

Attaching Physiological Effects to Motion-Captured Data

Taku Komura

Image Information Division, Advanced Computing Center

RIKEN, Japan

Yoshihisa Shinagawa

Department of Information Science, Graduate School of Science,

University of Tokyo, Japan

Abstract

Today, using motion capture devices is the most common way to create realistic human motion data. In addition to that, various methods have been proposed to edit, morph and retarget such kind of motion. However, there are still few methods to add physiological effects to motion which are caused by fatigue, injuries, muscle training and muscle shrinking. This is because the innate structure of the human body, such as the musculoskeletal system, has been mostly neglected when handling human motion in computer graphics. In this paper, we propose a method to use the musculoskeletal system of the human body for editing and retargeting human motion which were captured using a motion-capture device. Using our method, not only physiological effects such as fatigue, or injuries but also physical effects caused by external force can be added to human motion. By changing the muscular parameters and size of the body, it is also possible to retarget the motion to different bodies such as a very trained muscular body, weak and narrow body, or a small childish body.

Key words: muscle-based model, motion conversion, retargeting, motion capture, human animation

1 Previous Work

To make animation more realistic, methods based on physical simulation have been developed. However, a new problem that arises when controlling a human body model in a physical environment is that the animator must describe the changes in the torque and force applied to the model, instead of the kinematic trajectories. This is a difficult task because the effect of changing each dynamic parameter is not obvious. To resolve this problem, a method known as proportional-derivative (PD) control has been developed by many researchers. This approach has been used to simulate gaits [13, 19] and athletic movements such as running, jumping and cycling [8].

Another approach is by spacetime constraints [22] which treats a motion synthesis problem by constrained

optimization. Keyframe animation techniques based on this method [3, 15, 10, 12] have been developed.

A different way to obtain human motion data is to use real human data. Motion capturing devices are used in such a case. If the animator cannot find the desired motion, it is necessary to apply some kind of editing or modification to the available data. Even if an exact motion can be found, if the size of the body performing the action does not match that of the character, the motion must somehow be adapted to the character which the animator wants to control. For this reason, there is an increasing demand to edit, convert, and retarget real human motion data. Previous techniques to create human animation automatically have often utilized to achieve this.

Gleicher [6] has succeeded in retargeting motion data to a character with a different body size using space-time constraints.

Popovic *et al.* [17] have proposed a dynamics-based method to edit human motion data, which enabled animators not only to add constraints to the original motion, but also to change such physical parameters as the mass of the body segments or gravity which could not be done by previous kinematic methods. Their method is based on both PD control and spacetime constraints.

However, precise human body models such as the musculoskeletal models have rarely been used to yield or edit human motion data [10, 12]. Chen *et al.* [2] have created a very precise muscle model using FEM, but it has not been used for the control of human body models in a dynamic environment. Pandy *et al.* [16] have used a musculoskeletal model to simulate maximum-height jumping. Their method is based on forward dynamics and optimal control. However, since no feedback controller is incorporated into this method, it is not suited to edit actual human motion, because the lack of a feedback controller makes it difficult to handle motion such as a cyclic gait using forward dynamics.

To add physiological effects to motion, it is necessary to use a precise human body model such as the musculoskeletal model. Even though Popovic *et al.* [17] suc-

ceeded to create a limp motion from gait data, it was done by reducing the degree of freedom of the knee. As a result, the obtained motion is different from a motion induced by physiological effects such as injuries.

2 Outline of the algorithm

In this paper, we propose a method to convert a captured motion dynamically and physiologically. For this purpose, a musculoskeletal human body model was prepared. The outline of the resulting algorithm is as follows:

1. As the initial motion data is prepared, the size of the musculoskeletal model is tuned to the size of the body of the motion.
2. Physical and physiological parameters of the body are edited: Muscle parameters such as the effect of fatigue or the peak force that can be exerted by a muscle are changed to create a physiologically different motion. Dynamic parameters such as gravity or external force can also be changed to yield a physically different motion. Parameters such as the length or width of the body segments are changed to create motion by a different body.
3. Since the values of physical and physiological parameters have been changed, the first motion is no longer feasible by the musculoskeletal human body model. Therefore, the conversion algorithm is applied to the motion to yield one that is feasible by the musculoskeletal model with the new physical parameters and muscle parameters. Thus, motions under different physiological conditions, such as tired motion, motion by an injured body, or motion under a different physical environment can be obtained.

3 Musculoskeletal Model

In order to generate human motion based on the anatomy and physiology of a real human body, it is necessary to have a musculoskeletal human body model. The model used in our experiments has been generated from a number of sources.

For the legs, the model by Delp [4, 1] has been used. This data includes the attachment sites of 43 muscles on each leg and physiological parameters such as the muscle's fiber length, the tendon's slack length, maximum force extertable, etc. The lower half of the body (Figure 1) is composed of the pelvis, and the femur, tibia, patella, talus, calcaneous, and toes in each leg. The joints of the legs are assumed as either a 3-DOF gimbal joint (hip joint) or as a 1-DOF joint (knee, ankle, and calcaneous joint). The data of Veeger and of Van Der Helm *et al.* [7] have been used for the shoulder, and

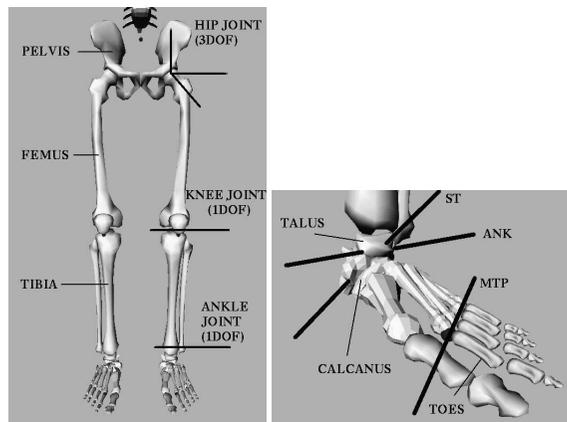


Figure 1: The rigid-body segments of the legs and joint axes. The ankle (ANK), subtalar (ST), and metatarsophalangeal (MTP) joints are modeled as pin joints with the axes shown

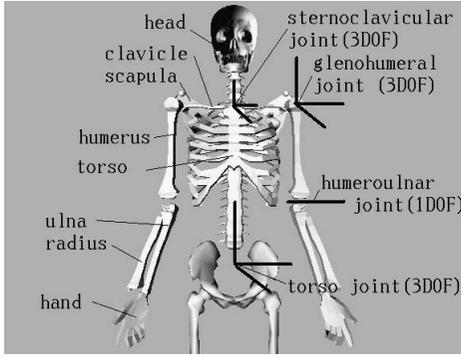
the data of Veeger *et al.* [18] has been used for the elbow. The upper half of the body (Figure 2) is composed of the torso, head, scapula, clavicle, humerus, ulna, radius, and the hand. The joints in the upper half of the body have been assumed to be either of 3-DOF gimbal type (torso, sternoclavicular and glenohumeral joint), 2-DOF type (acromioclavicular joint), 1-DOF type (humeroulnar), and sliding type (scapulothoracic gliding plane). There are 20 muscles on each arm, divided into a total 95 muscle elements.

We removed some DOFs from the parameters for optimization, which were difficult to handle (such as the radio-ulnar and acromioclavicular joint, and the scapulothoracic gliding plane) and which were considered not important for motion conversion(head and hand joints), to ease the convergence of the criteria. As a result, the total DOF number is 34. The front and back views of the body model with muscles are shown in Figure 3.

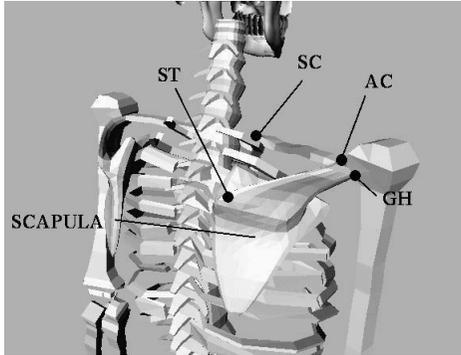
4 The Muscle Model

Each musculotendon is based on Hill's three component model (Figure 4). There are many muscle models which derive from Hill's model [21]. The model used here is that of Delp *et al* [4]. The model is composed of three elements: the contractile element (CE, muscle fibers), the parallel elastic element (PEE, connective tissue around the fibers and fiber bundles), and the series elastic element (SEE, muscle tendon).

Precise description of Hill's muscle model can be found in references such as [4, 12]. A muscle exerts its maximum force f^{max} when the muscle activation level a is 1 and the minimum force f^{min} when a is 0. Therefore,



(a)



(b)

Figure 2: (a) The rigid-body segments and joints of the upper half of the body. (b) The backward view of the upper body. The articulations listed are the sternoclavicular joint (SC), the acromioclavicular joint (AC), the glenohumeral joint (GH), and the scapulothoracic gliding plane (ST).

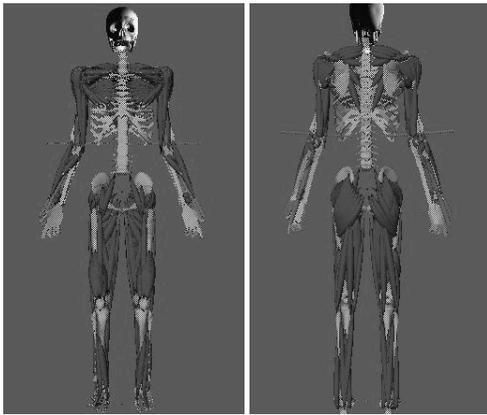


Figure 3: The frontal (left) and rear (right) views of the human body model

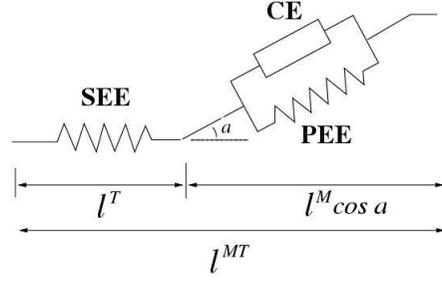


Figure 4: The muscle model by Hill used in this study

the musculotendon force, f^T , is limited by:

$$f^{min}(a = 0) \leq f^T \leq f^{max}(a = 1). \quad (1)$$

When a muscle exerts a large amount of force, fast glycolytic (FG) fibers in the muscle are recruited, which causes the intra-cellular pH level inside the muscle to decline. This causes the maximum amount of force exertable by the contractile element to decrease. This is called the *fatigue phase*. When the muscle is not used, the pH level increases, and the exertable force increases during the *recovery phase*. We also implemented the *fatigue and recovery* model of the muscle proposed by Giat *et al.* [5] (which is explained in the Appendix) for creating tired motion.

4.1 Changing the size of the body

The size of each body segment can be edited by scaling its size. The new inertia and mass of the segment are calculated using the scaling factor.

The new volume of muscle j ($v^{m_j l}$) is then calculated by the following equation:

$$v^{m_j l} = \sum_i v^{m_{ij}} s^i \quad (2)$$

while i is the counter of body segments over which muscle j crosses, s^i is the scaling factor of segment i , and $v^{m_{ij}}$ is the volume amount of muscle j included in segment i . It is also possible to set the proportion of the body by specifying the age. The data from [9] are used in that case.

Next, muscle parameters must be recalculated using the new size of the body. Variables which must be recalculated are those such as the optimal length of the muscle fiber l_o^M , the maximal force value f_o^M , and the tendon length when slack l_s^T .

A standing posture is used to determine these values. Using the original musculoskeletal model, the musculotendon length l^{MT} of each muscle at the standing posture is calculated. After the size of the human body model is

edited, again the musculotendon length of each muscle at the standing posture is calculated as $l^{MT}l$. The new optimal muscle fiber length l_o^Ml and tendon slack length l_s^Tl are obtained by

$$l_o^Ml = \frac{l^{MT}l}{l^{MT}} l_o^M \quad (3)$$

$$l_s^Tl = \frac{l^{MT}l}{l^{MT}} l_s^T. \quad (4)$$

Since the maximal force value f_o^Ml is proportional to the average cross sectional area, it can be obtained by

$$f_o^Ml = \frac{PCSA_l}{PCSA_o} f_o^M \quad (5)$$

while $PCSA$ is the average cross sectional area which can be obtained by

$$PCSA = \frac{v^{m_j}}{l_o^M}. \quad (6)$$

4.2 Balance

To maintain the balance in the human body model, it is necessary to define a function that evaluates the stability of the posture. The ‘zero moment point’ (ZMP) can be used to define such a stability function. When a human stands, on one foot or on both of the feet, a point exists where the moment applied to the body from the ground is zero. When the body is supported by a single leg, this point is at the sole of the support foot, but when the body is supported by both legs, it is in an area surrounded by the feet [20]. Since there is no joint between each foot and the ground, the moment that can be generated between the sole and the ground is limited. If the moment exceeds that limit, the body will fall to the ground. One way to check this constraint is by calculating the ZMP of the support to see if it is within the support area.

If the ZMP is within the support area, the posture is stable. However, if the ZMP is outside the support area, an additional moment must be added to the support foot to prevent the body from falling down. In our model, the value of this additional moment is used to evaluate the postural stability:

$$s(\theta, \dot{\theta}, \ddot{\theta}) = \begin{cases} \tau_+ & (\text{ZMP outside support area}) \\ \mathbf{0} & (\text{ZMP within support area}) \end{cases} \quad (7)$$

where τ_+ is the minimum external moment that must be added to the support foot to achieve stability in the specified posture.

5 Conversion of motion by the musculoskeletal model

In this section, the algorithm for motion conversion using the musculoskeletal model is explained.

The trajectories of the position of the body and the joint angles (θ) are represented by cubic B-spline functions. Using the kinematic values, $\theta, \dot{\theta}, \ddot{\theta}$, the joint moment can be calculated by inverse dynamics:

$$\tau = \mathbf{f}_1(\theta, \dot{\theta}, \ddot{\theta}). \quad (8)$$

It is necessary to evaluate at each moment whether each muscle can perform the motion in question. This evaluation is made by solving the following quadratic program:

$$\begin{aligned} & \min_{\mathbf{f}, \tau_{ext}} \|\tau_{ext}\|^2 \\ & \text{subject to } \begin{cases} \tau = \mathbf{A}\mathbf{f} - \tau_{ext} \\ \mathbf{f}^{min} \leq \mathbf{f} \leq \mathbf{f}^{max} \end{cases} \end{aligned}$$

where $\mathbf{f} = (f_1^T, \dots, f_{n_m}^T)$, $\mathbf{f}^{max} = (f_1^{max}, \dots, f_{n_m}^{max})$, $\mathbf{f}^{min} = (f_1^{min}, \dots, f_{n_m}^{min})$, n_m is the number of muscles, \mathbf{A} is the matrix that converts the norm of muscle force to joint torque, τ is the joint torque calculated by inverse dynamics using equation (8), and τ_{ext} is the supplementary torque which is applied when the motion cannot be achieved only by the muscle force. \mathbf{A} can be calculated from the moment arm, origin and insertion points of the muscles as explained in [10, 11, 12].

The human body model must also satisfy the balance constraint defined in Section 4.2. For the motion to be feasible, both $\|\tau_{ext}\|^2 = 0$ and $\|s(\theta, \dot{\theta}, \ddot{\theta})\|^2 = 0$ must be satisfied throughout the motion. Therefore, the feasibility of the motion is defined by

$$J = \int_{t_0}^{t_f} \|\tau_{ext}\|^2 + \|s(\theta, \dot{\theta}, \ddot{\theta})\|^2 dt. \quad (9)$$

Now, we calculate a physiologically feasible motion by minimizing J . The variables of the optimization process are the coefficients of the basis function and the time at which each motion terminates. The BFGS algorithm is used to optimize this function.

6 Experimental Results

In this section, a number of real human motions are physiologically converted by applying our method. The motions of the body were obtained by a magnetic motion capturing system. Unless the raw data were physiologically feasible, the conversion algorithm explained in the previous section was first applied to all the motions. After changing the muscle parameters or physical parameters, the conversion algorithm was applied to each motion again to obtain the final motion.

Two different kicking motions and a gait motion were first captured, and then new motions were obtained by changing the muscle parameters and dynamic parameters.

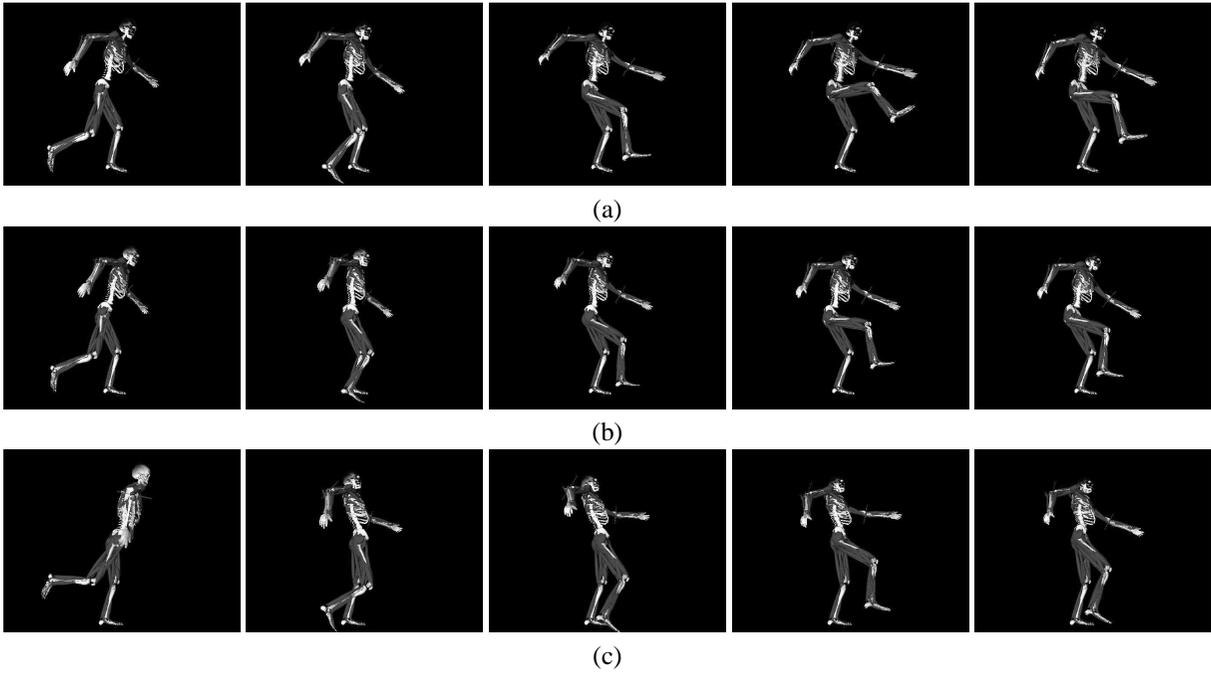


Figure 5: Conversion of the kicking motion: (a)initial motion, (b)converted motion after 20 repetitions, and (c) after 50 repetitions.

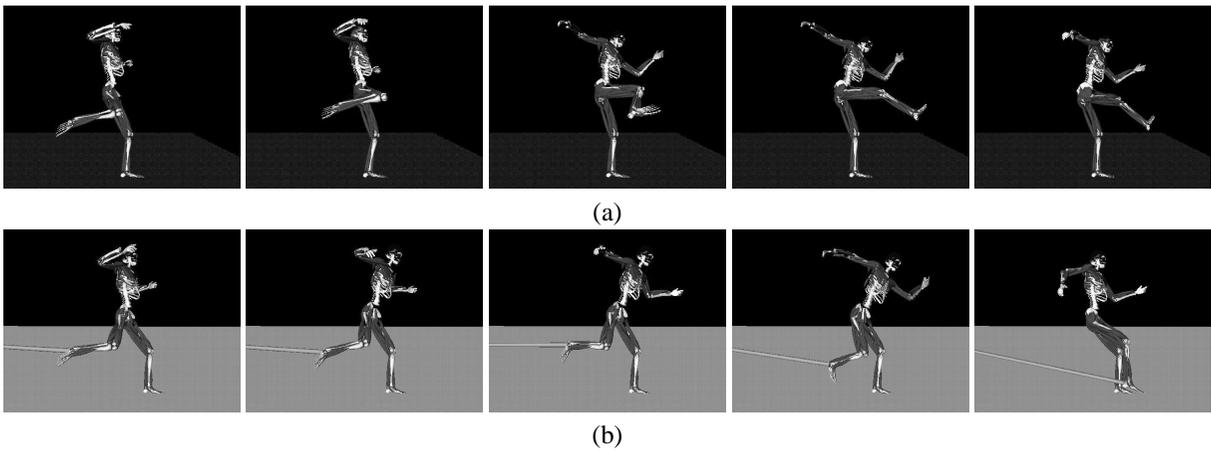


Figure 6: Conversion of the side kick: (a)initial motion and (b)converted motion.

6.1 Kicking Motion

The first motion captured from the real human body is shown in Figure 5(a). Using inverse dynamics and the musculoskeletal model, the force exerted by each muscle was calculated throughout the motion. The pH level of each muscle was then calculated, and the peak muscle force by the contractile element was updated using the fatigue and recovery model. As the motion is repeated, what was previously feasible becomes infeasible, because of the decrease in the peak force by the contractile element of the muscle. Thus, the optimization problem shown in equation (9) is solved, to provide a feasible motion. This operation was repeated 50 times and the feasible motions after 20 and 50 kicks are shown in Figures 5(b) and (c). It is possible to observe the effect of fatigue in the lower foot position of the kicking leg at the final posture. As the kick is repeated, the sway of the upper part of the body increases, and therefore the motion becomes more unstable. Another example of a kicking motion is shown in Figure 6. This time, the original motion is a side kick as shown in Figure 6(a). Next, an elastic spring was attached to the ankle of the kicking leg, and the conversion algorithm was applied to the motion. As shown in Figure 6(b), the leg is pulled down by the spring. Therefore, the right leg passes close to the ground during the kicking motion.

6.2 Retargeting the motion to different bodies

Next, kicking motions are retargeted to different bodies, by changing the size of bones and volume of muscles. In this subsection, the kicking motion was retargeted to motion by various bodies.

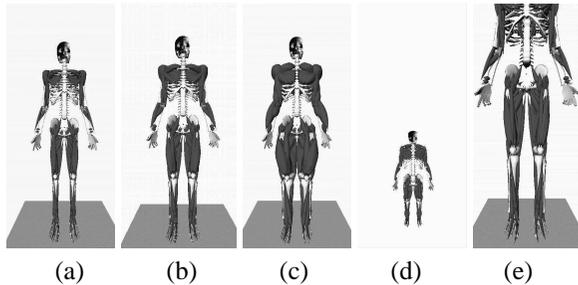


Figure 7: The musculoskeletal system of the (a) skinny body, (b) macho body, (c) more macho body (d) child body, and (e) giant body.

Kick by a weak, narrow body The muscles of the legs were all shrunk by 50 percent and a body as shown in figure 7(a) was composed. The average cross sectional area of all muscles were set to 50 percent of their original values, while the size of the bones and the muscle

attachment sites were unchanged. The converted kicking motion is shown in figure 8(a). Because of the weak muscle force, the motion is less powerful comparing to the original motion.

Kick by a macho body Next, the volume of the muscles of the legs were 1.4 times increased, and the body shown in figure 7(b) was composed. The size of the bones and the muscle attachment sites were unchanged again. The motion is shown in figure 8(b). This time, the motion is very powerful comparing to the previous motion. The velocity of the leg is faster and as a result, the leg reaches a higher position.

Kick by a more macho body Now, the volume of the muscles of the legs were further increased and set as 1.7 times the original value (Figure 7(c)). Even though the muscles can exert a greater amount of force, as the total mass and inertia of the legs have increased, the leg does not move as powerful as in the previous example (Figure 8(c)).

Kick by a 3year-old body The size and proportion of the body was changed to compose a body of a 3 year old child (figure 7(d)). The motion is shown in (Figure 8(d)).

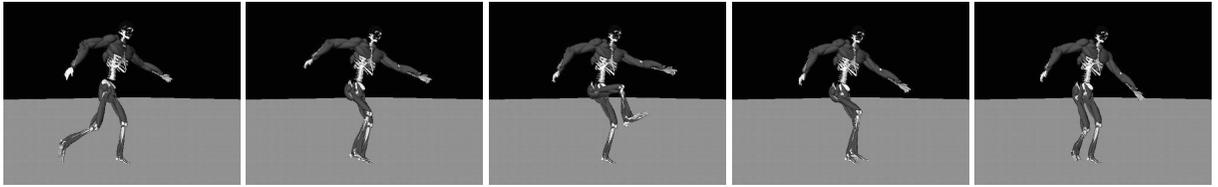
Kick by a giant body The size of the body is increased 1.5 times the original body, while the proportion is left the same (Figure 7(e)). The motion is shown in Figure 8(e).

6.3 Gait motion

A walking motion with a limp was created from an ordinary gait by reducing the maximum exertable force of the right muscles to $\frac{1}{5}$ of the original value and then optimizing equation (9). The initial gait motion and that with a limp are shown in Figures 9 (a) and (b). Next, a gait motion walking up a slope was created by changing the direction of the gravity. Starting the optimization from the motion in Figure 9 (a), a physiologically feasible slope-climbing gait was created as shown in Figure 9(c). The slope is 30 degrees. The footprints are set as constraints in these examples.

7 Discussion

Even though the use of a musculoskeletal system may seem complex and time-consuming, our method simplifies the process by avoiding the use of muscles as control parameters for optimization. In fact, the only extra computation necessary with our method, when compared with the traditional spacetime constraints problems, is the calculation of A , f^{max} and f^{min} in equation (1), and of τ_{ext} in equation (9). Even though the calculations for A , f^{max} and f^{min} increase in proportion to the number of muscles, since each computation can be completed very quickly, the total time for their computation is not critical. In addition, since the solution for minimum τ_{ext} is a quadratic programming problem, it can be solved very



(a)



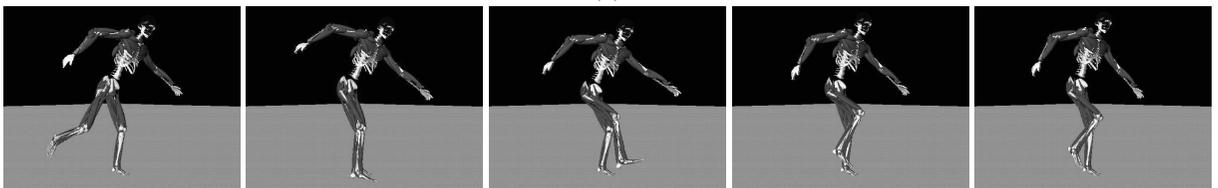
(b)



(c)

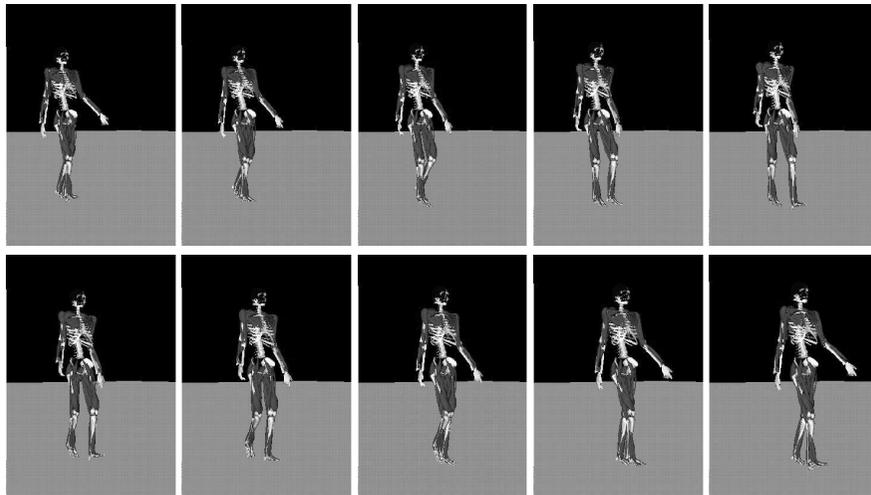


(d)

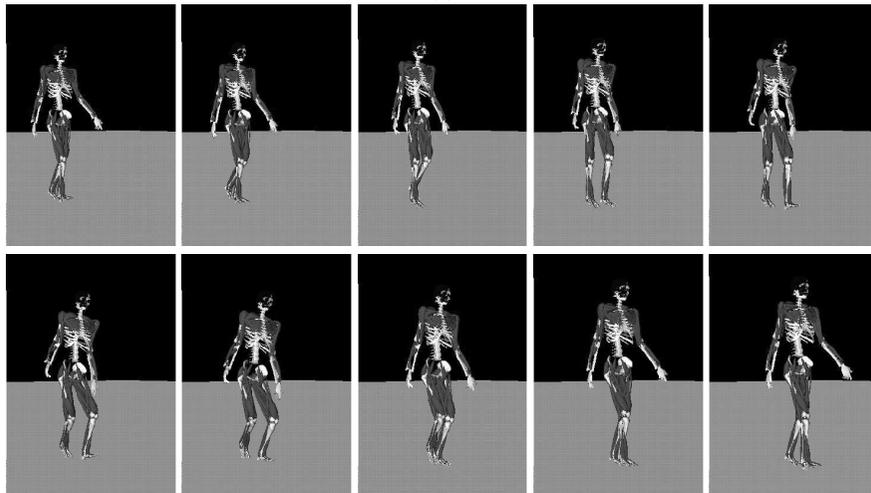


(e)

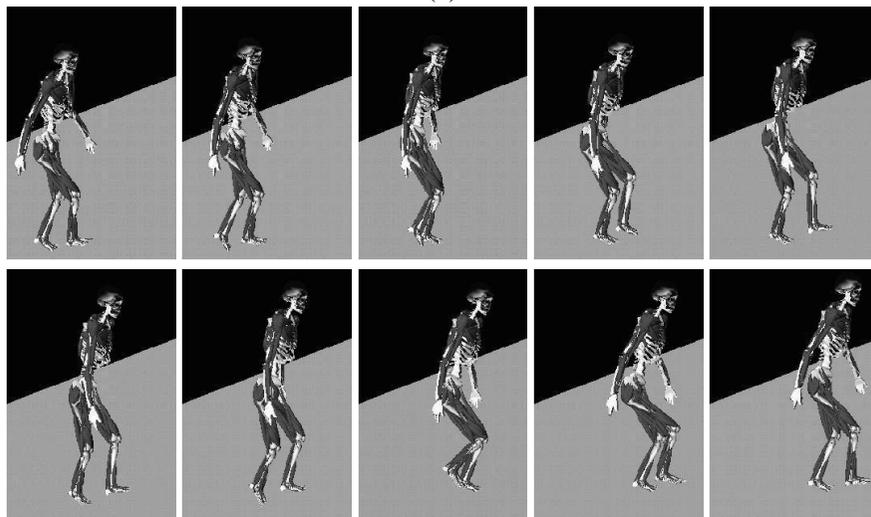
Figure 8: The kicking motion by various bodies



(a)



(b)



(c)

Figure 9: Conversion of gait motion: (a) the initial motion, (b) limp gait created by weakening the muscles of the right leg, and (c) a gait over a 30 degrees steep slope.

efficiently in a short time. As such, we consider the musculoskeletal system as more than just a special tool for specialized applications, it is also a tool that can be utilized in computer graphics for motion conversion.

We believe that rather than computations required for muscle parameters, the more critical problem is the number of DOFs of the human body. In order to further develop the human body model precisely, more DOFs and attached muscles will be required. In order to handle this increase, it will be necessary to seek for an optimization algorithm that can handle many DOFs. One potential solution may be multiresolutional methods such as those proposed by Liu [15].

8 Limitations

In this study, the torso has been greatly simplified. In particular, there are no muscles that connect the loins and the torso. In addition, the torso has only three degrees of freedom: the joints at the spine are all represented by one gimbal joint between the loins and the chest. In some cases, the chest joint oscillates in order to maintain the balance, and the motion does not look realistic. For example, in the kicking motion, as the the musculoskeletal model gets tired after repeating the kick, the chest part oscillates back and forth at a high frequency to maintain the balance. It will be necessary to model the spine more precisely to avoid this sort of motion.

9 Conclusions and Future Work

We have presented in this paper a method for physiologically converting real human motion data using a musculoskeletal model. We have shown how by combining the musculoskeletal model with spacetime constraints, it is possible to create dynamically and physiologically appropriate and feasible motions. Our method makes it possible to simulate physiological effects such as fatigue and injury. This kind of simulation has been difficult with contemporary systems that have not taken into account the internal structure of the body. We have also shown the use of this system for motion editing by changing physical properties such as gravity.

In respect of future work, we have begun constructing a musculoskeletal model of the torso which will enable the generation and retargeting of a greater number of motions and provide greater realism. As well as applications in pure computer graphics, this method may also be useful for simulating rehabilitation. We also believe that retargeting the musculoskeletal model to various characters by adapting muscle parameters and physical values to different body sizes would be an interesting topic.

References

- [1] <http://isb.ri.ccf.org/isb/data/delp>.
- [2] D T Chen and D Zeltzer. Pump it up: Computer animation of a biomechanically based model of muscle using the finite element method. *Computer Graphics (Proceedings of SIGGRAPH 92)*, 26:89–98, 1992.
- [3] M F Cohen. Interactive spacetime control for animation. *Computer Graphics (Proceedings of SIGGRAPH 92)*, 26:293–302, 1992.
- [4] S Delp, P Loan, M Hoy, F E Zajac, S Fisher, and J Rosen. An interactive graphics-based model of the lower extremity to study orthopaedic surgical procedures. *IEEE Transactions on Biomedical Engineering*, 37(8), August 1990.
- [5] Y Giat, J Mizrahi, and M Levy. A musculo-tendon model of the fatigue profiles of paralyzed quadriceps muscle under fes. *IEEE Transactions on Biomedical Engineering*, 40(7):664–674, 1993.
- [6] M Gleicher. Retargetting motion to new characters. *Computer Graphics Proceedings (Proceedings of SIGGRAPH98)*, pages 33–42, 1998.
- [7] F.C.T. Van Der Helm, H.E.J. Veeger, G.M.Pronk, L.H.V. Van Der Woude, and R.H.Rozendal. Geomoetry parameters for musculoskeletal modelling of the shoulder system. *Journal of Biomechanics*, 25(2):129–144, 1992.
- [8] J K Hodgins, W L Wooten, D C Brogan, and J F O'Brien. Animation of human athletics. *Computer Graphics (Proceedings of SIGGRAPH 95)*, pages 71–78, 1995.
- [9] R.K. Jensen. Changes in segment inertia properties between 4 and 20 years. *Journal of Biomechanics*, 22(6-7):529–536, 1989.
- [10] T Komura, Y Shinagawa, and T L Kunii. Muscle-based feed-forward controller of the human body. *Computer Graphics Forum*, 16(3):C165–C176, 1997.
- [11] T Komura, Y Shinagawa, and T L Kunii. The calculation and visualization of dynamic ability of the humna body. *The Journal of Visualization and Computer Animation*, 10(2):57–78, 1999.
- [12] Taku Komura, Yoshihisa Shinagawa, and Toshiyasu L. Kunii. Creating and retargetting motion by the musculoskeletal human body model. *The Visual Computer*, 16(5):254–270, 2000.
- [13] J Laszlo, M van de Panne, and E Fiume. Limit cycle control and its application to the animation of

balancing and walking. *Computer Graphics (Proceedings of SIGGRAPH 96)*, 30:155–162, 1996.

- [14] Richard L. Lieber. *Skeletal Muscle Structure and Function*, chapter 2, pages 95–100. Williams and Wilkins, 1992.
- [15] Z Liu, S J Gortler, and M F Cohen. Hierarchical spacetime control”. *Computer Graphics (Proceedings of SIGGRAPH 94)*, 28(2):35–42, 1994.
- [16] M G Pandy, F E Zajac, E Sim, and W S Levine. An optimal control model for maximum-height human jumping. *Journal of Biomechanics*, 23(12):1185–1198, 1990.
- [17] Z Popovic and A Witkin. Physically based motion transformation. *Computer Graphics Proceedings (Proceedings of SIGGRAPH 99)*, pages 11–20, 1999.
- [18] Veeger HE. Yu B. An KN. Rozendal RH. Parameters for modeling the upper extremity. *Journal of Biomechanics*, 30(6):647–652, 1997.
- [19] M van de Panne. Parameterized gait synthesis. *IEEE Computer Graphics and Application*, March:40–49, 1996.
- [20] M Vukobratović, B Borovac, D Surla, and D Stokić. *Biped Locomotion*. Springer-Verlag, 1990.
- [21] J M Winters. Hill-based muscle models: A systems engineering perspective. In Jack M. Winters and Savio L-Y. Woo, editors, *Multiple muscle systems: Biomechanics and movement organization*, chapter 5, pages 69–93. Springer-Verlag, 1990.
- [22] A Witkin and M Kass. Spacetime constraints. *Computer Graphics (Proceedings of SIGGRAPH 88)*, 22:159–168, 1988.

Appendix: Muscle Fatigue and Recovery Model

In this section, the fatigue and recovery model proposed by Giat et al. [5] is explained. This model can be used to determine the maximal amount of force exertable by a contractile element as time passes.

The decay of the pH level during the fatigue phase with time t is calculated by

$$pH^F(t) = c_1 - c_2 \tanh[c_3(t - c_4)] \quad (10)$$

with constant parameters c_1 , c_2 , c_3 and c_4 .

The pH level during the recovery phase is similarly calculated by

$$pH^R(t) = d_1 + d_2 \tanh[d_3(t - d_4)] \quad (11)$$

with constant parameters d_1 , d_2 , d_3 and d_4 . The force output is represented by the following function:

$$f_{pH}(pH) = d_5 [1 - e^{d_6(pH - d_7)}]. \quad (12)$$

where d_5 , d_6 and d_7 are constant values. Equation (12) is normalized by the force obtained at the beginning of the experiment:

$$f_{pH}^N(pH) = \frac{f_{pH}(pH(t))}{f_{pH}(pH(t_0))} \quad (13)$$

where $0 < f_{pH}^N < 1$. The values of the constant parameters defined here are listed in Table 1.

Function	Parameter	Value
pH^F	c_1	6.70
	c_2	0.502
	c_3	0.0406
	c_4	30.0
pH^R	d_1	6.55
	d_2	0.502
	d_3	0.0026
	d_4	475.806
f_{pH}	d_5	1136.
	d_6	-0.0097
	d_7	6.0934

Table 1: Parameter values for Giat’s fatigue and recovery model

The normalized force-pH function $f_{pH}^N(pH)$ is combined with the Hill’s muscle model to compose a new dynamic equation of the CE component [12]. The decay and recovery of the normalized force during the fatigue and recovery phase is shown in figure 10. The problem

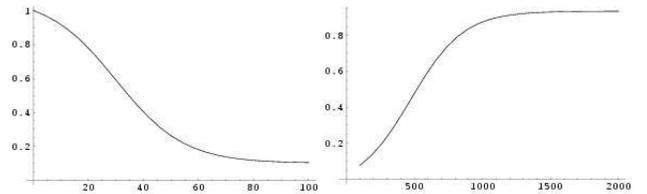


Figure 10: The relationship between time (seconds) and the normalized force by a muscle on fatigue (left) and recovery phase (right)

with this model is that the relationship between the activation level and the pH derivation is not included. Previous experiments have shown that the FG fibers begin to be recruited when the activation level is above 50% of maximal contraction [14]. Therefore, in this research, the phase of each muscle is switched between fatigue and recovery one according to its activation level. The threshold is set to 0.5.