Eurographics/SIGGRAPH Symposium on Computer Animation (2003)
D. Breen, M. Lin (Editors)

Mapping optical motion capture data to skeletal motion using a physical model

Victor B. Zordan and Nicholas C. Van Der Horst[†]

University of California, Riverside

Abstract

Motion capture has become a premiere technique for animation of humanlike characters. To facilitate its use, researchers have focused on the manipulation of data for retargeting, editing, combining, and reusing motion capture libraries. In many of these efforts joint angle plus root trajectories are used as input, although this format requires an inherent mapping from the raw data recorded by many popular motion capture set-ups. In this paper, we propose a novel solution to this mapping problem from 3D marker position data recorded by optical motion capture systems to joint trajectories for a fixed limb-length skeleton using a forward dynamic model. To accomplish the mapping, we attach virtual springs to marker positions located on the appropriate landmarks of a physical simulation and apply resistive torques to the skeleton's joints using a simple controller. For the motion capture samples, joint-angle postures are resolved from the simulation's equilibrium state, based on the internal torques and external forces. Additional constraints, such as foot plants and hand holds, may also be treated as addition forces applied to the system and are a trivial and natural extension to the proposed technique. We present results for our approach as applied to several motion-captured behaviors.

Categories and Subject Descriptors (according to ACM CCS): I.3.5,I.3.7 [Computer Graphics]: Physically based modeling, Animation

1. Introduction

While motion capture is an effective means for animating realistic motions, the process of motion capture presents several technical issues that must be resolved. One difficult problem specifically related to optical motion capture for skeleton-driven character animation is the non-trivial mapping of the markers, moving in Cartesian 3D-space, to a relative motion representation defined by joint angles plus a body center or root. In some cases, in addition to joint angles, scales in bone lengths are permitted to increase the fit of the skeletal motion to the raw marker data but this correction is not always desirable as it may be difficult to apply seamlessly in a given character.

In this paper we address the mapping problem going from 3D markers to a defined skeleton using a constrained three-dimensional physical model controlled to follow the Cartesian-based marker data. The metaphor of a dance instructor or a coach physically adjusting the posture of a dancer or player lead us to the novel solution for mapping presented here. Specifically, the marker data guide the physical simulation of the character to the closest posture for a given sample resolved from external forces and internal joint actuation. We consider the mapping problem to be separate from the retargeting problem as well as skeleton estimation. With this assumption, we focus our efforts on mapping to a chosen, fixed limb-length skeleton.

Popular tools used for mapping in production animation are commercial software packages such as Kaydara's Motionbuilder (formerly Filmbox) [4] and software packaged with motion capture equipment, such as Vicon's Bodybuilder [19]. While primarily undocumented, we believe these systems use inverse kinematics (IK) based approaches which often leave indicative side effects, such as knees and elbows that never fully extend. These systems are often unintuitive to control and lead to unexpected solutions due to ad hoc heuristics, such as the head turning upside down when an

---

[†] email:vbz|nvanderh@cs.ucr.edu

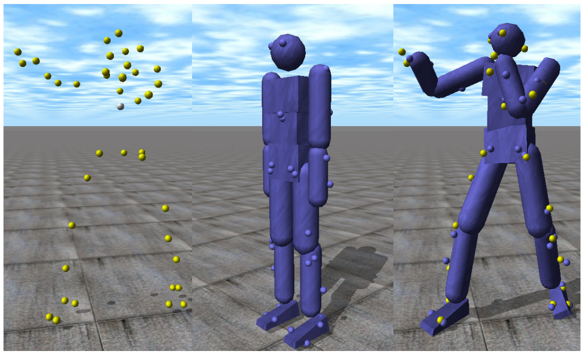

Figure 1: ***Marker data, simulation with "virtual" markers, and resolved initial posture.***

inversion is performed, presumably because the head is constrained to remain upright. Our physically based mapping approach has advantages over priority-based [1, 15] or simplified analytical [6] IK solutions. In particular, the system simultaneously finds the whole-body posture including joint angles and root body estimates without compromising end-effector placement by automatically reducing penalty errors according to all the recorded data of the sample. The approach does not need hand-crafted heuristics, does not calculate intermediate body orientations or joint centers and does not need to give priority to any specific markers, such as those associated with the pelvis seen in many root-centric approaches.

1.1. Related Work

Most graphics-related research topics in modification and synthesis with human motion capture have focused on editing data for modifications at the behavior level, as in [3, 6, 13, 2, 15, 7] among others. Although their focus is on modifying behaviors, many of the efforts use data that has already been mapped as joint angle data, or perform mapping as a pre-step utilizing numerical methods to solve redundancy in highly articulated figures. Thus our work fits nicely with many of the approaches proposed. In general the mapping problem is under-constrained and optimization requires additional metrics to find a unique posture. Such ad hoc heuristics often include errors for end-effector and/or joint center positions or for body or joint angle variation. Optimization-based IK approaches are not new for character animation, in general, or for skeletal mapping of motion capture [21, 1].

Some researchers have focused on problems related to mapping, mainly for skeleton creation and modification for various forms of motion data [9, 8, 10]. Monzani and colleagues use an IK-based approach with a reduced set of degrees of freedom to find motion for characters of varying size. O'Brien and his colleagues offer a solution to the problem of estimating the joint skeleton for magnetic data [10] as does Silaghi et al. for optical marker data [16]. These efforts compliment the work described here by offering methods for generating reasonably sized skeletons directly from the data. Kovar et al. focus on an isolated problem of mapping for foot placement, a problem dubbed 'footskate', offering a hybrid solution that modifies joint angles for the leg and allows a small amount of leg scaling [5]. A slightly different variation on the mapping problem is proposed by researchers interested in applying motion capture to skeletal forms for humanoid robotics with limited ranges of motion [18, 12]

Several physical models have been introduced to modify motion capture. For example, motion may be generated to minimize energy for transitioning between motion sequences [14]. In comparison, Rose et al. use an inverse dynamic, rather than forward dynamic solution. Popović and Witkin use a simplified physical model to transform motion sequences [13] with spacetime constraints. Our previous work focuses on tracking joint angle trajectories [22] with a physical model simulating impacts and reactions for motion capture-driven characters. Perhaps the most closely related work to this uses a linear dynamic system to constrain motion for synthesizing motion textures [7]. Their simple control approach is similar to ours but they focus at the behavior level using joint angles that have already been mapped. Westenhofer and Hahn also offer a similar approach for modifying keyframe data, using springs to control a simulated clone, in order to incorporate dynamic effects in the final animation [20].

2. Guiding posture with force

When a child is taught to bat a ball, a coach angles her shoulders, points her head and gives her verbal pointers on how to swing. Physical adjustment is a useful instructional tool for swinging a bat, for properly aligning a leg or an arm in ballet, and for many other training exercises involving the physical body. From these examples, we introduce the notion of "force as interface" i.e. that external forces may act as a reasonable guide for aligning skeletal postures.

2.1. Optical marker data

Marker motion capture data from optical systems is distilled into a set of 3D Cartesian position traces. The 3D data is easy to manipulate because it is rotationless and is accurate, with most current commercial optical systems boasting a range of millimeters for medium-size markers in a multi-meter sized capture region. However, the data also includes errors making mapping more complicated. Errors stem from: poor calibration and system noise; marker movements from unintentional motion of cloth or skin; and other artifacts such as articulation that is captured but not modeled in the rigid-body skeleton, like the shrugging of a collar bone or the squeezing of cartilage between bones at a joint. Thus, mapping to a character with a specific set of degrees of freedom, even a close representation in the size of the human actor, can be problematic, leading to for instance accumulated errors that especially propagate in the end effectors, creating large mismatch and footskate especially in root-centric solutions.

2.2. Force as interface

To map optical data to a skeleton, our system treats all markers with equal weight and applies external forces in response to consecutive samples of motion data. Starting with an initialization process, we position markers from a known marker set on the simulation, specifically we selected the common RTK marker set. Interactively, we "glue" each marker on the simulation where the approximate landmarks should appear. The initialization is completed by applying forces that move the simulation from its starting posture to the initial pose of the motion sequence. Subsequently, each data sample uses the previous posture as its initial state. The system automatically updates new marker positions and the character is simulated forward in time until it comes to rest at a new pose for that sample. Once the simulation has stopped moving within a set threshold, the posture is recorded and the next sample begins.

A combination of external forces act on the skeleton simulation while it is simulated between samples. Specifically, the simulation feels the following forces:

$$F_{external} = F_{marker} + F_{damping} + F_{contact} \qquad (1)$$

at each timestep. Spring forces, as $F_{marker} = -kX$, pull between the motion capture data points and the simulation's equivalent markers with stiffness k. To speed up the process, each body part is damped, as $F_{damping} = -bV$, based on the body's global velocity, V. This damping allows a relatively large simulation timestep to be taken, around 0.01 s in practice. Additional forces, $F_{contact}$, may be applied optionally for ground collision constraints as described in the next section.

The simulation responds to the sum of the external forces and to its own control system to achieve each recorded posture. The force computation yields two values, k and b, that must be selected by hand though a wide range of values appear adequate from our inspection. For markers seated close to two bodies, for example the knee position, springs are added to influence both nearby bodies, in this case the lower and upper leg, to aid in orienting the attached bodies.

2.3. Internal actuation

Opposing the external forces, a simple feedback controller acts to give the skeleton some rigidity (backbone, if you will). Internal control for the physical system is used to dictate constraints based on a neutral or "comfortable" position for the character. Torques are computed as

$$\tau = k_t\ (\theta_d - \theta) - b_t\ (\dot{\theta}) \qquad (2)$$

where θ and $\dot{\theta}$ correspond to the simulation's current joint angles and joint velocities, θ_d is the desired joint angle, and k_t and b_t are the gain and damping terms, respectively. This controller does not set explicit joint limits but, instead, relies on the underlying human motion to avoid limit violations. We set the desired value as the neutral posture, a simple arms-at-side standing posture. Inertia-scaled gains [22] act to reduce the number of parameters yielding two single torsional stiffness and damping values that must be tuned.

The effect of this controller is to keep the skeleton close to its neutral posture while the controller resists joint motion through damping. With the controller, the physical system behaves like a springy toy or mannequin that is easy to bend into different postures though it has a preferred state for each of its joints. This preference helps by adding a temporally coherent bias when the mapping would be otherwise ambiguous. The collective force and torque inputs move the skeleton to a valid but "comfortable" posture according to the 3D marker position data, the kinematic skeleton and the neutral pose. To resolve a posture from the collective inputs, the forces pull the body toward a lower error posture, locally tugging the marker position on the character's body toward the marker recorded position while the damping and internal actuation act to resist motion. With a reasonable amount of damping, the system quickly slows to rest at an equilibrium posture for each sample.

3. Additional constraints

To correct footskate, our system is easily adapted to include foot contact using a data-driven contact model. We implement a penalty-based ground constraint to add friction and correct foot penetration with the ground plane. We use the absolute value of the congregate velocity for the markers, V^*_{foot}, on each foot to allow the system to discern (or "tag") when the foot is planted or not, automatically. If the foot is determined to be in contact, friction forces are applied. Unidirectional ground reaction forces correct the foot by pushing it above the surface of the ground plane regardless of whether the foot is found to be in motion or not:

$$F_{contact} = \begin{cases} F_z & V^*_{foot} \geq \alpha \\ F_z + F_{fric} & V^*_{foot} < \alpha \end{cases}$$

where α is the threshold for slipping, F_z pushes up using a unidirectional spring based on the ground penetration error and F_{fric} is computed as μF_z and applied in the xy-plane in the direction opposing the motion of the simulated foot at the point of contact. To create smooth contact that minimizes chatter, an array of such contact forces are spread uniformly along and across the bottom of the foot model.

Poor foot contact is a common ailment of motion capture mapping and retargeting that merits its own attention [5]. Kovar and his colleagues present a stand-alone solution to footskate with comparisons of their work to the commercial alternative, Filmbox. In our system, adding such corrections via forces allows us to offer a complete solution to the mapping problem within a single framework. Similar techniques using ground contact models have been used for editing motion capture data previously [11]

4. Examples and implementation

The dynamic simulation software used in this work is Open Dynamic Engine, a free shareware tool [17]. The articulation we chose includes 39 independent degrees of freedom, with three degree-of-freedom ball-and-socket joints for ankles, knees, hips, shoulders, elbows, neck, and two joints in the trunk. We selected a skeleton with the proportions of an average, slender male and scaled the system to be the same height as the motion capture actors. No other specialized adjustments were made to fit the skeleton to the recorded human's kinematics or dynamics. Mass and inertial parameters are determined by default within ODE based on the geometric models for the bodies. Two example sequences from different actors are considered for martial arts fighting and simple walking. A small number of internal parameters and the marker placement were selected by hand for the martial arts examples in a straightforward, consecutive manner and the system required no re-tuning of these values to create the walking sequence.

4.1. Martial arts

The fighting examples shown contain an array of rich behaviors including a variety of kicks, punches and elbow strikes as well as some upper-body blocking defenses, see Figures 3 and 4. The diverse set of examples make the repertoire ideal for showing the power of our approach. Motions include highly irregular, fast-paced motion and quick and subtle footwork. Through the examples we see that the system was able to find postures with believable joint angles for a large range of different marker positions.

4.2. Walking

While walking motion is more tame and slow-paced than the fighting examples, walking behaviors commonly yield footskate artifacts from the mapping process. Our example includes the foot constraining forces and shows crisp footplants that are aligned with the original data, see Figure 5. Using the foot constraint, the system automatically identifies when the foot is stationary or not, re-orients the foot to avoid penetration with the ground plane, applies friction to resist motion as appropriate while maintaining the subtle motion of the foot.

5. Conclusions

We present a straightforward general approach to a complex mapping problem that can be implemented easily using available software. A small number of parameters were hand tuned to build such a system but it is our intention to make a version of the software freely available so that others can avoid this tuning process. The resulting posture motion is smooth, as can be seen in Figure 2 which shows a time-lapsed rendering of a series of karate kicks.

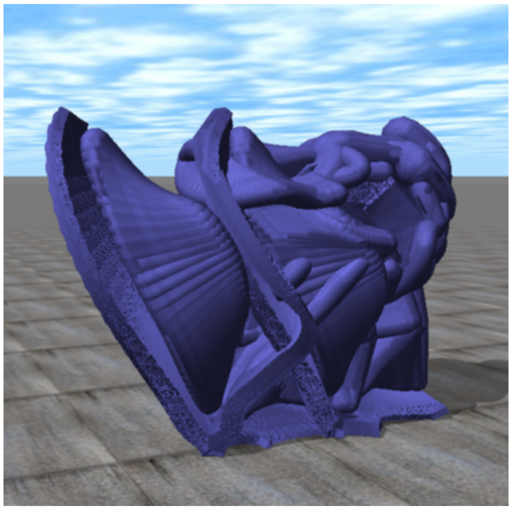

Figure 2: ***Time-lapsed sampling of motion from two consecutive karate kicks. Notice how uniform and smooth the final posture samples appear in kicking leg to the left of the image.***

While it was unnecessary for creating the examples shown, several additions to the system seem appealing. Inherently, the technique trivially overlooks small glitches and outliers in the data because each spring's effect is small, relying little on each individual marker. But, to further ensure that outliers are not problematic, we could add a feature in which the spring turns off if its magnitude hits some threshold. If caught before being applied such forces can be prevented from effecting the posture. Second, we have not implemented joint limits, instead relying on the motion capture data to prevent undesirable postures. But with internal control applied at each joint, it seems reasonable to add joint limits explicitly as well.

Though effective, the approach described has limitations. Our system does not solve the skeleton estimation problem and would likely benefit from a working skeleton estimator. An important consideration when mapping 3D positions to a skeleton is whether the scaling of limbs is permissible. It is often not. With respect to oddly-shaped or highly non-humanlike characters, we believe a secondary process would be beneficial. While our system can sustain and correct for differences between the skeleton and the recorded actor's kinematics, mapping first to a close approximation would give a good starting point for subsequent retargeting. Because mapping is usually an offline process, speed does

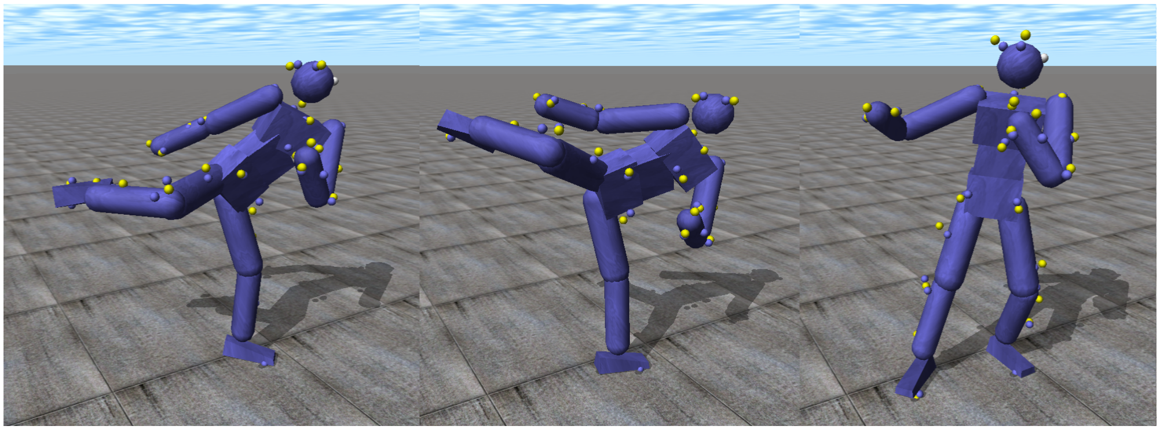

Figure 3: ***Close up stills showing marker placement (lighter spheres show motion capture, darker are virtual markers).***

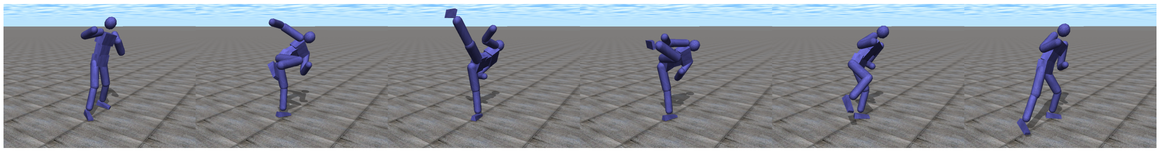

Figure 4: ***Filmstrip of a kicking motion, frames spaced at 0.17 sec.***

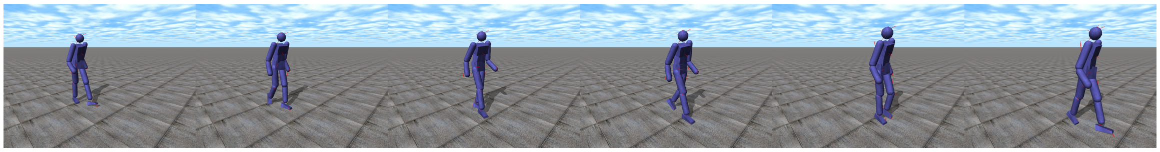

Figure 5: ***Filmstrip of walking motion, frames spaced at 0.50 sec.***

not present a problem. The system current runsat interactive rates around 2-3 frame/second on a 2.4 GHz Pentium IV processor. However, a true real-time system would be more ideal because it would allow for real-time motion capture animation of the skeleton.

Among the benefits of our approach is that it is not tied to a single marker set and any of many markers sets may be used as long as a template is set up. This set-up entails placing the markers on the virtual landmarks or closest approximations on the character skeleton. This is the most labor intensive aspect of the approach described. However, for each new marker set, this process must only be repeated once. Motion capture using a marker set that targets this technique would be ideal. Such a custom marker set would evenly disperse markers on the bodies aiding the mapping system. Also, the approach does not rely on any individual data specifically but finds solutions as the congregate of all the markers for a given sample, smoothing large full-body disturbances over time.

Optical systems for motion capture are becoming increasingly popular. And data in the form of Cartesian motion data from such systems must be mapped to be useful, not just by commercial products but by programmers in many a research lab and production house who wish to use novel marker sets or custom systems. Whether in products supplied to consumers or for users “cleaning" their own data, this simple physically based approach will be a useful tool for those interested in reliably generating reasonable joint angles from 3D position trajectories.

Acknowledgements

Special thanks to the reviewers who helped improve this work and its presentation, to Steve Park for his help capturing the motion data, to Clint Hidinger and Joe Tullio for their help as motion talents, for the supportive commentary from those special members at GIG and, not least, for the support and software of Russell Smith et al. at ODE.

References

1. B. Bodenheimer, C. Rose, S. Rosenthal, and J. Pella. The process of motion capture: Dealing with the data. In *Computer Animation and Simulation '97*, pages 3–18. Eurographics, Springer-Verlag, Sept. 1997.

2. K.-J. Choi and H.-S. Ko. Online motion retargetting. *The Journal of Visualization and Computer Animation*, 11(5):223–235, Dec. 2000.

3. M. Gleicher. Retargeting motion to new characters. In *Proceedings of SIGGRAPH '98*, pages 33–42. ACM SIGGRAPH, July 1998. Held in Orlando, Florida.

4. Kaydara. Kaydara's motionbuilder. WWW Site, 2003. http://www.kaydara.com/products/motionbuilder/.

5. L. Kovar, J. Schreiner, and M. Gleicher. Footskate cleanup for motion capture editing. In *ACM SIGGRAPH Symposium on Computer Animation*, pages 97–104, July 2002.

6. J. Lee and S. Y. Shin. A hierarchical approach to interactive motion editing for humanlike figures. In *Proceedings of SIGGRAPH '99*, pages 39–48. ACM SIGGRAPH, Aug. 1999. Held in Los Angeles, California.

7. Y. Li, T. Wang, and H.-Y. Shum. Motion texture: A two-level statistical model for character motion synthesis. *ACM Transactions on Graphics*, 21(3):465–472, July 2002.

8. T. Molet, R. Boulic, and D. Thalmann. Human motion capture dirven by orientation measurements. *Presence*, 8(2):187 – 203, 1999.

9. J. S. Monzani, P. Baerlocher, R. Boulic, and D. Thalmann. Using an intermediate skeleton and inverse kinematics for motion retargeting. *Eurographics Eds. M. Gross and F. R. A. Hopgood*, 9(3), 2000.

10. J. O'Brien, R. Bodenheimer, G. Brostow, and J. Hodgins. Automatic joint parameter estimation from magnetic capture data. *Graphics Interface 2000*, 2000.

11. N. S. Pollard and F. Behmaram-Mosavat. Force-based motion editing for locomotion tasks. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2000.

12. N. S. Pollard, J. K. Hodgins, M. J. Riley, and C. G. Atkeson. Adapting human motion for the control of a humanoid robot. *International Conference on Robotics and Automation, 2002*, 2002.

13. Z. Popović and A. Witkin. Physically based motion transformation. In *Proceedings of SIGGRAPH 99*, pages 11–20. ACM SIGGRAPH, Aug. 1999. Held in Los Angeles, California.

14. C. Rose, B. Guenter, B. Bodenheimer, and M. F. Cohen. Efficient generation of motion transitions using space-time constraints. In *Proceedings of SIGGRAPH '96*, pages 147–154. ACM SIGGRAPH, Aug. 1996. Held in New Orleans, Louisiana.

15. H. J. Shin, J. Lee, M. Gleicher, and S. Y. Shin. Computer puppetry: An importance-based approach. *ACM Transactions on Graphics*, 20(2):67–94, Apr. 2001.

16. M.-C. Silaghi, R. Plankers, R. Boulic, P. Fua, and D. Thalmann. Loal and global skeleton fitting techniques for optical motion capture. *CAPTECH '98: Modeling and Motion Capture for Virtual Environments*, 1537:26–40, Nov. 1998.

17. R. Smith. Open dynamics engine, 2003. http://www.q12.org.

18. A. Ude, C. Mann, M. Riley, and C. G. Atkeson. Automatic generation of kinematic models for the conversion of human motion capture data into humaniod robot motion. *IEEE Humanoids 2000*, Sept 2000.

19. Vicon. Vicon's bodybuilder. WWW Site, 2003. http://www.vicon.com/products/bodybuilder/.

20. W. Westenhofer and J. Hahn. Using kinematic clones to control the dynamic simulation of articulated figures. In *Computer Graphics International 1996*, 1996.

21. J. Zhao and N. Badler. Inverse kinematics positioning using nonlinear programming for highly articulated figures. 13(4):313–336, 1994.

22. V. B. Zordan and J. K. Hodgins. Motion capture-driven simulations that hit and react. In *ACM SIGGRAPH Symposium on Computer Animation*, pages 89–96, July 2002.