Transferring and animating a non T-pose model to a T-pose model

Nasim Hajari, Irene Cheng and Anup Basu

Department of Computing Science, University of Alberta

Abstract

Non T-pose animation is a technique that attempts to generate natural transformations between any non T-pose skeletons to the neutral T-pose skeleton. It is not always easy to extract or embed a T-pose animation skeleton into a 3D human model in an arbitrary initial position. This is even more problematic for natural human models obtained by 3D scanning, especially models of babies and kids. In addition, transforming a non T-pose to a T-pose requires a large amount of calculations. Hence, many commercially available software do not provide efficient methods to standardize non T-pose skeletons. This paper focuses on developing a simplified transformation method, which enables skeletons in arbitrary poses to be standardized and used in other media conveniently.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Computer Animation—3D Skeleton, Automatic Retargetting, Motion Generation

1. Introduction

Improvements in 3D capture technologies have led to many 3D models being widely accessible to the general public. Now, people would like to bring these articulated 3D models to life easily and quickly, by animating them automatically. However, the quality of the animations created depends largely on the realistic motion, and generating such motion is quite expensive and challenging. To get the best results, people usually record natural human movement of a live character through a motion capture system such as Vicon motion capture [Vic]. The captured motion is transferred to new articulated characters (Motion Retargetting), or several captured motions are combined together to generate new movements (Motion Generation). To get a realistic animation the source and target models needs to have the same initial position, which is not always possible. This problem is more severe if the target model is that of a baby or a child obtained through 3D scanning, as they cannot always maintain the neutral T-pose. One of the first work in the area of motion retargetting was proposed by [Gle98]. In this study they used the motion of an articulated model for another model with the same topology but different bone lengths. Even with the same topology, two models cannot directly share the motion and some adaption or modification is required. Some features of the motion are quite important and should be maintained during the animation. They set some constraints manually and modified the motion during retargetting to make sure that the constraints are always valid. This method has no information about the motion and relies on the constraints. Thus, if the solver is not provided with a comprehensive set of constraints the resulting animation may be unrealistic. On the other hand, [SC02] proposed a motion aware retargetting approach. They analyzed and classified the motion to determine the motion structure and identify its constraints. This eliminates the step of manually defining the constraints. The editing and retargetting of MoCap data can be useful to adapt the available motion to different characters and models. However, motion editing does not help if the required motion is different from the one already captured. In this case one needs to capture more motion sequences, which is a very expensive and time consuming step. This motivates the need for motion generation and motion synthesis. [KGP02] used motion graph, which is a structure that enables the captured data to be reassembled in different ways. This graph contains both pieces of original motion captured and automatically generated motions. New motions can then be generated by building walks on the graph. Another approach that is used to generate new motion is motion cut and paste as discussed in [AF02]. Motion database is quite important to build a rich set of generated motions and behaviour [LCR*02]. In order to generate new movements, one can use a collection of similar movements. [KP08] used Principal Component Analysis (PCA) to extract the set of basis elements from existing human motion data and then used Hidden Markov Models (HMM) to find the optimal linear combination of basis elements to describe a natural generated movement.

2. Proposed Method and Details

To animate a 3D human model one needs to have an animation or an IK skeleton, as shown in Fig. 1,which can be in different formats. Regardless of its format, the IK skeleton always contains information about the number of joints, the location of the joints relative to their parents, the length of the bones and the position and orientation of the bones. We propose a two step framework to automatically animate a non T-pose human model into a neutral pose.

DOI: 10.2312/egp.20161051

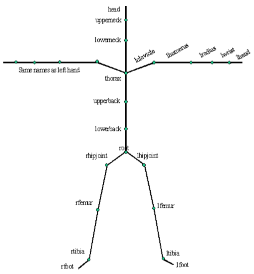

Figure 1: A human animation skeleton extracted from a MoCap file.

(a) (b) (c)

Figure 2: (a) The initial frame with sitting position; (b) Frame twenty shows the intermediate step (c) Final frame shows the neutral position

First, we calculate the final position of the skeleton joints in order to form a T-pose skeleton. Suppose the relative initial length and normal direction of joint j to its parent is given by l and $\mathbf{d} = [u_0\ v_0\ w_0]$ and the actual direction of the same joint relative to its parent for a normalized T-pose skeleton is given by $\mathbf{d}_t = [u_T\ v_T\ w_T]$. The final position of joint j relative to its parent is then given by Eq. 1.

$$\mathbf{p}_n = l \times \mathbf{d}_T \tag{1}$$

Note that the bone directions for a T-pose skeleton is fixed for human models and once obtained they can be used for different skeletons and models. Also, the length and direction of each joint are relative to its parent.

After we get the initial frame, which is the initial position of the model, and the final frame, which is the model transferred into a T-pose, we can interpolate to create some intermediate frames. For this research we used linear interpolation to get the joints positions of the intermediate frames. However, one can use other interpolation approaches to get more robust results. Suppose we would like to have an animation clip with n frames. The position of joint j in frame k is given by Eq. 2, where $\mathbf{p}_0$ is the position of joint j in the first frame.

$$\mathbf{p}_k = \frac{k \times (\mathbf{p}_0 + \mathbf{p}_T)}{n-1} \tag{2}$$

We should also maintain the physical property of human body movements, for example the maximum and minimum angles that each joint can have through rotation. For each frame we check these properties to make sure that they are within the natural range and the limits are the Degrees of Freedom (DOF) for each joint which can be obtained from a MoCap file. Suppose each joint j can rotate along the three axes with the values of $\mathbf{r}_j$. Then, to maintain the physical property of human movement we will have Eq. 3, where $\mathbf{min}_j$ and $\mathbf{max}_j$ correspond to the minimum and maximum of the DOF respectively.

$$\mathbf{min}_j \leq \mathbf{r}_j \leq \mathbf{max}_j \tag{3}$$

3. Experimental Results and Conclusion

The goal of this research is to transfer a non T-pose 3D model into a T-pose automatically and smoothly. To do so we proposed a two step framework where we initially compute the final rotation of the skeleton joints to form a T-pose skeleton and then generate some intermediate frames by linear interpolation between the joints initial position and their final rotations. Fig. 2 shows the initial, one intermediate and the final frame, generated by the proposed algorithm to transfer a sitting model into a neutral pose. We created the animation through a forty step interpolation, which resulted in a forty-frame clip. The video of the generated motion is provided as supplementary material. In future work we will investigate refining the automatically generated animation skeleton to achieve better perceptual quality in the animation.

References

[AF02] Arikan O., Forsyth D. A.: Interactive motion generation from examples. In *ACM Transactions on Graphics (TOG)* (2002), vol. 21, pp. 483–490. 1

[Gle98] Gleicher M.: Retargetting motion to new characters. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniq* (1998), pp. 33–42. 1

[KGP02] Kovar L., Gleicher M., Pighin F.: Motion graphs. In *ACM transactions on graphics (TOG)* (2002), vol. 21, pp. 473–482. 1

[KP08] Kwon J., Park F. C.: Natural movement generation using hidden markov models and principal components. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on 38*, 5 (2008), 1184–1194. 1

[LCR*02] Lee J., Chai J., Reitsma P. S., Hodgins J. K., Pollard N. S.: Interactive control of avatars animated with human motion data. In *ACM Transactions on Graphics (TOG)* (2002), vol. 21, pp. 491–500. 1

[SC02] Savenko A., Clapworthy G.: Using motion analysis techniques for motion retargeting. In *Information Visualisation, 2002. Proceedings. Sixth International Conference on* (2002), pp. 110–115. 1

[Vic] Vicon motion capture system. http://www.vicon.com. 1