In Proceedings of IEEE Computer Vision and Pattern Recognition, pp. 475-482., Vol. 2, Madison, Wisconsin, USA, June 16-22, 2003.

Markerless Kinematic Model and Motion Capture from Volume Sequences

Chi-Wei Chu, Odest Chadwicke Jenkins, Maja J Matarić
Robotics Research Laboratory
Center for Robotics and Embedded Systems
Department of Computer Science
University of Southern California
Los Angeles, CA, USA 90089-0781
chuc,cjenkins,mataric@usc.edu

Abstract

We present an approach for model-free markerless motion capture of articulated kinematic structures. This approach is centered on our method for generating underlying nonlinear axes (or a skeleton curve) of a volume of genus zero (i.e., without holes). We describe the use of skeleton curves for deriving a kinematic model and motion (in the form of joint angles over time) from a captured volume sequence. Our motion capture method uses a skeleton curve, found in each frame of a volume sequence, to automatically determine kinematic postures. These postures are aligned to determine a common kinematic model for the volume sequence. The derived kinematic model is then reapplied to each frame in the volume sequence to find the motion sequence suited to this model. We demonstrate our method on several types of motion, from synthetically generated volume sequences with an arbitrary kinematic topology, to human volume sequences captured from a set of multiple calibrated cameras.

1. Introduction

The ability to collect human motion data is invaluable for applications such as computer animation, activity recognition, human-computer interfaces, and humanoid robot control and teleoperation. This fact is evidenced by the increasing amount of research geared towards developing and utilizing *motion capture* technologies. Typical motion capture mechanisms require that the subject be instrumented with several beacons or markers. The motion of the subject is then reconciled from the sensed positions and/or orientations of the markers. However, such systems can:

1. be prohibitively expensive;
2. require subjects to be instrumented with cumbersome markers;
3. greatly restrict the volume of capture;
4. have difficulty assigning consistent labels to occluding markers;
5. have difficulty converting marker data into kinematic motion.

An emerging area of research suited to address these problems involves uninstrumented capture of motion, or *markerless motion capture*. For markerless motion capture, subject data are acquired through some passive sensing mechanism and then reconciled into kinematic motion. Several *model-based* markerless capture approaches [6, 14, 3, 8, 4, 13, 22, 10] have been proposed that assume an *a priori* kinematic or body model. However, it would be preferable to eliminate this model dependence to capture both the subject's motion and kinematic model and, thus, perform *model and motion capture*.

In this paper, we introduce a solution for model-free vision-based markerless motion capture of subjects with tree-structured kinematics from multiple calibrated cameras. Using the functional structure of a motion capture system described by Moeslund and Granum [15], we summarize our approach for markerless motion capture. Moeslund and Granum describe a motion capture system as consisting of four components: initialization, tracking, pose estimation, and recognition. For initialization, a set of cameras is calibrated, using a method such as Bouguet's [2]. Because we assume no *a priori* kinematic model, no model initialization is necessary. We assume for the tracking component a system capable of capturing an individual subject's movement over time as a volume sequence, such as [16, 19]. The pose estimation component we develop is more than pose estimation because it performs model and

motion capture. In this component, we perform automatic model and posture estimation for each frame in the volume sequence. The models and postures produced from each frame are aligned in a second pass to determine a common kinematic model across the volume sequence. The common kinematic model is then applied to each frame in the volume sequence to perform pose estimation with respect to a consistent model. Our current methodology for capture does not include a recognition component. However, we envision our capture system providing vast amounts of motion data for other uses. For instance, Jenkins and Matarić [11] require long streams of motion data as demonstrations for automatically deriving vocabularies of behaviors and controllers for humanoid robot control.

Central to our model and motion capture approach is the ability to estimate a kinematic model and its posture from a subject's volume in a single frame. Towards this end, we developed a model-free method, called *nonlinear spherical shells (NSS)*, for extracting *skeleton point features* that are linked into a tree-structured *skeleton curve* for a particular frame within a motion. A skeleton curve is an approximation of the "underlying axes" of a subject, similar to principal curves [9], the axis of a generalized cylinder, or the wire spine of a posable puppet. NSS works by accentuating the underlying axes of a volume through Isomap nonlinear dimension reduction [21] and traversing the resulting "Da Vinci"-like posture. Isomap essentially eliminates the nonlinearities caused by joint rotations. Using skeleton curve provided via NSS, we automatically estimate the tree-structured kinematics and posture of the volume.

Several advantages arise in using our approach for markerless motion capture. First, our method is fast and accurate enough to be tractably applied to all frames in a motion. Our method can be used alone or as an initialization step for model-based capture approaches. Second, our dependence on modeling human bodies is eliminated. Automated model derivation is especially useful when the subject's kinematics differ from standard human kinematics due to missing limbs or objects the subject is manipulating. Third, the posture of the human subject is automatically determined without complicated label assignments.

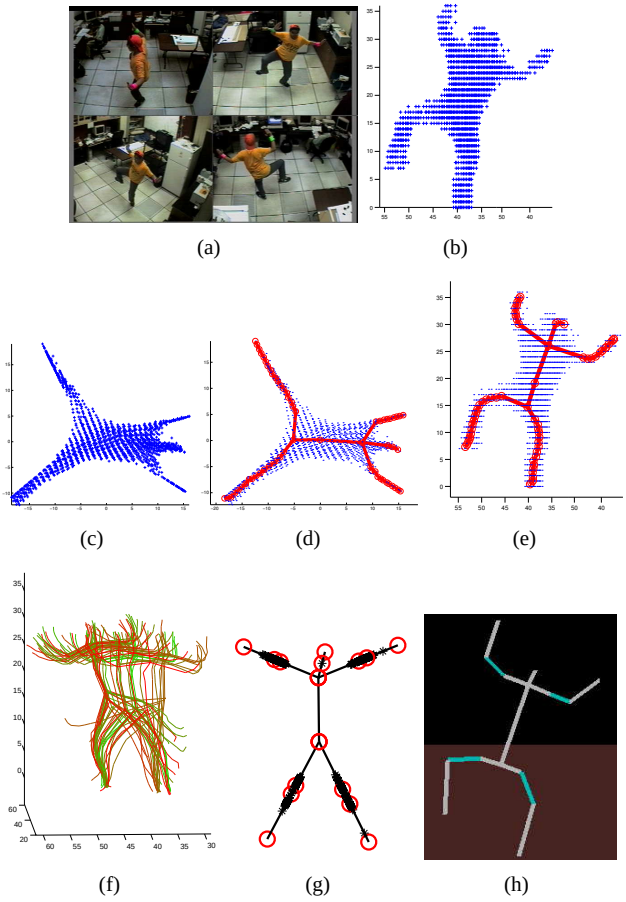

Figure 1. An illustrated outline of our approach. (a) A subject viewed in multiple cameras over time is used to build (b) a Euclidean space point volume sequence. Postures in each frame are estimated by: transforming the subject volume (c) to an intrinsic space pose-invariant volume, finding its (d) principle curves, project the principal curves to a (e) skeleton curve, and breaking the skeleton curve into a kinematic model. (f) Kinematic models for all frames are (g) aligned to find the joints for a normalized kinematic model. The normalized kinematic model is applied to all frames in the volume sequence to estimate its (h) motion, shown from an animation viewing program.

2. Volume Sequence Capture

The volume sequence data used for this work came from two sources. One source of captured volume data is from real-world subjects (humans) by multiple cameras. The other source was synthetically generated volume data from an articulated 3D geometry with arbitrary kinematics.

For real-world volume capture, we used an existing volume capture technique for multiple calibrated cameras. While not the focus of our work, this implementation does provide an adequate means for collecting volume sequences. The implementation is derived from the work of Penny et. al. [16] for real-time volume capture; however, several other approaches are readily available (e.g., [19, 4]). The capture approach is a basic brute-force method that checks each element of a voxel grid for inclusion in the

point volume. In our capture setup, we place multiple cameras around three sides of a hypothetical rectangular volume, such that each camera can view roughly all of the volume. This rectangular volume is a voxel grid that divides the space in which moving objects can be captured.

The intrinsic and extrinsic calibration parameters for the cameras are extracted using a camera calibration toolbox designed by [2]. The parameters from calibration allow us to precompute a look-up table for mapping a voxel to pixel locations in each camera. For each frame in the motion, silhouettes of foreground objects in the capture space are segmented within the image of each camera and used to carve the voxel grid. A background subtraction method proposed in [7] was used. It can then be determined if each voxel in the grid is part of a foreground object by counting and thresholding the number of camera images in which it is part of a silhouette. One set of volume data is collected for each frame (i.e., set of synchronized camera images) and stored for offline processing.

For synthetic data, we artificially create motion sequences from a synthetic articulated object with arbitrary tree-structured kinematics. We use this data to test our approach for objects readily available or controllable in the real world. In creating this data, we manually specified the kinematic model, rigid body geometries (cylinders), and joint angle trajectories. The motion of the object is converted into a volume sequence by scan converting each frame according to a voxel grid.

3. Nonlinear Spherical Shells

Nonlinear spherical shells (NSS) is our model-free approach for extracting a skeleton curve feature from a Euclidean-space volume of points. For NSS, we assume that nonlinearity of rigid-body kinematic motion is introduced by rotations about the joint axes. By removing these joint nonlinearities, we can trivially extract skeleton curves.

Fortunately for us, recent work on manifold learning techniques has produced methods capable of uncovering nonlinear structure from spatial data. These techniques include Isomap [21], Kernel PCA [18], and Locally Linear Embedding [17]. Isomap works by building geodesic distances between data point pairs on an underlying spatial manifold. These distances are used to perform a nonlinear PCA-like embedding to an *intrinsic space*, a subspace of the original data containing the underlying spatial manifold. Isomap, in particular, has been demonstrated to extract meaningful nonlinear representations for high dimensional data such as images of handwritten digits, natural hand movements, and a pose-varying human head.

The procedure for (NSS) works in three main steps:

1. removal of *pose-dependent* nonlinearities from the volume by transforming the volume into an intrinsic space using Isomap;
2. dividing and clustering the *pose-independent* volume such that principal curves are found in intrinsic space;
3. project points defining the intrinsic space principal curve into the original Euclidean space to produce a skeleton curve for the volume.

Isomap is applied in the first step of the NSS procedure to remove pose nonlinearities from a set of points compromising the captured human in Euclidean space. We use the implementation provided by the authors of Isomap (available at http://isomap.stanford.edu/). This implementation is applied directly to the volume data. Isomap requires the user to specify only the number of dimensions for the intrinsic space and how to construct local neighborhoods for each data point. Because dimension reduction is not our aim, the intrinsic space is set to have 3 dimensions. Each point determines other points within its local neighborhood using k-nearest neighbors or an epsilon sphere with a chosen radius.

The application of Isomap transforms the volume points into a pose-independent arrangement in the intrinsic space. The pose-independent arrangement is similar to a *"Da Vinci"* pose in 3 dimensions (Figure 2). Isomap can produce the Da Vinci point arrangement for any point volume with distinguishable limbs.

The next step in the NSS procedure is processing intrinsic space volume for principal curves. The definition of principal curves can be found in [9] or [12] as "self-consistent" smooth curves that pass through the "middle" of a d-dimensional data cloud, or nonlinear principal components. While smoothness is not our primary concern, we are interested in placing a curve through the "middle" of our Euclidean space volume. Depending on the posture of the human, this task can be difficult in Euclidean space. However, the pose-invariant volume provided by Isomap makes the extraction of principal curves simple, due to properties of the intrinsic space volume. Isomap provides an intrinsic space volume that is mean-centered at the origin and has limb points that extend away from the origin.

Points on the principle curves in intrinsic space be found by the following subprocedure (Figure 4):

1. partitioning the intrinsic space volume points into *concentric spherical shells*;
2. clustering the points in each partition;
3. averaging the points of each cluster to produce a principal curve point;
4. linking principal curve points with overlapping clusters in adjacent spherical shells.

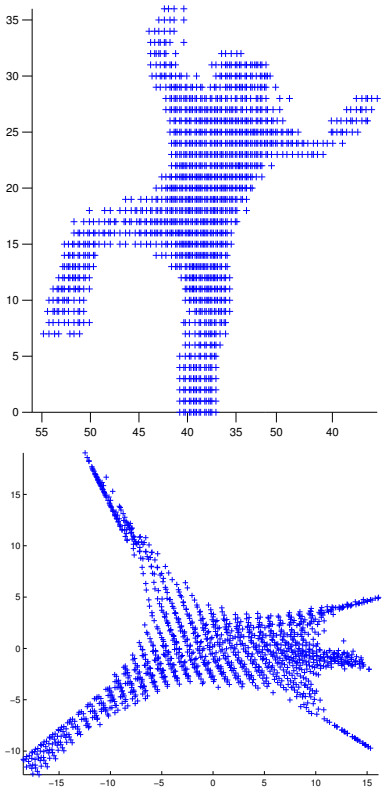

Figure 2. A captured human volume in Euclidean space (top) and its pose-invariant intrinsic space representation (bottom).

Clustering used for each partition was developed from the one-dimensional "sweep-and-prune" technique, described by Cohen et al. [5], for finding clusters bounded by axis-aligned boxes. This clustering method requires specification of a separating distance threshold for each axis rather than the expected number of clusters. The result from the principal curves procedure is a set of points defining the principal curves linked in a hierarchical tree-structure. These include three types of *indicator nodes*: a *root node* located at the mean of the volume, *branching nodes* that separate into articulations, and *leaf nodes* at terminal points of the body.

The final step in the NSS procedure projects the intrinsic space principal curve points onto a skeleton curve in the original Euclidean space. We use Shepard's interpolation [20] to map principal curve points onto the Euclidean space volume, producing skeleton curve points. The skeleton curve is formed by reapplying the tree-structured linkages of the intrinsic space principal curves to the skeleton curve points.

Other methods for volume skeletonization are available. These approaches include the distance coding [23], boundary peeling [23], and self-organizing feature maps [1]. For our purposes, it is important to ensure that the skeletonization produces a bordered 1-manifold, not necessarily a medial axis that is potentially a 2-manifold.

3.1. Skeleton Curve Refinement

The skeleton curve found by the NSS procedure will be indicative of the underlying spatial structure of the Euclidean space volume, but may contain a few undesirable artifacts. We handle these artifacts using a skeleton curve refinement procedure. The refinement procedure first eliminates *noise branches* in the skeleton curve that typically occur in areas of small articulation, such as the hands and feet. Noise branches are detected as branches with depth under some threshold. A noise branch is eliminated through merging its skeleton curve points with a non-noise branch.

The refinement procedure then eliminates noise for the root of the skeleton curve. Shell partitions around the mean of the body volume will be encompassed by the volume (i.e., contain a single cluster spread across the shell). The skeleton curve points for such partitions will be roughly located near the volume mean. These skeleton curve points are merged to yield a new root to the skeleton curve. The result is a skeleton curve having a root and two or more immediate descendants.

The minor variations in the topology of the skeleton curve are then eliminated by merging adjacent branching nodes. These are two skeleton points on adjacent spherical shells with adjacent clusters that both introduce a branching of the skeleton curve. The branches at these nodes are assumed to represent the same branching node. Thus, the two skeleton points are merged into a single branching node.

4. Model and Motion Capture

In this section, we describe the application of NSS within the context of our approach for markerless model and motion capture. The model and motion capture (MMC) procedure automatically determines a common kinematic model and joint angle motion from a volume sequence in a three-pass process. In the first pass, the procedure applies NSS independently to each frame in the volume sequence. From the skeleton curve and volume of each frame, a kinematic model and posture is produced that is specific to the frame. A second pass across the specific kinematic models of each frame is used to produce a single normalized kinematic model with respect to the frames in the volume sequence. Finally, the third pass applies the normalized model to each volume and skeleton curve in the sequence to produce estimated posture parameters.

The described NSS procedure is capable of producing skeleton curve features in a model-free fashion. The skeleton curve is used to derive a kinematic model for the volume in each frame. First, we consider each branch (occurring between two indicator nodes) as a kinematic link. The root node and all branching nodes are classified as joints. Each branch is then segmented into smaller kinematic links based on the curvature of the skeleton curve. This division is performed by starting at the parent indicator node and iteratively including skeleton points until the corresponding volume points become nonlinear. Nonlinearity is tested by applying a threshold to the skewness of the volume points with respect to the line between the first and last included skeleton point. When the nonlinearity occurs, a segment, representing a joint placement, is set at the last included skeleton point. The segment then becomes the first node in the determination of the next link and the process iterates until the next indicator node is reached. The length of these segments, relative to the length of the whole branch, is recorded in the branch. The specific kinematic models derived from the volume sequence may have different branch lengths and each branch may be divided into a different number of links.

In the second pass, a normalization procedure is used across all frame-specific models to produce a common model for the sequence. For normalization, we aim to align all specific models in the sequence and look for groupings of joints. The alignment method we used iteratively collapsed two models in subsequent frames using a matching procedure to find correspondences. The matching procedure uses summed error values of minimum squared distance between branch parents, the difference between angles of branches, and the difference between branch lengths. The normalization procedure finds the mapping that minimizes the total error value. We have also begun to experiment with a simpler alternative alignment procedure. This procedure uses Isomap to align by constructing neighborhoods for each skeleton point that considers its intra-frame skeleton curve neighbors and corresponding points on the skeleton curve in adjacent frames.

Once the specific kinematic models are aligned, clustering on each branch is performed to identify joint positions. Each branch is normalized by averaging the length of the branch and number of links in the branch. The location of the aligned joint locations along the branch forms a 1D data sequence. An example is shown (Figure 3) for a branch with an average number of joints rounded to three. In this figure, the joint positions roughly form three sparse clusters of joint points along the branch, with some outliers. To identify the joint clusters, we used a clustering method that estimates density of all joint locations and places a joint cluster where peaks in the density are found.

In the third pass, the common kinematic model is applied to the skeleton curve in each frame to find the motion of the model (Figure 3). The coordinate system of the root node of the model is always aligned to the world coordinate system. For every joint, the direction of the link is the Z axis of the joint's coordinate system. The Y axis of the joint is derived by the cross product of its Z axis and its parent's X axis. The cross product of the Y and Z axis is the X axis of the joint. The world space coordinate system for each joint is converted to a local coordinate system by determining its 3D rotational transformation from its parent. The set of these 3D rotations provides the joint angle configuration for the current posture of the derived model.

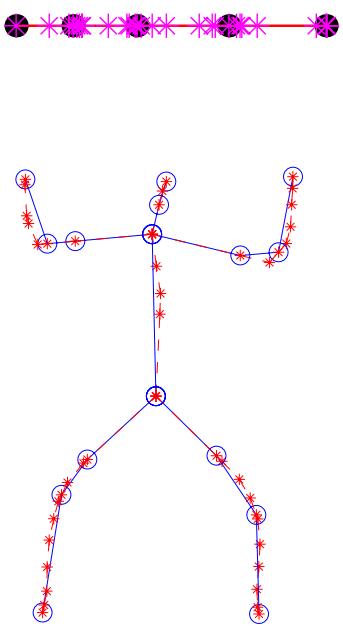

Figure 3. (top) Aligned segmentation points (stars) and joints clusters (circles) of one of the branches in the synthetic data. (bottom) The normalized kinematic model (circles as joints) with respect to the aligned skeleton curve sequence.

5. Results and Observations

In this section, we describe the implementation of our markerless model and motion capture approach and the results from its application to both captured human volume data and synthetic data. The human volume data contain two different motion sequences: waving and jumping jacks.

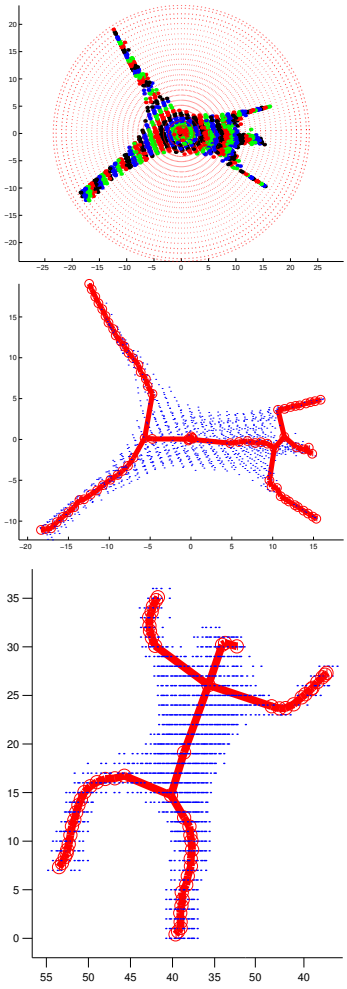

Figure 4. Partitioning of the pose-invariant volume (top), its tree-structured principal curves (middle), and project back into Euclidean space (bottom).

Our approach was implemented in Matlab, with our volume capture implementation in Microsoft Visual C++. The execution of the entire implementation was performed on a 350 MHz Pentium with 128 MB of memory.

For each human motion sequence, a volume sequence was captured and stored for offline processing by the model and motion capture procedure. Using the Intel Image Processing Library, we were able to capture volumes within a $80 \times 80 \times 50$ grid of cubic $50mm^3$ voxels at 10 Hz. Each volume sequence consisted of roughly 50 frames. Due to our frugal choices for camera and framegrabber options, our ability to capture human volumes was significantly restricted. Our image technology allowed for 320×240 image data from each camera, which produced several artifacts such as incorrectly activated voxels from shadows, occlusion ghosting, and image noise. This limitation restricted our capture motions to exaggerated, but usable, motion, where the limbs were very distinct from each other. Improving our proof-of-concept volume capture system, with more and better cameras, lighting, and computer vision techniques, will vastly improve our capture system, without having to adjust the model and motion capture procedure.

Using the captured volume sequences, our model and motion capture mechanism was able to accurately determine appropriate postures for each volume without fail. We used the same user parameters for each motion, consisting of an Isomap epsilon-ball neighborhood of radius $(50mm^3)^{1/2}$ and 25 for the number of concentric sphere partitions. In addition to accurate postures, the derived kinematic model parameters for each sequence appropriately matched the kinematics of the capture subject. However, for camera captured volume data, a significant amount of noise occurred between subsequent frames in the produced motion sequence. Noise is typical for many instrumented motion capture systems and should be expected when independently processing frames for temporally dependent motion. We were able to clean up this noise to produce aesthetically viable motion using standard low pass filtering.

When applied to synthetic data, our method can reconstruct its original kinematic model with reasonable accuracy. This data were subject to the problem of oversegmentation, i.e., joints are placed where there is in fact only one straight link. There are three causes for this problem. First, a joint will always be placed at branching nodes in the skeleton curves. A link will be segmented if another link is branching from its side. Second, the root node of the skeleton curve is always classified as a joint, even if it is placed in the middle of an actual link. Third, noise in the volume data may add fluctuation of the skeleton curves and cause unwanted segments.

Motions were output to files in the Biovision BVH motion capture format. Figure 5 shows the kinematic posture output for each motion. More images and movies of our results are available at http://robotics.usc.edu/˜cjenkins/markerless/.

In observing the performance of our markerless model and motion capture system, several benefits of our approach became evident. First, the relative speed of our capture procedure made the processing of each frame of a motion tractable. Depending on the number of volume points, the elapsed time for producing a posture from a volume by our Matlab implementation ranged between 60 and 90 seconds, with approximately 90 percent of this time spent for Isomap processing. Further improvements can be made to our implementation to speed up the procedure and process volumes with increasingly finer resolution. Second, our implementation required no explicit model of human kinematics,

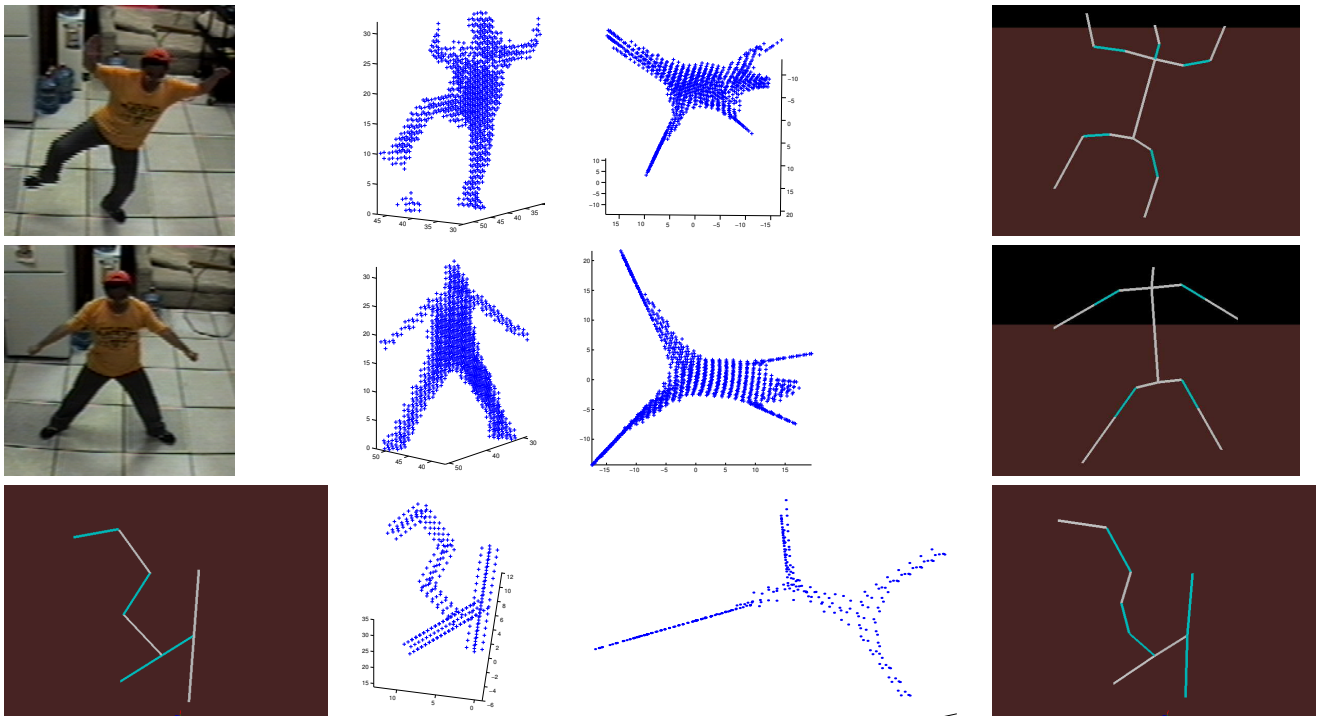

Figure 5. Results from producing kinematic motion for human waving, jumping jacks and synthetic object motion (rows). The results are shown as a snapshot of the performing human or object, the capture or generated point volume data, the pose-invariant volume, and the derived kinematic posture (columns).

no initialization procedure, and no optimization of parameters with respect to a volume. Our model-free NSS procedure produced a representative skeleton curve description of a human posture based on the geometry of the volume. Lastly, the skeleton curve may be a useful representation of posture in and of itself. Rigid-body motion is often represented through typically model-specific kinematics. Instead, the skeleton curve may allow for an expression of motion that can be shared between kinematic models, for purposes such as robot imitation.

6. Issues for Future Work

Using our current work as a platform, we aim to improve our ability to collect human motion data in various scenarios. Motion data are critically important for other related projects, such as the derivation of behavior vocabularies [11]. Areas for further improvements to our capture approach include: *i)* more consistent mechanism for segmenting skeleton curve branches, *ii)* different mechanisms for aligning and clustering joints from specific kinematic models in a sequence, *iii)* automatically deriving kinematic models and motion for kinematic topologies containing cycles (i.e., "bridges", volumes of genus greater than zero), *iv)* and exploring connections between model-free methods for robust model creation and initialization and model-based methods for robust temporal tracking, *v)* extensions to Isomap for volumes of greater resolutions and faster processing of data, *vi)* using better computer vision techniques for volume capture to extend the types subject motion that can be converted into kinematic motion.

7. Conclusion

We have presented an approach for model-free markerless model and motion capture. In our approach, a kinematic model and joint angle motion are extracted from volume sequences of subjects with arbitrary tree-structured kinematics. We have presented the application of Isomap nonlinear dimension reduction to volume data for both the removal of pose-dependent nonlinearities and extractable skeleton curve features for a captured human volume. We proposed an approach, nonlinear spherical shells, for extracting skeleton curve features from a human volume. This

feature extraction is placed within the context of a larger approach for capturing a kinematic model and corresponding motion. Our approach was successfully applied to different types of subject motion.

8. Acknowledgments

This research was partially supported by the DARPA MARS Program grant DABT63-99-1-0015 and ONR MURI grant N00014-01-1-0890. The authors wish to thank Gabriel Brostow for valuable discussions and feedback.

References

[1] C. M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[2] J.-Y. Bouguet. Camera calibration toolbox for matlab. http://www.vision.caltech.edu/bouguetj/calib_doc/index.html.

[3] C. Bregler and J. Malik. Tracking people with twists and exponential maps. In *IEEE Conference on Computer Vision and Pattern Recoginition*, pages 8–15, Santa Barbara, CA, USA, 1998.

[4] K. M. Cheung, T. Kanade, J.-Y. Bouguet, and M. Holler. A real time system for robust 3d voxel reconstruction of human motions. In *Proceedings of the 2000 IEEE Conference on Computer Vision and Pattern Recognition (CVPR '00)*, volume 2, pages 714 – 720, June 2000.

[5] J. D. Cohen, M. C. Lin, D. Manocha, and M. K. Ponamgi. I-COLLIDE: An interactive and exact collision detection system for large-scale environments. In *Proceedings of the 1995 symposium on Interactive 3D graphics*, pages 189–196, 218, Monterey, CA, USA, 1995. ACM Press.

[6] J. Deutscher, A. Blake, and I. Reid. Articulated body motion capture by annealed particle filtering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 126–133, Hilton Head, SC, USA, 2000.

[7] A. R. Francois and G. G. Medioni. Adaptive color background modeling for real-time segmentation of video streams. In *Proceedings of the International on Imaging Science, Systems, and Technology*, pages 227–232, Las Vegas, NV, USA, June 1999.

[8] D. Gavrila and L. Davis. 3d model-based tracking of humans in action: A multi-view approach. In *IEEE Conference on Computer Vision and Pattern Recoginition*, pages 73–80, San Francisco, CA, USA, 1996.

[9] T. Hastie and W. Stuetzle. Principal curves. *Journal of the American Statistical Association*, 84:502–516, 1989.

[10] A. Hilton, J. Starck, and G. Collins. From 3d shape capture to animated models. In *1st International Symposium on 3D Data Processing Visualization and Transmission (3DPVT'02)*, pages 246–257, Padova, Italy, Jun 2002.

[11] O. C. Jenkins and M. J. Matarić. Automated derivation of behavior vocabularies for autonomous humanoid motion. In *To appear in the Second International Joint Conference on Autonomous Agents and Multiagent Systems (Agents 2003)*, Melbourne, Australia, July 2003.

[12] B. Kegl, A. Krzyzak, T. Linder, and K. Zeger. Learning and design of principal curves. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(3):281–297, 2000.

[13] J. Luck, D. Small, and C. Q. Little. Real-time tracking of articulated human models using a 3d shape-from-silhouette method. In *Robot Vision, International Workshop RobVis*, volume 1998, pages 19–26, Feb 2001.

[14] I. Mikić, M. Trivedi, E. Hunter, and P. Cosman. Articulated body posture estimation from multi-camera voxel data. In *IEEE International Conference on Computer Vision and Pattern Recognition*, pages 455–460, Kauai, HI, USA, December 2001.

[15] T. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81(3):231–268, March 2001.

[16] S. G. Penny, J. Smith, and A. Bernhardt. Traces: Wireless full body tracking in the cave. In *Ninth International Conference on Artificial Reality and Telexistence (ICAT'99)*, December 1999.

[17] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[18] B. Scholkopf, A. J. Smola, and K.-R. Muller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Computation*, 10(5):1299–1319, 1998.

[19] S. M. Seitz and C. R. Dyer. Photorealistic scene reconstruction by voxel coloring. In *Proc. Computer Vision and Pattern Recognition Conf.*, pages 1067–1073, 1997.

[20] D. Shepard. A two-dimensional interpolation function for irregularly-spaced data. In *Proceedings of the ACM national conference*, pages 517–524. ACM Press, 1968.

[21] J. B. Tenenbaum, V. de Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[22] C. R. Wren, A. Azarbayejani, T. Darrell, and A. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.

[23] Y. Zhou and A. W. Toga. Efficient skeletonization of volumetric objects. *IEEE Transactions on Visualization and Computer Graphics*, 5(3):196–209, July-September 1999.