Component Based Human Animation Architecture

Huang Jing, Chin Kah Fi and Edmond C. Prakash

School of Computer Engineering (SCE)
Nanyang Technological University, Singapore - 639 798
{p146587934 | ASprakash}@ntu.edu.sg

Abstract

This paper describes a method to predict the free reach movement of different components of the human body. Since the human body is composed of the following components: arms, legs, torso, head & neck, and each of these components has similar features, the same geometric inverse kinematics algorithm can be used in different parts of the body to realize free reach movement task. The main advantage of the proposed method is that it avoids matrix inverse calculation, which is very time consuming. And by using this method, the task can be mostly simplified.

1. Introduction

The study of human reach postures and animation has been an important area of research[2, 5, 4, 6, 9] in computer graphics as well as other industries. For example, computer animated human figures are widely used in film industry to achieve realistic special effects. Another important application is computer-aided design(CAD) and concurrent engineering. However, a realistic simulation of human movement and postures is not an easy task because of the redundancy of the human body at different control levels. Most of the research in this area has focused on inverse kinematics modeling, finite element analysis and assembly sequencing. What we use is a geometric method to simulate human free reach movement, which is the basic task for such computer animation. The objective of this paper is to use a single geometric inverse kinematics algorithm to predict the reach postures of different body parts. The structures and links of these body components are so similar to each other that we do not have to develop different algorithms for each of them. What we should only do is to reuse the same model to every body component. So by using this idea, the efforts to realize the task of free reach movement can be greatly minimized.

1.1. Problem Definition

The problem of posture prediction can be stated as follows: given an initial end-effector position and the posture of body component, how to calculate the final component posture. Since every body component has been simplified, the problem is equivalent to how to determine the position of the middle joint, for example, elbow, knee, etc., of each component. And in this case, the motion path of the end-effector is supposedly known.

1.2. Contributions

1. We provide an architecture for easy specification of human animation for global and local control.
2. We show that a component based Inverse Kinematics(IK) is suitable for all links in a human body.
3. We show that all computations can be done using a single IK kernel.
4. The IK kernel can also be replaced by a different IK kernel and adapted for hierarchical refinement of natural posture computation.

2. Previous Work

Animation of articulated figures have been studied in detail by several researchers:

In Auslander[1], a controller based animation helps the animators to influence the motion generated for articulated figures. In the other constraint based model used by Zhao and Badler[8] an articulated figure for an animator wishing to set a figure to a posture satisfying a set of positioning constraints, to achieve both generality or performance by a nonlinear programming technique has been proposed. The spatial constraint is applied on the end-effector, and one on the goal which is defined in the spatial environment.

The IK method proposed by Wang[7] has been chosen for our work. This is a geometric method to predict arm reach movement posture. It is proposed because it is largely based on the criterion of minimization of the norm of joint angular velocities. In this algorithm, the arm model is fashioned as a four degrees-of-freedom kinematics linkage system, with three at the shoulder and one at the elbow. One of the important advantages of this method is its ability to solve the non-linearity of the shoulder joint limit in a direct and easy way. No matrix inverse calculation is required in this algorithm, thus improving the performance by shortening the calculation and processing time. Some applications have also been made by using this method to realize some different arm movement tasks. And experimental results have proven that the arm reach postures predicted by this algorithm are very close to the real ones in a large arm reachable space. All these results point to the fact that the proposed geometric algorithm can be used as a very efficient arm posture manipulation primitive. A configuration space approach for efficient animation of human figures has been reported by Bandi and Thalman [3]. This approach allows the computation of the configuration space required for animation.

However, there is no system today that helps you to easily define a human model interactively, intuitively specify constraints and to achieve a task without an expert animator around. The goal hence is to:

1. Specify a human model for animation with simple two link elements in 3D space.
2. Perform all the animation computation using a two link kinematic links.
3. Specify task and behavior at the two-link level.

If this is achieved, we believe modeling and animation of humans will be as easy as typing an email. And the applications are enormous for example, every email that is sent will have a 3D customized human avatar that will give an animated sequence to convey what we intend to say over email at the other end.

3. Component Architecture

Components are body parts, which have similar links and structures so that the same method can be used to realize different tasks of different parts. The whole body can be divided into several body components, such as left arm, right arm, left leg, right leg, torso and neck & head. If we are required to simulate some more complex motions, the palm and even the thumb can also be thought of as components. All the components have a root of each and two or more links depending on the task. The component architecture we have developed is shown in Fig. 1.

At a higher level, the human architecture has three major parts: *ModelCreate* subsystem, *ModelAnimate* subsystem and *ModelBehavior* subsystem. The *ModelCreate* helps to specify the model, which in turn is composed of

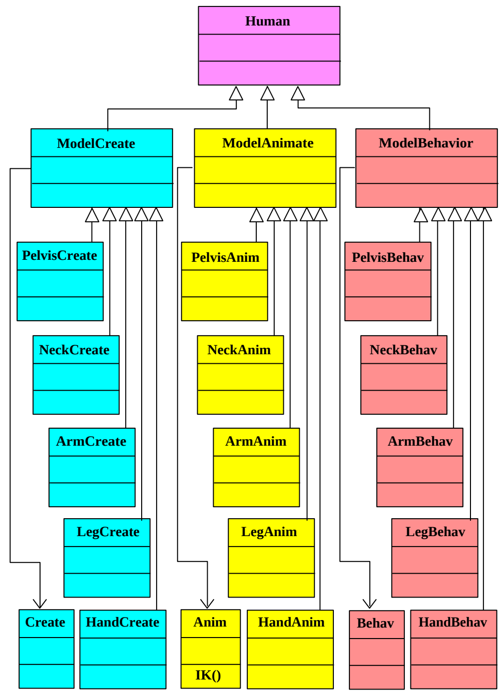

Figure 1: *Component Human Architecture*

smaller components. The nodes, coordinate systems and appearance attributes form the core of this subsystem. The *ModelAnimate* subsystem on the other hand helps to specify the trajectory for animation and also estimates the natural posture for animation of the model and all associated components. The *ModelBehavior* subsystem helps to specify the behavioral aspects of human animation. Separate behavior for each of the component of this subsystem can also be specified. The behavior of human neck is much different from the behavior of the human leg. These can internally be represented as a set of constraints. For example, in the movement of the human eye during gaze, the eye and head can independently move, and no head movement is noticed in the first ten degrees of eye rotation. Such factors are captured to our system using our *ModelBehavior* subsystem.

The advantage of such an architecture is that the subsystems are independent of each other. The components within the subsystem are also independent, so we can easily mix and match components.

The links of the human model are shown in Fig. 2. Some of the components, the number of links in each component, the constraints and the root node of the component are given in Table. 1.

4. Implementation and Results

The IK algorithm has been successfully implemented. More details of the IK algorithm are given in Wang's paper[7]. Some

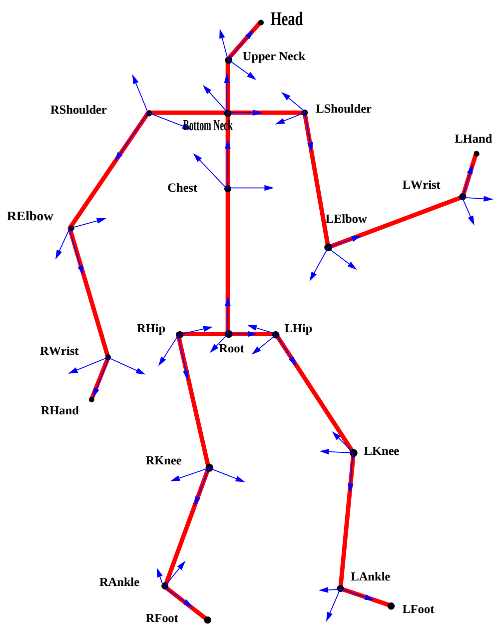

Figure 2: *Human Links and Nodes*

Table 1: *Constraints for Each Component*

Component	Number of links	Constraint DOF	Root Node
right arm	2	4	right shoulder
left arm	2	4	left shoulder
right leg	2	4	right hip
left leg	2	4	left hip
torso	2	6	pelvis
neck & head	2	6	bottom of neck

refinements have been made especially in the initialization of the algorithm. The IK algorithm currently takes very little time and can generate more than 30 new frames a second with 2 component IKs. We believe this will help interactive full human animation where each frame with all the component IKs can be computed in less than a second.

Experimental results have also shown that the final posture of the body component is fully determined by the initial posture and the motion path of the end-effector. Thus if the initial posture of the body component is not too awkward, a smooth and natural-looking motion can be generated. In this model, we use straight lines to describe the motion paths. Though this is not true in reality, by using this method, it is easy to control the motion and by comparing with the ideal path that the end-effector should follow, the calculation error is within a very small range in most cases. However, some problems will occur when the body component has to move a long distance from its initial position. The final position will show deviation from the ideal target position. This problem is mainly due to the straight motion path we have selected. In order to minimize the error, the full motion path has to be broken into several connected straight line sections, which form the trajectories close to a curve. Experimental results have shown that this is a quite effective way of reducing errors.

4.1. Input of the Animation System

Take the arm as an example, the input to the system are

- the initial position of the shoulder, elbow and wrist and
- final position of only the wrist.

The output for the rest of the arm movement between the initial and final position is computed automatically.

4.2. IK For Arm Posture

This experiment demonstrates the component IK for the arm. The animation sequence shows the movement of the human figure reaching for the telephone receiver and bringing it back to himself. Fig. 3 shows eight frames of the phone-lifting motion animation sequence.

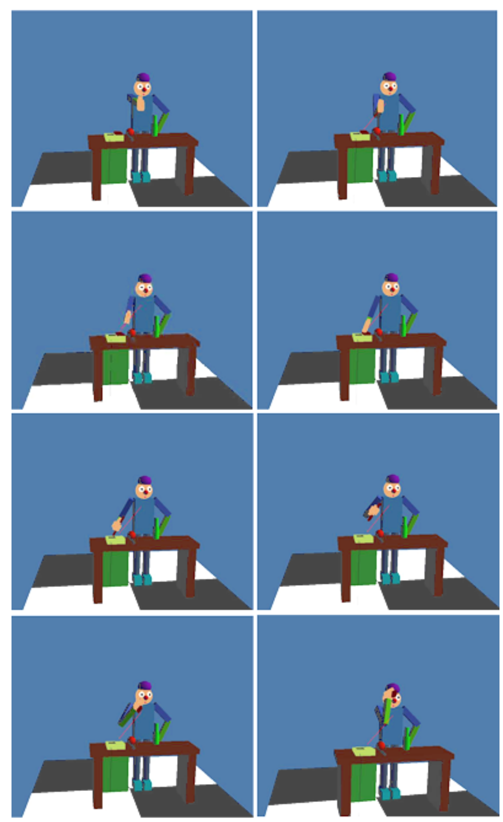

Figure 3: *Arm Animation using IK*

4.3. IK For Neck Posture

Fig. 4 shows an animated sequence of vertical gaze trajectory of normal smooth pursuit. The top row shows the snapshots when the gaze is pointing upward. And the bottom row indicates when the gaze is directed towards the floor. The trajectory for the gaze direction is on a vertical straight line. As the gaze is directed along the trajectory, for each new position, the new mid-eye position has been computed. This in turn is used by the IK algorithm to compute the neck posture and head posture. The hand position is moved in the opposite direction to that of the head.

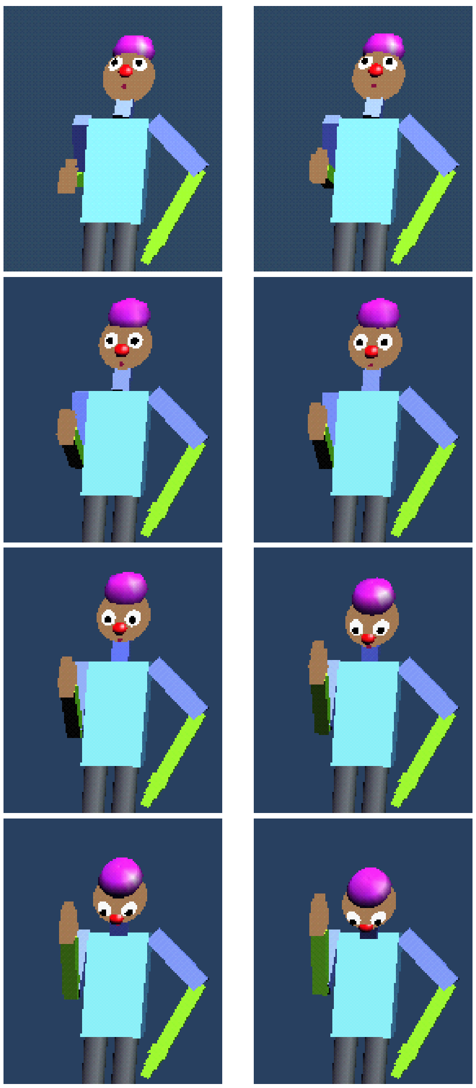

Figure 4: *Gaze on Vertical Trajectory*

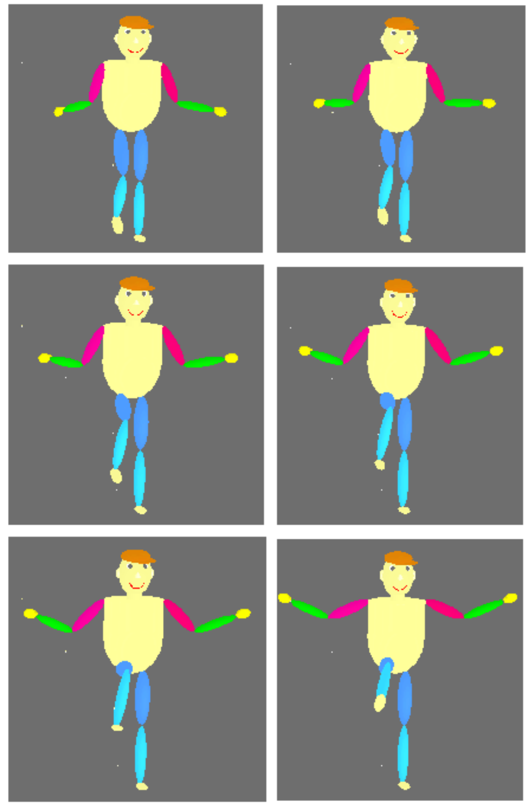

Figure 5: *Leg Animation using IK*

4.4. IK For Arm and Leg Posture

This animation sequence shows the motion that the arms open widely with the right leg lifted at the same time. The four white dots represent the start and end positions of the hand and leg respectively. The same method has been used in both the arm and leg without affecting each other.

5. Discussion

The advantage of this system is that only the start and end points of different body parts have to be specified. The system will automatically generate a smooth and natural motion of the whole human body. Thus it is very easy and simple to control. And since all the components of the human body use the same method, which can be realized by a specific program function, if any changes or replacements have to be made to the method, the modification will be quite simple. And also, from the results we can see that this system can successfully realize the task of free reach movement of different portion of the human body and the calculation is mostly simplified, which saves much time and space.

Any other IK scheme can replace our IK kernel. The model can be refined further by more careful evaluation of trajectory specification. A hierarchical scheme for refinement of IK will yield better results. For example, the neck has 8 joints, that we have approximated with 3 nodes. So a hierarchical solution will have a coarse IK at the top level, with a multinode IK at a detailed level.

6. Future Work

In this algorithm, all the motion paths have been thought of as straight lines, which is not true actually. So one desired hand trajectory has to be selected in the future to perform more natural movements. This model is simplified greatly to only four degrees-of-freedom so that only arm, leg and neck reach movement task can be easily realized. What we should do is to add more degrees-of-freedom and also more constraints to body components to make their motion more flexible. The input portion of this system has to be improved to allow more complex tasks to be easily specified.

We also note that the torso is not completely rigid; actually, the backbone is made of many vertebrae, resulting in a complex movement control. Currently our research is in progress to study the behavior of our system for the torso to see if it is still possible to define only the two limit points. When animating a walk, arms and legs have interrelated movements. And research is in progress to incorporate this to our system which would be a good benefit for animators. An improved method has to be used when the motions of the body components affect each other. That means, if the root of each component moves then we should consider more than just adding every component motion together.

7. Conclusions

This paper proposes the use of a geometric inverse kinematics algorithm for human motion based on a component architecture model. This algorithm only takes the position of the starting and terminating points of the body components as an input, and computes human posture in the following way. First every posture of the individual component of the body is calculated by the model separately and secondly all the components are assembled together hierarchically. Then the task can be fully realized. This algorithm has been shown to be applicable to human free reach movement. It is quite reliable and has been successfully used to generate many complex motions of the whole body. The result is a system that reliably finds the natural human motions. However, this algorithm is not the only method to simulate natural human motions. And it is only a method of free reach movement. When some more difficult tasks have to be realized, for example, grasping objects, some other algorithms must be integrated. So we need other inverse kinematics algorithms to make this system more robust.

References

1. Joel Auslander, Alex Fukunaga, Hadi Partovi, Jon Christensen, Lloyd Hsu, Peter Reiss, Andrew Shuman, Joe Marks, and J. ThomasNgo. Further experience with controller-based automatic motion synthesis for articulated figures. *ACM Transactions on Graphics*, 14(4):311–336, 1995.
2. N. Badler, C. Phillips, and B. Webber. *Simulating Humans: Computer Graphics, Animation, and Control*. Oxford University Press, New York, 1993.
3. S. Bandi and D. Thalmann. A configuration space approach for efficient animation of human figures. *Nonrigid and Articulated Motion Workshop 1997*, pages 38 –45, 1997.
4. Jessica K. Hodgins. Animating human motion. *Scientific American*, 278(3):46–51, Mar 1998.
5. Jessica K. Hodgins, Wayne L. Wooten, David C. Brogan, and James F. O'Brien. Animating human athletics. *Computer Graphics, SIGGRAPH 95*, pages 71–78, Aug 1995.
6. M.Cavazza, R.Earnshaw, N.Magnenat Thalmann, and D.Thalmann. Motion control of virtual humans. *IEEE Computer Graphics and Applications*, 18(5):24–31, 1998.
7. Xuguang Wang and Jean Pierre Verriest. A geometric algorithm to predict the arm reach posture for computer-aided ergonomic evaluation. *The Journal of Visualization and Computer Animation*, 9:33–47, 1998.
8. Jianmin Zhao and Norman I.Badler. Inverse kinematics positioning using nonlinear programming for highly articulated figures. *ACM Transactions on Graphics*, 13(4):313–336, 1994.
9. Wu Zhongke and Edmond C. Prakash. *Chapter 14: Voxel Human Animation.* in Volume Graphics, ISBN 1-85233-192-5, Min Chen(Ed), Springer, NY, 1999.