r_i^k)$ for $i = 1 \ldots . N$

where,

C^k, θ_r^k come from captured interaction

ϕ^k, q^k describe the spatial velocity and configuration of the environment

f_i^k are the forces

r_i^k are the contact locations

The *quasi-static compliant structure* is approximately maintained in equilibrium by:

$$(7)\colon \theta^k - \theta_r^k + \theta_{\text{err}} = \sum_i C^k J_i^{k^T} f_i^k$$

where,

θ_{err} is a small error in joint angles

J_i is the Jacobian for contact r_i and configuration θ^k (from (4))

We want to find:

1. f_i^{k+1} the forces that evolve the state of the rigid body forward (thus providing ϕ^{k+1} and q^{k+1} upon integration in time).

2. θ^{k+1} the joint angle configuration necessary to keep the compliant structure in equilibrium given new reference angles θ_r^{k+1}.

7.1.1 Friction And Breaking Contact

We first define a *friction cone* at the point of contact by defining a set of vectors as follows:

1. d_{i0} is the *outward unit normal* at contact point i.

2. A set of *unit tangent vectors* d_{ij}, $j = 1...2m$, such that

    i. $\|d_{ij}\| = 1$

    ii. $d_{i(2j-1)} = -d_{i(2j)}$, $j = 1...m$ that is, $d_{i1} = -d_{i2}, d_{i3} = -d_{i4}$ and so on.

We assemble the above vectors into 2 matrices:

1. $D_i = (d_{i0}, ..., d_{i(2m)})$

2. $D_{i\star} = (d_{i1}, ..., d_{i(2m)})$, a matrix of just the tangent vectors (i.e., excluding d_{i0})

***

The *Coulomb friction cone* is defined as the set of possible forces that can be supported by the frictional surface. We build a polyhedral approximation to this cone using D_i.

The *tangential friction force* allowed by Coulomb friction lies inside a circle in the tangent plane of radius $\mu_i \beta_{i0}$,

where,

μ_i is the *coefficient of friction*

β_{i0} is the *normal force* at contact i, i.e., $\beta_{i0} = d_{i0} f_i$

This tangential friction force can thus be approximated by the *convex hull* of tangent vectors d_{ij} scaled by the normal force β and the coefficient of friction μ:

$$\left\{ \sum_{j=1}^{2m} d_{ij}\beta_{ij} \text{ where } \beta_{ij} \geq 0 \text{ and } \sum_{j=1}^{2m} \beta_{ij} \leq \mu_i \beta_{i0} \right\}$$

Hence, we approximate the Coulomb friction at contact point i by:

$$f_i = D_i \beta_i, \beta_{i\star} \geq 0, F_i \beta_i \geq 0$$

where,

$\beta_i = (\beta_{i0}, ..., \beta_{i(2m)})^T$

$F_i = (\mu_i, -1, ..., -1)$

$D_i \beta_i$, the force at contact i is actually f_i^{k+1}, the force at the next step which is what we are solving.

***

The Coulomb friction approximation gives a constraint on the force, but we also need to constrain the motion to satisfy the *principle of maximum dissipation*. Maximization of dissipation implies a minimization of the negative frictional work:

$$\min_{\beta_{i\star}} \Delta\mu_i^T D_{i\star}\beta_{i\star}, \beta_{i\star} \geq 0, F_i\beta_i \geq 0$$

where,

$D_{i\star}\beta_{i\star}$ is the *friction force*

Δu_i is the motion at the contact point

We can rewrite this constrained minimization problem using Lagrange multipliers:

$$\mathcal{L} = \Delta u_i^T D_{i\star}\beta_{i\star} - v_{i\star}^T\beta_{i\star} - \lambda_i F_i\beta_i$$

The KKT (Karush-Kuhn-Tucker) optimality condition can be got by (partial) differentiation with respect to $\beta_{i\star}$:

$$0 = D_{i\star}^T\Delta u_i - v_{i\star} - E_{i\star}^T\lambda_i$$

where, $E_i = (0, -1, ..., -1)$

Rearranging terms in above equation we get:

$$v_{i\star} = D_{i\star}^T\Delta u_i - E_{i\star}^T\lambda_i$$

This KKT conditions can be rewritten as:

$$(8)\colon \beta_{i\star} \perp v_{i\star} = D_{i\star}^T\Delta u_i - E_{i\star}^T\lambda_i$$

$$(8)\colon \lambda_i \perp \sigma_i = F_i\beta_i$$

where,

$\perp$ is the *complementarity notation*, $a \perp b$ meaning $a \geq 0, b \geq 0, a^T b = 0$

$\sigma_i = F_i\beta_i$

***

Non-interpenetration needs to be addressed, i.e., the finger shouldn't get inside the rigid body. The separation of contact point i is given by v_{i0}:

$$(9a)\colon v_{i0} = d_{i0}^T\Delta u_i + s_{i0}$$

where,

s_{i0} is the linear distance between the closest points (or contact points) along the contact normal

For non-penetration, v_{i0} must be positive.

If we want *non-velcro forces* or no *contact forces* (i.e., finger doesn't stick to the surface), then we require the magnitude of the normal force to be positive:

$$(9b)\colon \beta_{i0} = d_{i0}^T f_i \geq 0$$

(9a) and (9b) are actually complementary to each other since contact forces exist only when the separation is 0.

Hence, combining (9a) and (9b) with (8) we can write the complementarity conditions at point i as:

$$(9)\colon \beta_i \perp v_i = D_i^T \Delta u_i - E_i^T \lambda_i + s_i$$

$$(9)\colon \lambda_i \perp \sigma_i = F_i \beta_i$$

where,

$v_i = (v_{i0}, v_{i\star}^T)^T$

$s_i = (s_{i0}, 0, \ldots, 0)^T$

7.1.2 Contact Point Motion

We define Δu_i, the contact point motion by combining the 3 sources of its motion:

$$(10)\colon \Delta u = G\Delta f + J\Delta\theta_r - (U_\phi f^{k+1} + u_\phi)$$

where,

1. $G\Delta f$ accounts for the movement of the contacts due to the change in contact forces ($\Delta f = f^{k+1} - f^k$).
2. $J\Delta\theta_r$ gives the motion of the contact points necessary to preserve equilibrium given the current forces and change in reference angles ($\Delta\theta_r = \theta_r^{k+1} - \theta_r^k$).
3. $(U_\phi f^{k+1} + u_\phi)$ is the motion of the contact point on the rigid body due to rigid body motion. Here,
   i. $U_\phi f^{k+1}$ describes the motion due to contact forces.
   ii. u_ϕ is due to the body's current spatial velocity and external forces.

***

The wrench on the body in body coordinates is given by:

$$w^{k+1} = -\,{}^w_b\mathrm{Ad}^T \Gamma^T f^{k+1}$$

where,

${}^w_b\mathrm{Ad}$ is an adjoint matrix that converts body coordinates to world coordinates ($b \rightarrow w$), so ${}^w_b\mathrm{Ad}^T$ does the vice versa.

Γ is a block column matrix of size $3N \times 6$ (for N contacts) with block i equal to $\Gamma(r_i^k)$

f^{k+1} is the sum of forces f_i^{k+1} at contact points r_i^k

By rearranging the Newton-Euler equation for a rigid body (3) to solve for acceleration, we get:

$$\dot{\phi} = M^{-1}(w^{k+1} + [\phi^k]^T M \phi^k + w_{\text{ext}})$$

where,

w_{ext} is the *external wrench* and includes forces such as gravity.

Taking one Euler step of size h, the new velocity we get is:

$$\phi^{k+1} = \phi^k + hM^{-1}(w^{k+1} + [\phi^k]^T M\phi^k + w_{\text{ext}})$$

The linear approximation of the motion of the contact points due to rigid body motion can be written by taking an additional Euler step and multiplying by $h\Gamma{}^w_b\text{Ad}$. This gives us the implicit linear approximation $U_\phi f^{k+1} + u_\phi$ where,

$$U_\phi = -h^2\Gamma{}^w_b\text{Ad}\, M^{-1}{}^w_b\text{Ad}^T\Gamma^T$$

$$u_\phi = h\Gamma{}^w_b\text{Ad}(\phi^k + hM^{-1}([\phi^k]^T M\phi^k + w_{\text{ext}}))$$

7.1.3 Linear Complementarity Problem

In Linear Algebra, the *Linear Complementarity Problem (LCP)* consists of starting with a n-dimensional column vector q (where $q \in \mathbb{R}^n$) and a $n \times n$ matrix M (where $M \in \mathbb{R}^{n\times n}$), and finding 2 n-dimensional vectors w and z such that:

1. $q = w - \text{Mz}$
2. $w_i \geq 0$ and $z_i \geq 0$ for each i
3. $w_i \times z_i = 0$ for each i (i.e., either $w_i = 0$ or $z_i = 0$)

***

We use the values derived in the above sections to build a Linear Complementarity Problem (LCP).

The complementarity conditions at each point given in (9) can be combined into a single system where the forces and displacements are coupled using (10). This results in a LCP of the form:

$$\begin{pmatrix} D^T(G - U_\phi)D & -E^T \\ F & 0 \end{pmatrix}\begin{pmatrix} \beta \\ \lambda \end{pmatrix} + \begin{pmatrix} s' \\ 0 \end{pmatrix} = \begin{pmatrix} v \\ \sigma \end{pmatrix} \perp \begin{pmatrix} \beta \\ \lambda \end{pmatrix}$$

where,

$s' = s + D^T(J\Delta\theta_r - Gf^k - u_\phi)$

$D = \text{diag}(D_1, ..., D_N)$

$E = \text{diag}(E_1, ..., E_N)$

$F = \text{diag}(F_1, ..., F_N)$

$s = (s_1^T, ..., s_N^T)^T$

$\beta = (\beta_1^T, ..., \beta_N^T)^T$

$v = (v_1^T, ..., v_N^T)^T$

$\lambda = (\lambda_1, ..., \lambda_N)^T$

$\sigma = (\sigma_1, ..., \sigma_N)^T$

This LCP can be solved using *Lemke's method*. It provides the contact forces f^{k+1} using which we advance the rigid body system by integrating (3) twice.

$$f^{k+1} = D\beta$$

We then use these forces to compute the new joint angles from the equilibrium equation:

$$\theta^{k+1} = \theta_r^{k+1} + C^{k+1}J^T f^{k+1}$$

Finally, we identify new contact points for the new configuration θ^{k+1} and body configuration q^{k+1}.