Motion Capture Data Retrieval Using an Artist's Doll

Tien-Chieng Feng, Prabath Gunawardane, James Davis
University of California, Santa Cruz
{txfeng, prabath, davis}@soe.ucsc.edu

Bolan Jiang
California State University, Sacramento
jiangb@ecs.csus.edu

Abstract

In this paper, we present a keyframe-based human motion capture data retrieval system which uses a wooden doll as the input device. A user inputs a keyframe by posing an artist's doll with painted joints in front of a stereo camera rig. The system interactively gives real-time feedback on the results of the joint detection and 3D pose reconstruction as the user is positioning and rotating the doll. The robust 3D joint reconstruction is achieved by integrating 3D joint positions from multiple views of the same pose. After the user has finished inputting all the keyframes, the motion sequences are retrieved from the database and ranked based on their similarities to the keyframes. Experiments show that the presented system is simple to use and has high quality of retrieval results.

1. Introduction

Information retrieval on existing motion capture (mocap) data is an important technique for areas such as the study of human motions and character animation in games and movies. For example, reusing existing mocap data for an animation sequence is much more time and cost efficient than capturing the whole motion from scratch. A key capacity for accomplishing high-quality motion retrieval is an interface which allows users to specify queries effectively and easily.

Some motion retrieval systems use text-based input interface, which is hard for users to specify complex queries. Some other systems use short motion clips or keyframes as query input. However, it requires a steep learning curve to create short motion or keyframes from using traditional WIMP interface. So we need a system which can allow users, especially novice users, to specify complex keyframe poses effectively and easily for query input.

In this paper, we present a keyframe-based human motion capture data retrieval system using a cheap wooden doll as the input device. A user inputs keyframe queries by repeatedly posing an artist's doll with painted joints in front of a pair of calibrated webcams. Using computer vision techniques the system detects the center of each joint on the doll and performs 3D reconstruction of the 3D pose. The system gives real-time feedback on the results of the joint detection and pose reconstruction as the user is posing the doll. To avoid occlusion of the joints, the user can rotate the doll to a position where occluded joints are visible and detected. The system then integrates the 3D joint position data from multiple views of the same posture. After user inputs all the keyframes, the motion sequences which best match the input keyframes, are retrieved using relational feature indexing and retrieval method. The results are ranked on their similarities to the input keyframe postures using Dynamic Time Warping (DTW) technique. Experiments show that the system is simple to use and has high quality of retrieved results. The main contribution of this paper is a low-cost vision-based interface which allows users to specify motion queries effectively and intuitively. We have also built a complete and reliable keyframe-based motion retrieval system for the interface.

2. Related work

Some motion capture libraries use text input for specifying queries. For example, CMU motion capture library [1] asks users to input present tense verb such as "jump" to search for motion sequences. The motion systems presented in [2, 3] use short 3D motion clips as input queries. However, it is hard and tedious to

create 3D motion using traditional WIMP-based software such as Maya or Poser.

There are some systems which use non-WIMP interfaces to create 3D motion. These systems are more intuitive and easier to use compared with WIMP-based systems. However, each system has its limitation.

Sketch-based systems [4, 5, 6] allow users to specify keyframes by 2D sketches. These keyframes are then interpolated to produce an animation sequence. However this type of interface may seem only appealing to skilled animators or artists who are trained to specify keyframes by drawing. In this paper we present a simple interface that is suitable for not just trained animators but also the general users who have little art training.

Some systems allow users to manipulate virtual characters by acting out. User's motion could be tracked by cameras [7, 8] or a foot pressure sensor [9]. These systems can be quite accurate in projecting users' motions to those of the avatars. However, it requires large amount of users' physical effort to specify the motion they want. And it is hard for users to perform some difficult and extreme motions. For example, not everyone can do jumps of figure skating. With our system users are able to specify postures of the 3D character by easily posing the doll to the desired postures.

The systems which utilize tangible input devices have been developed to allow intuitive ways of interacting with virtual objects. The Monkey, which is a 25" human-like input device, was created to allow easy keyframe specification through manipulating the parts on the Monkey's body [10]. Despite of its convenience, the Monkey comes with high price. Blumberg et al. [11, 12] presented a system which uses an instrumented plush toy to control the behaviors of an animated character. Barakonyi et al. developed an augmented reality system using an input device similar to ours [13, 14]. They attached magnetic trackers to the head and limbs of a wooden mannequin. While a user is posing the mannequin, the system maps the pose data to rotation information of the animated character using inverse kinematics and motion mapping techniques. All of these systems provide novel ways to interact with virtual characters. However, they require carefully instrumented and costly input devices, of which the setup process is complicated and time-consuming. We created a tangible user interface that is easy to use and affordable for users at all levels.

For the mocap data retrieval part of the system, we use a modified version of the method described in [2]. Unlike other systems, users are no longer required to input short motion segments as queries. Besides, unlike other keyframe-based mocap data retrieval system [15], we allow users to easily generate input keyframe queries through a tangible and simple interface.

3. Methodology

There are two parts to the implementation of this system; the front-end user keyframe input interface, and the back-end motion data retrieval module. Figure 1 provides an overview of the pipeline of this system.

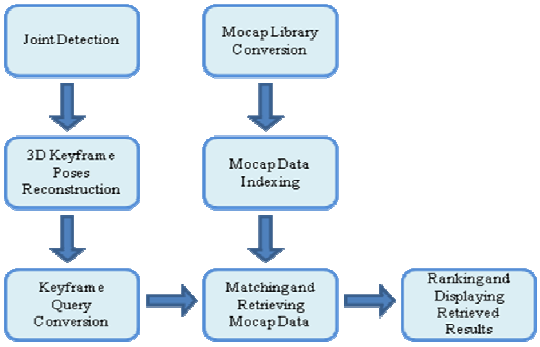

Figure 1. The system's pipeline.

3.1. Keyframe input interface: joint extraction

We use an ordinary artist's doll that can be easily obtained in any artist's shop to maintain the simplicity of the interface without compromising the usability. The body of the doll is painted black and the joints are painted in 7 different colors to facilitate the detection process. The colors on the doll's joints are chosen such that they are distinctly separate in color space so there would be minimum room for confusion even under changing lighting conditions. The user places the doll, standing upright, with the desired pose at a fixed location in front of a pair of video cameras; here we use Logitech Communicate STX webcams. A piece of black paper is placed behind the doll to avoid background noise (see figure 2). Then each frame of the left and right captured image is first smoothed with median filtering, and then transformed into binary images. By this time only the joints of the doll, and possibly some noise, appear as connected regions on the images. We traverse through each connected region and compare its color to the 7 joint colors. If the difference to any color falls within a certain threshold, the connected region is assigned to that color group. Then the centroids of the largest 2 connected regions of any pre-selected color become the 2 detected joint centers of the doll pose (see figure 3).

3.2. Keyframe input interface: 3D pose reconstruction

To find 3D positions of the joints, we triangulate the known 2D image coordinates of each joint. For each keyframe pose we cast rays from the center of the left camera through each joint of the left doll image. We do the same for the right camera and its images. Then we find the intersections of the 2 rays for each joint to determine its 3D position (see figure 4). To ensure the correctness of reconstructed 3D pose, we need to filter and correct the reconstructed 3D poses as necessary. Since pose image data are fed to the system at an interactive frame rate, the system processes each frame of captured images and rules out noisy 3D joint data by imposing both epipolar line and bone length constraints. However it is possible that sometimes a captured pose may have joints occluded in the images. In these cases the 3D keyframe poses cannot be fully reconstructed. To solve this problem, the system integrates 3D pose data of the same doll pose from multiple viewpoints to obtain the full 3D pose reconstruction (see figure 5). Finally the set of all generated keyframes form the input queries for retrieving mocap data.

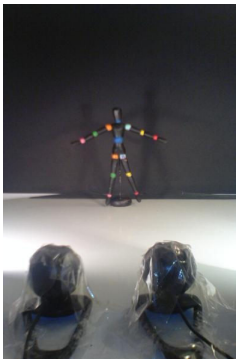

Figure 2. The system's physical setup.

Figure 3. To the left is a doll in the natural position. To the right are the centers of joints detected and colored in the joint's color.

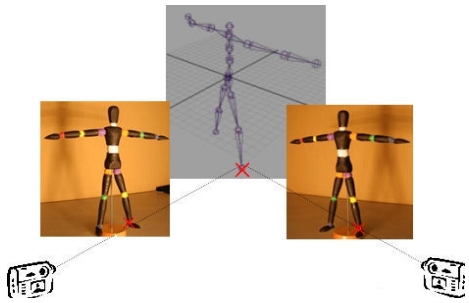

Figure 4. Rays are casted from camera centers through each joint center to find its 3D position.

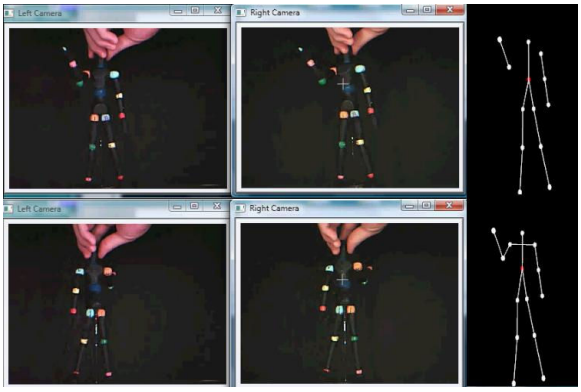

Figure 5. The top left 2 images are the captured left and right images of a pose with occluded right shoulder joint. The top right image is the partial 3D pose reconstruction. The bottom left 2 images show a different view showing the right shoulder joint. The bottom right image shows the integrated 3D pose with the missing joint detected.

3.3. Mocap data indexing and retrieval

The motion data library used in this system is a subset of the mocap database by CMU [1]. Our system adapts a modified version of the indexing technique based on relational feature [2]. The main difference between our indexing method and the one described in [2] is the set of features we use as index terms. We selected a specific set of features for indexing and retrieving pose data using keyframe poses. All motion data segments are indexed based on certain features, such as arm above neck, bending leg, etc. Then motion sequences that contain segments with the same feature vectors as each of the input keyframes are retrieved. Then for all retrieved sequences, DTW is performed between the input doll poses and the frames of the segments they matched to. The cost function for DTW is the weighted Euclidean distance between the joints

of the doll pose and library motion data. The sequence with lowest score is ranked highest.

4. Retrieval results

Retrieved motion sequences are first sorted in increasing order of the score obtained from DTW comparison. Then they are displayed in a top-down fashion with 4 results per page (see figure 6).

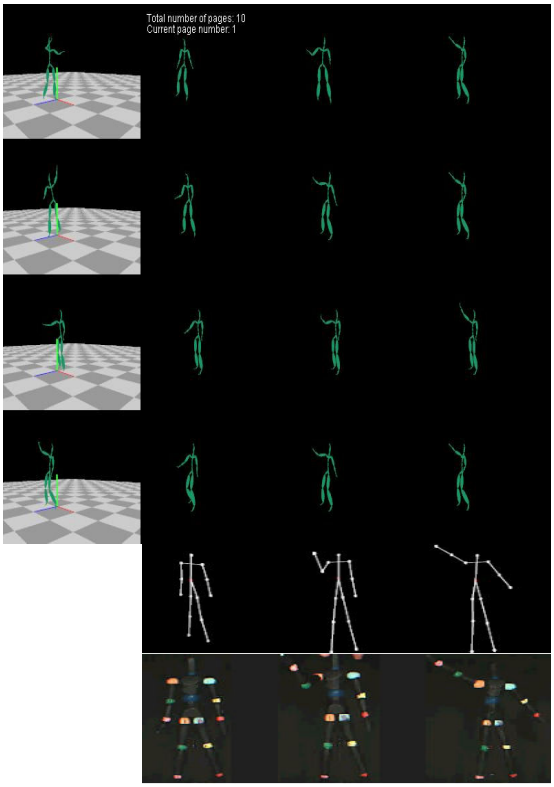

Figure 6. The top 4 ranked motion sequences are shown in the left column. The 3 poses within each motion sequence that match to the input keyframes are also shown to the right of each motion sequence playback. The 3 user specified keyframe poses are below the results in the order specified by user. The bottom-most row shows the original doll poses, from one camera, input by the user.

5. Conclusion

This paper presents a complete mocap data retrieval system that is simple, effective and cost efficient. Through the unique artist's doll interface, users of any skill level can easily pose the doll into keyframe poses. Then the series of poses are used to retrieve motion sequences that contain these poses in the order specified. The experimental results show that the system is successful in retrieving the desired results.

References

[1] Carnegie-Mellon MoCap Database, 2003. http://mocap.cs.cmu.edu

[2] Müller, M., Röder, T. and Clausen, M. Efficient content-based retrieval of motion capture data. *ACM Trans. Graph., 2005.*

[3] Müller, M. and Röder, T. Motion templates for automatic classification and retrieval of motion capture data. *Proceedings of SCA '06.*

[4] Davis, J., Agrawala, M., Chuang, E., Popovic', Z., and Salesin, D. 2003. A sketching interface for articulated figure animation. *In Proceedings of SCA '03, 320-328.*

[5] Li ,Q. L., Geng,W. D., Yu, T., Shen, X.J., Lau, N. and Yu, G. 2006. MotionMaster: authoring and choreographing Kung-fu motions by sketch drawings. *Proceedings of SCA '06.*

[6] Thorne, M., Burke D. and Panne, M. Motion Doodle: An Interface for Sketching Character Motion. *Proc. of SIGGRAPH '04.*

[7] Lee, J., Chai, J., Reitsma, P. S. A., Hodgins, J. K., and Pollard, N.S. Interactive control of avatars animated with human motion data. *Proceedings of SIGGRAPH '02.*

[8] Park, J., Jiang, B. and Neumann, U. Vision-Based Pose Computation: Robust and Accurate Augmented Reality Tracking. *Proceedings of IWAR '99.*

[9] Yin, K. and Pai D. K. FootSee: an interactive animation system. *Proceedings of SCA '03.*

[10] Esposito, C., Paley, W. B., Ong, J. C.: Of mice and monkeys: a specialized input device for virtual body animation. *In Proc. of the Symposium on Interactive 3D Graphics*, 1995, Monterey, CA, USA, pp. 109-114.

[11] Blumberg, B.M. Swamped! using plush toys to direct autonomous animated characters. *SIGGRAPH '98*

[12] Johnson, M. P., Wilson, A., Blumberg, B., Kline, C. and Bobick, A. Sympathetic interfaces: using a plush toy to direct synthetic characters. *Proceedings of CHI '99.*

[13] Barakonyi, I., Schmalstieg, D. Augmented Reality Agents in the Development Pipeline of Computer Entertainment. *Proc. of ICEC'05.*

[14] Barakonyi, I., Schmalstieg D. Augmented Reality in the Character Animation Pipeline. *SIGGRAPH sketch, SIGGRAPH '06*

[15] Sakamoto, Y., Kuriyama, S. and Kaneko, T. Motion map: image-based retrieval and segmentation of motion data. *Proceedings of SCA '04*