Adding Hand Motion to the Motion Capture Based Character Animation

Ge Jin and James Hahn

Computer Science Department, George Washington University, Washington DC 20052
{jinge, hahn}@gwu.edu

Abstract. Most character motion capture data does not contain secondary motions like detailed hand motion, therefore the resultant animation looks unnatural due to the stiffness of hand motion. In this paper, we analyzed the pose space distance from the character's motion capture data and used stepwise searching algorithm to find the key poses for hand motion synthesis. We adaptively changed the contrast of the local distance map to capture the small motions. If an appropriate hand motion data available, temporal alignment with speed matching and spatial warping of motion data can produce realistic hand motion. If there is no motion capture data available, key frame with cubic or gaussian based interpolation can be used to generate in between frames.

1 Introduction

Motion capture based character animation has become major trend in the computer animation and computer game industry. The skeleton driven mesh deformation is an efficient approach for real-time applications like computer games and virtual reality. Although the motion capture technology can capture realistic human character motions, there has been less effort to simultaneously capture the secondary motions, like hand motion. There are some difficulties to capture gross body motion and intricate hand motion simultaneously. For the vision based motion capture device, the simultaneous capturing of hand motion is restricted due to the occlusion and capturing volume. Further more, most of currently available motion capture data does not contain the detailed hand motion. If we use motion data with no detailed hand motion, the skeleton driven character animation will look unnatural due to the stiffness of hand motion. Current researches in motion capture based character animation concentrate on utilizing existing motion capture data to synthesize new motion. Secondary motions, like hand motion is often neglected in motion synthesis research. Animation of human hand motion has been studied separately in regard to the full body animation. Early works in human hand animation focused on grasping and gesture motion using inverse kinematics and hand motion constraint. Recent works in hand animation concentrated on ASL (American Sign Language) animation, music driven hand animation and realistic hand deformation. Data glove and Cyber glove can be used to capture the hand motion in real time. However, by our knowledge, there has been no work reported for adding hand motion detail to the full body motion capture based character animation.

Local arm motion is closely related with the hand motion. This type of motion includes basketball dribble, catching and throwing motion but not limited to. We have

G. Bebis et al. (Eds.): ISVC 2005, LNCS 3804, pp. 17 – 24, 2005.

analyzed the pose space distance map and found out that the local extreme points in the pose space distance map are the important transition point for the hand motion synthesis. Some of the hand motions are not closely related with the arm motion, like counting with fingers. If we use the global pose space distance map, these small features will be neglected. In order to capture these small changes in the motion capture data, we give more weight to the hand tip difference and increase the contrast of the local region in the pose space distance map. Using adaptively changed pose space distance map, we can detect the local extreme point for small hand motions.

If the hand motion and character animation is closely matched, for example, throwing and catching hand motion to the same throwing and catching character motion, temporal alignment with speed matching is used to seamlessly combine hand and character motion. With similar but not exactly same hand motion, like throwing and catching hand motion to the basketball batting motion, we have used spatial warping of hand motion data to approximate the basketball batting motion. If there is no hand motion capture data available, we defined some key frames and used hermite curve based and gaussian based interpolation to generate in between hand poses.

2 Related Works

Motion synthesis from existing motion capture data has been studied for many years. Early works in motion editing includes motion warping, motion retargeting, motion interpolation, and motion signal processing. Motion warping [1] can make small changes to the end effect of existing motion clip. Motion interpolation [2] method uses pre-existing motion samples to generate new motion clip with linear interpolation. Motion retargeting with space-time constraint [3] [4] has been proposed to map a character's motion to different character while maintaining important constraint. Motion signal processing approach [5] [6] can be used to modify locomotion like exaggerating the motion or changing the step size. Noise based [7] method used procedurally defined motions to simulate nervousness. Recent approaches try to find the closet transition point among various motion clips to synthesize natural long sequence of new motion. Motion graph [8] is used to organize the motion clip into graph structure and automatically generate a transition between the graph nodes.

Introducing secondary motion (like hand motion) to the motion captured character animation has not been reported. But this work is closely related with motion transition and key pose selection work. A seamless transition point is detected using distance between point clouds driven by the skeleton [9]. Another approach uses the translation and weighted average of joint rotation vectors to detect the transition point. [4] An affinity matrix composed of joint position, joint angle, joint velocity and joint angular velocity is used to detect the global key poses for illustrating an animation with a few key poses [10].

In motion transition, the minimum distance between two poses is good candidate point for motion transition. In our case, we are interested in the maximum distance between two poses, which will be potential hand motion transition point. Our key pose detection method for hand motion synthesis is closely related with key pose illustration work [10]. However, for our purpose, only the local arm motion will be considered for the key pose selection. Another difference is the key pose illustration work [10] pays more attention to the global extreme poses, while for hand animation we only need to consider the locally continuous and related transition positions. By adaptively changing the weight of local area of pose space distance map, we can detect subtle motions that could have been neglected by global pose space distance map.

Early works in hand motion animation has discovered a commonly used joint angle dependency $\theta_{DIP}=2/3\theta_{PIP}$ [11]. More detailed hand constraint is reported in [12]. They defined the lower and higher bound of finger joint angles and discovered that the transition from one hand pose to another pose is a linear transformation. We have used the joint angle dependency $\theta_{DIP}=2/3\theta_{PIP}$ to capture the hand motion using electro magnetic trackers. Procedural algorithm and neural net approach have been introduced to synthesize the hand motion of playing musical instrument [13] [14]. Realistic hand skin deformation methods have been reported using bone and muscle structure [15] and pose space deformation [16].

3 Skeleton Structure and Vertex Weight Calculation

We manually choose the feature points on the 3D surface mesh to generate human skeleton structure for limbs, torso and hand. We select the major joint position on the front and backside of 3D mesh model. The skeleton structure can be generated using median position of front and backside feature points. The vertex to skeleton weight is automatically calculated using angular median of skeleton vectors. For every vertex, we choose the shortest and second shortest distance skeleton as weight calculation candidate. The dot product of the angular median vector and vertex to skeleton joint vector is used to calculate the weight for each skeleton. Each mesh vertex has two weights ($w1,w2$) for two nearest skeletons. The constraint is: $0\leq w1\leq 1$, $0\leq w2\leq 1$,and $w1+ w2=1$. Random color is assigned to each skeleton to show the controlling vertexes. For the vertexes that affected by two skeletons ($w1>0$, $w2>0$), the vertex color was alpha-blended to show the region controlled by two skeletons.

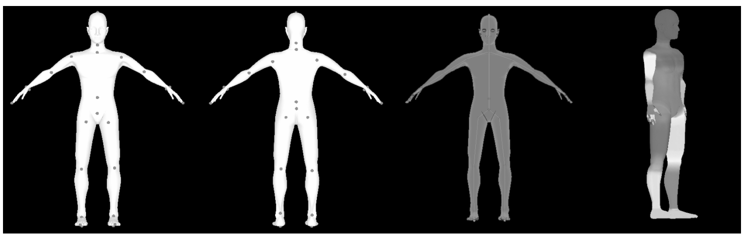

Fig. 1. The first two images show the feature point selection on front and backside of 3D model, the third image shows the skeleton structure from these surface points, the last image shows the result of vertex to skeleton weight calculation

4 Key Pose Selection from Character Motion Capture Data

We are interested in the key poses that are closely related to the hand motion. For this purpose, we only consider local arm motion for key pose selection. We choose the shoulder, elbow, wrist and hand tip position to calculate pose space distance map. Since the global translation and pelvis rotation will affect the joint location, we modified the motion data so that the global translation and pelvis rotation remain unchanged. We used the corresponding joint distance between two frames in motion sequence to generate a pose space distance map. P_t^i, P_t^j is the 3D position of certain joint in *i*th and *j*th animation frame. w_t is the weight for each different joint. A distance map pixel value $D_{i,j}$ is calculated using equation (1).

$$D_{i,j} = \sum_{t \in \text{joint}} w_t dist(P_t^j - P_t^i) : dist\ is \text{ euclidian distance} \tag{1}$$

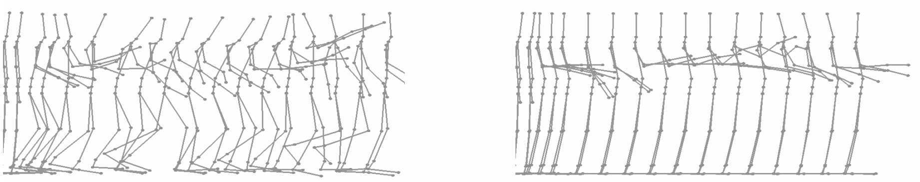

Fig. 2. The sequence of basketball dribble animation and the standing and counting animation

We have computed pose space distance map of right arm for dynamic motion such as basketball dribble and static arm motion like standing and counting (Figure 2). Initially, we give the same weight to the different joints (shoulder, elbow, wrist and tip).

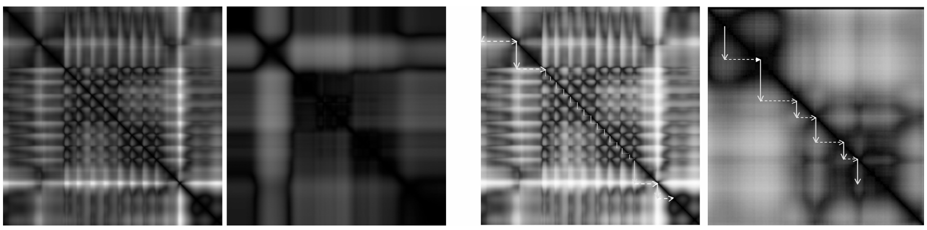

Fig. 3. Left image pair is pose space distance map of basketball and counting motion and right image pair is stepwise searching for key poses from them

For the highly dynamic motions, the resultant pose space distance map shows the clear cue of the maximum distance between two arbitrary frames (Figure 3). For the static arm motion, the distance map could not provide clear cue of counting motion with fingers, only the big movement is indicated (Figure 3).

In order to catch the small motions related with hand tip movement, we give more weight to the hand tip distance and increased the intensity contrast with certain areas: where the mean distance value is less than certain threshold. By doing so we can discover the small motion details like tiny hand tip movement (Figure 4). For the real static frames, increasing the contrast will not provide more cues of motion (Figure 4). Using adaptively weighted and contrast increased pose space distance map, we can detect the key poses from both high dynamic and static arm movement.

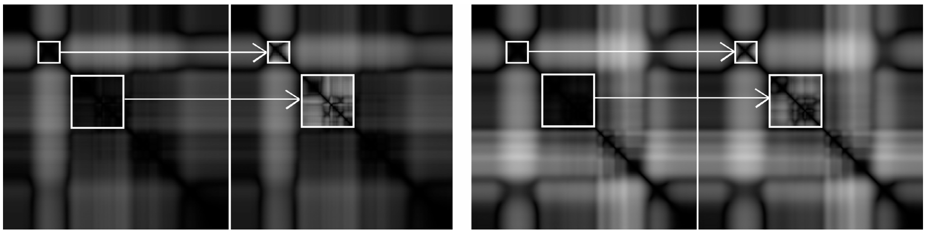

Fig. 4. The pose space distance map of counting motion and contrast increased maps

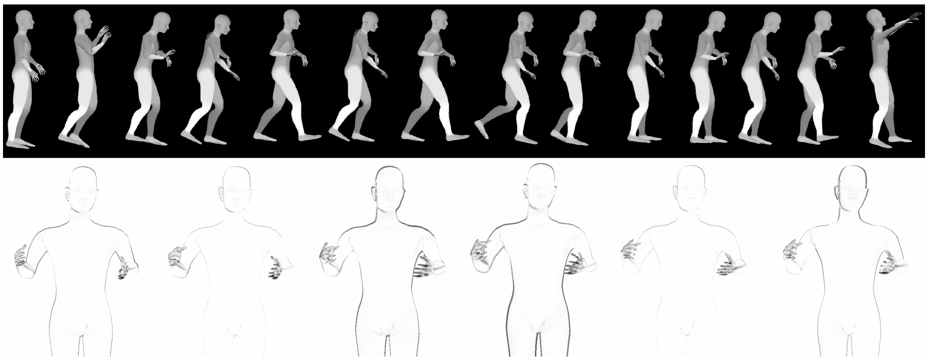

Fig. 5. Automatically calculated key poses from basketball batting motion (upper) and difference of two consecutive key poses in standing and counting motion (lower)

Searching for the key poses from pose space distance map works as follows: we started from the first frame and search for the first local maximum distance with first frame, we searched for local area around this extreme point to find if there is higher local maximum value in this area. The local maximum point $D_{ij}(i<j)$ indicates the frame i and j have maximum distance in pose space. We jump over to the next frame j, and repeat the search with same method. The figure 3 shows the stepwise searching algorithm from basketball batting motion and local contrast increased standing and counting motion. The result of key pose selection is shown in figure 5.

5 Hand Motion Synthesis

In order to add realistic hand motion to the full body motion capture data, we need the hand motion that matches with the character motion. We started from the simple case:

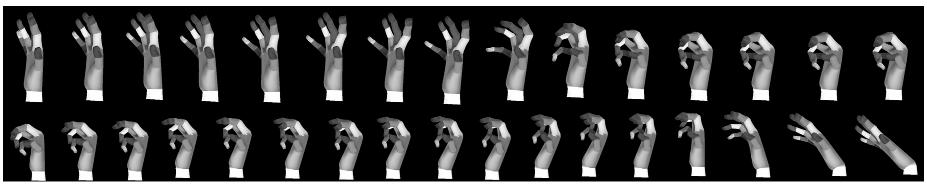

Fig. 6. Capture of catching (upper image) and throwing (lower image) motion

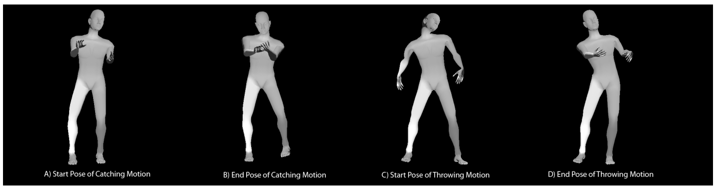

Fig. 7. Extracted motion cycle from the catching and throwing motion

adding catching and throwing hand motion to the catching and throwing motion of character animation. The captured hand motion is illustrated in figure 6.

The extracted key poses (Figure 7) from pose space distance map indicate the motion cycle such as starting and ending pose of catching motion. If we simply add the hand motion cycle to these key poses, the animation will look strange due to the speed mismatch. The motion capture data contains the frame rate information, and we can use this information to calculate the speed of these motion cycles. Another constraint is that: the end pose of hand motion and end pose of arm swing motion should be matched. By matching the end poses and the speed, we can align hand motion capture data to the character animation in temporal domain.

If there is no exact hand motion that matches the character animation, we can use the similar hand motion with spatial warping to synthesize new hand motion. We can use the catching and throwing hand motion to synthesize basketball-batting motion. The main difference in basketball-batting and throw-catch motion is the flexion degree of fingers. We warped the flexion degree of throwing and catching hand motion to approximate the basketball batting motion.

If there is no similar hand motion that can be used with the character animation, key frame interpolation between several predefined hand poses will generate smooth hand motion. For a finger bending motion, the angular velocity at start and end pose is smaller than the velocity at the middle of bending motion. We have experimented with linear, hermite curve and gaussian based interpolation. In hermite curve based interpolation, the interpolation parameter $t(0 \leq t \leq 1)$ is not a constant time interval. Suppose X is time axis and Y is the finger angle axis, the interpolation between two hand pose can be expressed as: $X(t) = f(t)$, $Y(t) = g(t)$ ($f(t)$, $g(t)$: cubic function). We solved general cubic function for $f(t) = x$(certain time), and used the resultant t to get the interpolation value $g(t)$. A gaussian function with appropriate standard deviation will generate good interpolation curve. Both the hermite curve based and gaussian based interpolation generate good result.

6 Results and Discussions

The figure 8 is the animation result of our approach. The resulting hand motion is smoothly integrated into the original human character animation. However, one limitation in our approach is: the predefined hand motion should be close enough to the real hand motion in human character motion capture data. Some of the character animation does not show clear cue of hand motion, or the rough hand motion is missed due to the occlusion in motion capture device. In this case, it is hard to apply our method. Inverse

Kinematics based approach is a good candidate when the end constraint of hand tip is already known. But in most case, motion capture data does not contain this kind of constraint. Inverse Kinematics or the Neural Network based optimization for hand motion synthesis usually takes long time to calculate. For our approach the calculation of pose space distance map is done during the preprocessing stage, and the merging of hand animation to the motion capture data can be done in real time.

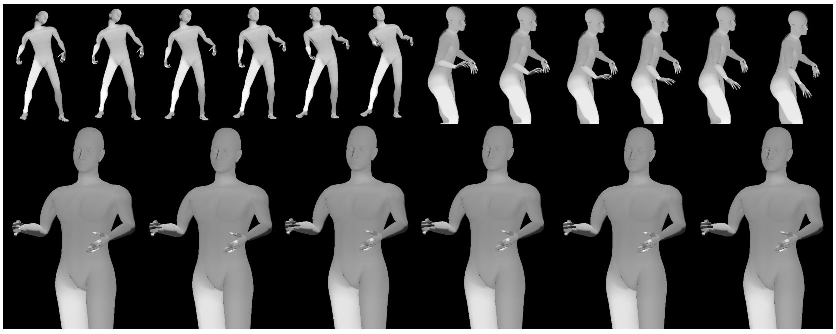

Fig. 8. Catching motion with hand animation, Basketball batting motion using spatial warping, Counting motion with key frames and gaussian based interpolation kernel

7 Conclusions and Acknowledgements

We have proposed a stepwise searching algorithm to capture key poses from local pose space distance map. The pose space distance map can be locally adjusted to capture subtle motions. Temporal alignment with speed matching and spatial warping was used to add hand motion capture data to the character motion capture data. If there is no hand motion data available, we can use key frame based interpolation with hermite curve or gaussian kernel to generate in between frames. The future research will be focused on more practical and general method: like dubbing a movie, while playing the character animation, simultaneously capture hand motion using data glove. The main challenge will be the seamless synchronization of two motions.

The character motion capture data, which we used in our experiment, are downloaded from http://www.animazoo.com/bvh/. The 3D human character and hand models are downloaded from INRIA Gamma Team Website Collections. (http://ftp-rocq.inria.fr/Eric.Saltel//download/download.php) The hand motion is captured using 3D Articulograph from Carstens Medizinelektronik. (http://www.articulograph.de/)

Acknowledgement. The authors wish to thank anonymous reviewers for their careful comments.

References

1. A. Witkin and Z. Popovic: Motion Warping, Proc. SIGGRAPH (1995) 105-118
2. D. Wiley, J. Hahn: Interpolation Synthesis of Articulated Figure Motion, IEEE CG&A, Vol. 17. No. 6. November/December (1997), 39-45

3. M. Gleicher: Retargetting Motion to New Characters, Proc. SIGGRAPH (1998) 33-42
4. J. Lee and S. Yong Shin: A hierarchical approach to interactive motion editing for human-like figures, Proc. SIGGRAPH (1999) 39-48
5. A. Bruderlin, L.Williams: Motion signal processing. Proc. SIGGRAPH (1995) 97-104.
6. M. Unuma, K. Anjyo, R. Takeuchi: Fourier Principles for Emotion-based Human Figure Animation, Proc.SIGGRAPH (1995) 91-96
7. K. Perlin, L. Velho: Live paint: painting with procedural multiscale textures. Proc. SIGGRAPH (1995) 153-160
8. L. Kovar, M. Gleicher, F. Pighin: Motion Graphs. Proc. SIGGRAPH (2002) 473-482
9. L. Kovar, M. Gleicher: Automated extraction and parameterization of motions in large data sets, Proc. SIGGRAPH (2004) 559-568
10. J. Assa, Y. Caspi, D. Cohen-Or: Action Synopsis: Pose Selection and Illustration, Proc. SIGGRAPH (2005)
11. H. Rijpkema , M. Girard, Computer animation of knowledge-based human grasping, Proc. SIGGRAPH (1991) 339-348
12. J. Lin, Y. Wu, T. Huang: Modeling the Constraints of Human Hand Motion, Proc. of the Workshop on Human Motion (2000) 121-126
13. G. ElKoura, K. Singh: Handrix: animating the human hand, Proc. of ACM SIGGRAPH/Eurographics SCA (2003) 110-119
14. J.Kim, F. Cordier, N. Magnenat-Thalmann: Neural Network-Based Violinist's Hand Animation, Proc. of International Conference on Computer Graphics (2000) 37-41
15. I. Albrecht, J. Haber, H. Seidel: Construction and animation of anatomically based human hand models, Proc. of ACM SIGGRAPH/Eurographics SCA (2003) 26-27
16. T. Kurihara, N. Miyata: Modeling deformable human hands from medical images, Proc. of ACM SIGGRAPH/Eurographics SCA (2004) 355 - 363