3D Reconstruction of Human Motion Through Video

Thomas Keemon

Advisor: Professor Hao Jiang

2010 Undergraduate Honors Thesis

Computer Science Department, Boston College

Abstract

In this thesis, we study computer vision methods to capture 3D human movements in videos. The goal is to extract high quality movement from general videos using minimum human interaction. With a few mouse clicks on the body joints in each video frame, all the possible 3D body configurations are automatically constructed and their likelihoods are quantified using a prior trained with millions of exemplars from the CMU motion capturing database. The 3D body movement is optimized using dynamic programming using the pose hypotheses and the temporal smoothness constraints. The proposed method can be used for unconstrained motion capturing and our experiments show that it is efficient and accurate. We further study two applications based on the proposed motion capturing method. The first one is to animate characters using the motion captured from videos. The second one is for sports performance analysis. With the 3D movement information, we can measure body part speed, coordination, and various other parameters.

Contents

1. Introduction

Human pose estimation and action recognition are two very large areas of research within the domain of computer vision. Humans possess an uncanny ability to both detect and recognize different human poses and forms of motion, however, training a computer to do the same thing is not such a trivial task.

The first person to bring the study of human motion into a laboratory setting was Gunnar Johansson [8]. Johansson's experiments involved illuminating a number of key points on the human body as it performed different types of movements including walking, running, and dancing. The results of these experiments indicate that using just 10–12 illuminated points was enough to evoke a strong impression of human motion. Essentially, Johansson discovered that the collection of moving points resulted in a special response from the brain that gave structure to the entire system.

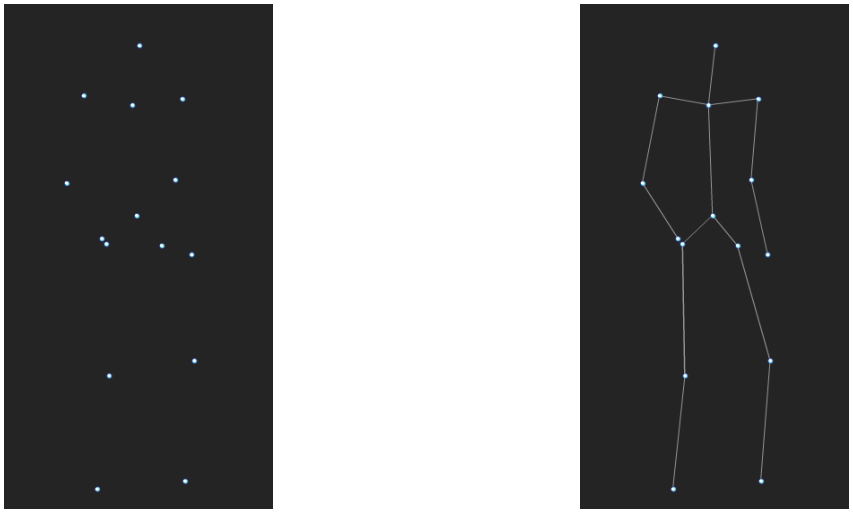


Figure 1. The collection of 15 seemingly random dots in the image on the left in fact represent a human pose. The image on the right connects the dots in a meaningful way, similar to how Johansson's experiments gave structure to seemingly random patterns of dots. Images taken from BML Walker [14].

If the brain possesses this special method for recognizing biological motion, why has a computer not been used to replicate it? The answer is because the underlying mechanism for the system is not completely understood. The brain itself can be looked at as a "black box." Experiments can be run on a human subject where the visual stimulus sent to the brain is controlled and the resulting subject response can be observed, however, the actual steps in the neural pathway responsible for transforming the input to the output response remain unknown.

Because a computer makes such a poor human brain, problem solving that utilizes a computer often relies on computationally intensive approaches. From this sort of computational standpoint, the human body can be though of as a tree structure. With this kind of depiction, it is easy to see how the extremities are related back to the torso (the root) and how the movement and range of motion of the extremities are constrained by the intermediate joints (the intermediate nodes).

Research in the area of human pose estimation has been approached from many different angles in an attempt to make a robust and accurate system. There are two parts to the problem of human pose estimation. The first is finding a pose in an image. A couple of approaches to this problem include using background subtraction, template matching [5] or detecting motion in a video sequence. The second part of the problem involves mathematically describing the pose. This has been done with both tree and non-tree [7] structures.

The same two problems exist in the area of human action recognition. Detecting action can take the form of detecting movement in sequential frames, finding motion descriptors [6], or some form of template matching [13]. Recognizing actions, on the other hand, often uses some sort of probabilistic approach, the most common being Markov chains [1, 2, 12] or some more complex Bayesian based system [3].

2. Overview

This thesis focuses on generating a 3D stick figure from either a single still image or a sequence of frames from a video. Section 3 details how a 3D pose is generated from a single still image, while section 4 shows how multiple 3D poses are strung together into a coherent action. Section 5 shows the results from both the single frame and multiple frame reconstructions. Finally, section 6 demonstrates a number of different applications for this system, related work in the field, and future work.

3. Single Frame Reconstruction

Reconstructing a 3D pose from an image involves three steps:

1. Collecting user input
2. Generating all possible poses
3. Finding the most likely pose

3.1 Collecting user input

The first step in the process involves the user marking the joints of the person in the provided image. Given an input image, the user clicks on the shoulder, elbow, and wrist for both arms, as well as the hip, knee, and ankle for both legs. In total, 12 joints are needed. If any of the joints are not visible in the image, the user must give as close an approximation as possible. As a result of the collected points, a stick figure is generated with 11 line segments.

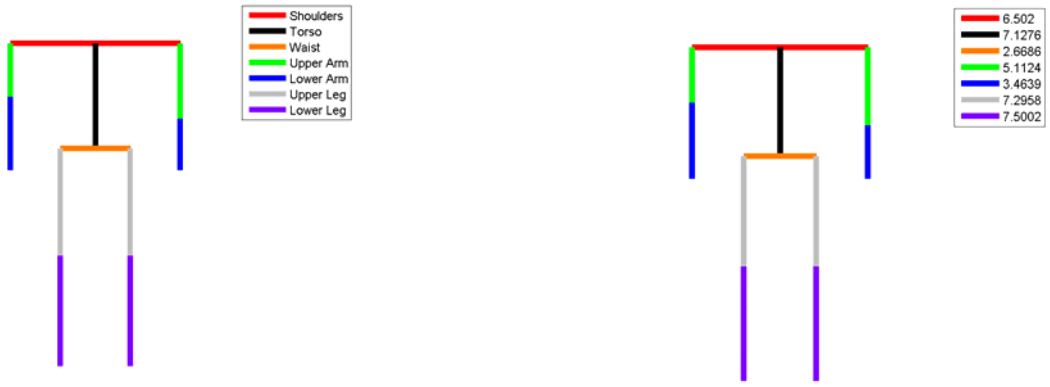


Figure 2. The image on the left shows a stick figure with all of its line segments labeled. The legend on the right image indicates assumed lengths for all line segments.

3.2 Generating all possible poses

In mathematical terms, a projection is defined as the mapping of a set of points and lines from one plane to another. Within the scope of this project, the generated stick figure can be seen as a projection from a 3D plane to a 2D plane. This type of projection is characterized by a loss of depth information. Essentially, all 3D points (X,Y,Z) in the system become (X,Y).

An easy way of thinking about a 3D to 2D projection is to picture someone shining a flashlight against a flat wall. If an individual were to place his hand midway between the flashlight and the wall, the hand would cast a shadow. In this situation the hand is an object in a 3D environment and the shadow is the projection of the hand onto a 2D plane. Given this setup, it is easy to imagine how either moving or rotating your hand would change the shape of the shadow against the wall.

Using this logic, consider a line in 3 dimensions with the points (x_1, y_1, z_1) and (x_2, y_2, z_2) and its projection onto a 2D plane with points (x_1, y_1) and (x_2, y_2). Calculating the distance between the two points in their respective dimensions is simply an application of the Pythagorean Theorem:

$$C_{2D} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (1)$$

$$C_{3D} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2} \quad (2)$$

Where C_{2D} and C_{3D} are distances between the two points in 2 dimensions and 3 dimensions respectively. Similarly, since C_{2D} and C_{3D} both utilize the same X and Y coordinates, they can be related to one another in the following way:

$$C_{3D}^{\ 2} = C_{2D}^{\ 2} + Z^2 \quad (3)$$

where Z is defined as $(z_2 - z_1)$.

Looking back at the stick figure, the goal is to calculate some sort of depth information. This could be done utilizing equation 3, however, there are still two unknowns. This is overcome by fixing the value of C_{3D} for all line segments in the stick figure (Figure 2). Solving equation 3 for Z yields:

$$Z = \sqrt{C_{3D}{}^2 - C_{2D}{}^2} \qquad (4)$$

Having the length, however, does not completely solve the problem. There is still a problem of the line segment's orientation on the Z axis. This is shown graphically in figure 3.

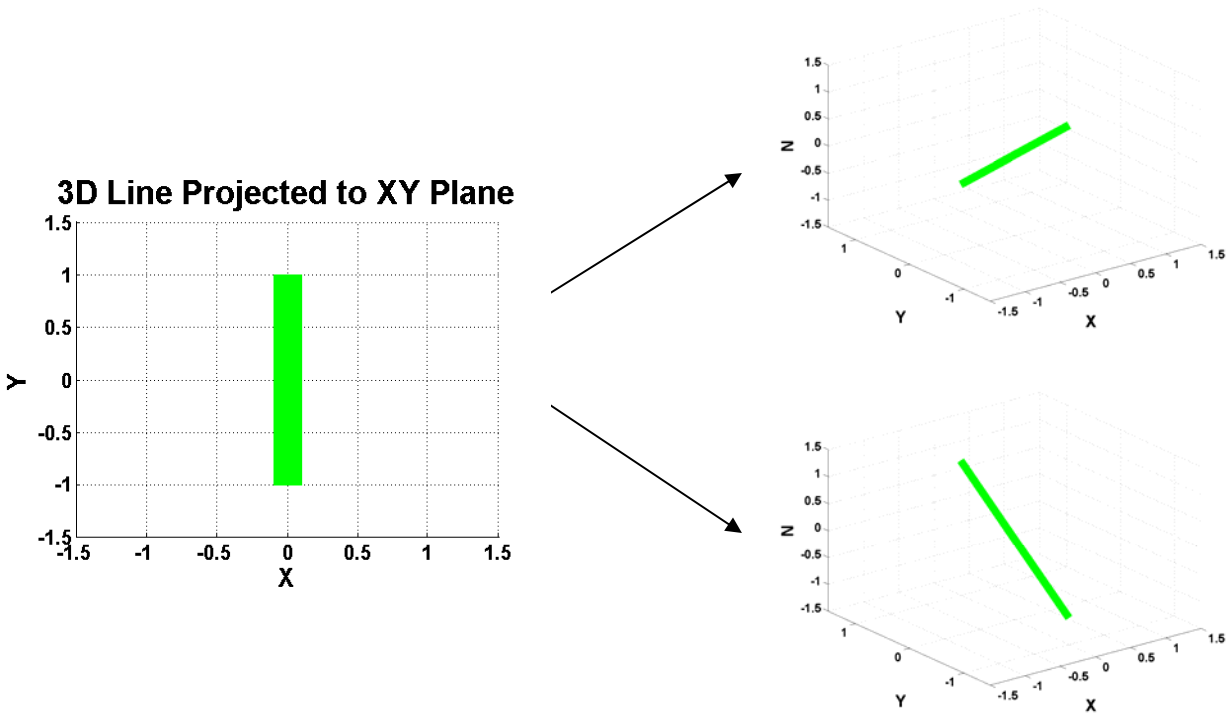


Figure 3. The line graphed on the left represents a projection from a 3 dimensional space to the XY plane. The end points of this line are (0,1) and (0,-1). The problem with orientation on the Z-axis is demonstrated here because either of the two lines on the right could be used to create the projection on the left. The line on the top right graph has end points (0,1,-1) and (0,-1,1), while the line on the bottom right graph has points (0,-1,-1) and (0,1,1).

Given that there are 11 line segments in the stick figure and two possible solutions for each segment, there are a total of 2048 possible 3D pose configurations.

3.3 Finding the most likely pose

After all 2048 possible poses are generated, a method is needed in order to determine which configuration is most likely to occur. This was accomplished through the use of a Gaussian mixture model. A Gaussian mixture model is a collection of some number of Gaussian functions in a fixed number of dimensions.

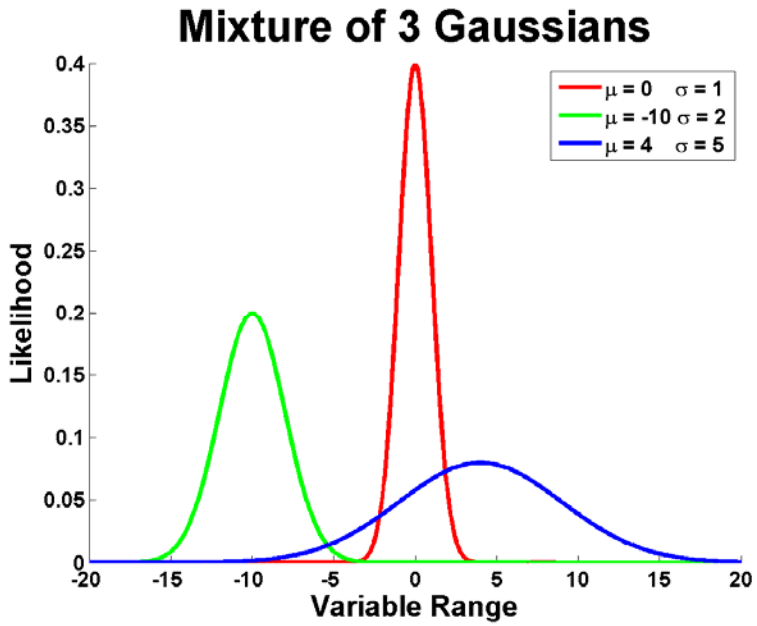


Figure 4. The above graph depicts a collection of 3 Gaussian distributions in one dimension making up a Gaussian mixture model.

Given any point in one dimension, the above model can numerically define how likely that point is to occur in the mixture through a probability density function (pdf). This same frame of logic is applied to describing how likely a human poses is.

In order to create a Gaussian mixture model for this project, a large database was necessary. Carnegie Melon University's Motion Capture Database (MoCap) is a database filled with millions of examples of human poses programmatically recorded using special cameras and tracking dots [15]. This database served as the source of the data for the Gaussian mixture model. In order to create the model, the poses were first extracted from the database as 12 element vectors. The vectors were then clustered in a 12D subspace using the expectation-maximization (EM) algorithm. The algorithm yielded estimates for μ (mean) and σ (standard deviation) for each cluster, which define the model. The EM algorithm was constrained to use only 50 clusters.

After creating the Gaussian mixture model, quantifying the likelihood of all 2048 poses was accomplished by simply comparing each individual pose against the model. Each comparison yielded a pdf and the pose with the highest pdf constituted the most likely configuration.

The above process was further optimized by utilizing two separate Gaussian mixture models; one for the upper body and one for the lower body. By doing this, only 96 unique configurations were necessary in order to describe all 2048 possible poses. After finding the largest pdf for both the upper and lower bodies, the two configurations were simply combined. Because the Gaussian mixture models were computed ahead of time, this optimization increased the speed and decreased the amount of memory used by the algorithm.

4. Multiple Frame Reconstruction

Generating a 3D pose from a video sequence mirrors the procedure for the single frame reconstruction with one caveat. The ordered sequence of video frames adds a temporal aspect to the system. Because of this, a generated pose must not only be likely in its own regard, but must also be related to the poses that occur before and after it in the sequence. This essentially turns multiple frame reconstruction into a graph problem.

4.1 Travel Cost

Just as in the single frame reconstruction, user input is required to label the joints in all frames of the video sequence. All possible pose configurations are then generated for each frame and quantified with a pdf. There are two conditions that need to be met in order to generate an accurate pose sequence. First of all, the pdf of the pose must be maximized, indicating that the pose is likely to occur in the first place. Second, poses in a sequence must be relatively close to one another in terms of some distance metric. In effect, the goal is to find make a series of frame to frame connections that satisfy the following equation. The travel cost to go from pose *i* of frame *n-1* to pose *j* of frame *n* is defined by:

$$S = dist(pose_{n-1,i}, pose_{n,j}) - \alpha PDF_{n-1,i} \quad (5)$$

where *dist* is a function representing the sum of the differences in Euclidean distances between the provided poses, $PDF_{n-1,i}$ is the pdf of the i^{th} pose of frame *n-1* and α is some scaling factor. For each frame in the sequence, there are 2048 possible poses that could potentially connect to any of the 2048 poses in the next frame. Because of this, 4,194,304 connections need to be computed for each pair of frames.

4.2 Dynamic Programming

This situation describes a large connected graph problem.

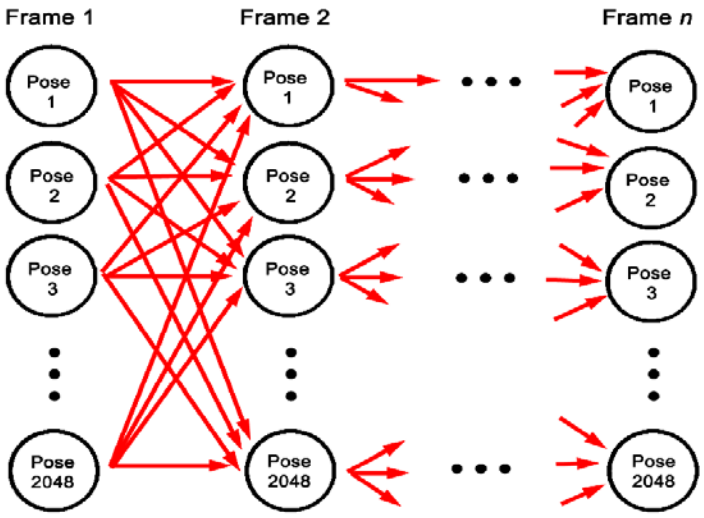


Figure 5. The graph depicted here represents the connections of 2048 human pose configurations across *n* frames.

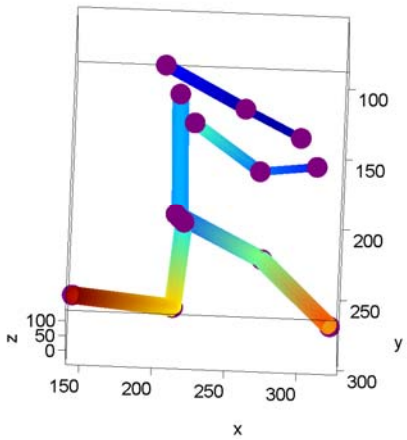
Given that a video sequences can exceed 100 frames, hundreds of millions of computations are needed. This type of large computation is optimized using dynamic programming. Using equation 5, navigating the graph can be mathematically represented as:

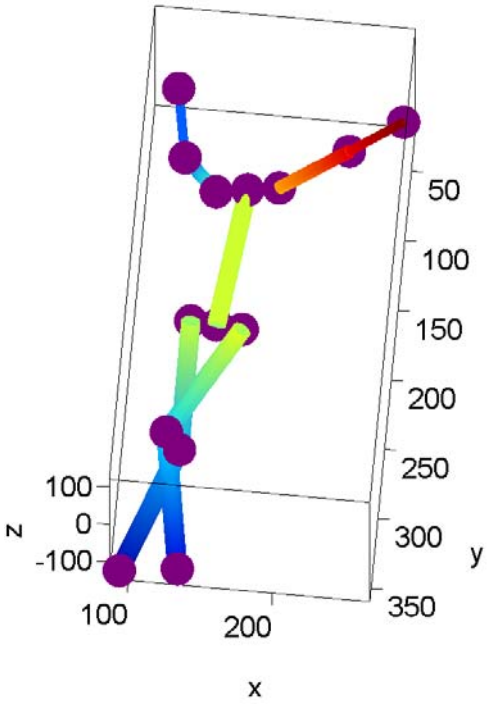
$$S_{n,j} = \arg\min_{i}(dist(pose_{n-1,i}, pose_{n,j}) - \alpha PDF_{n-1,i} + S_{n-1,i}) \qquad (6)$$

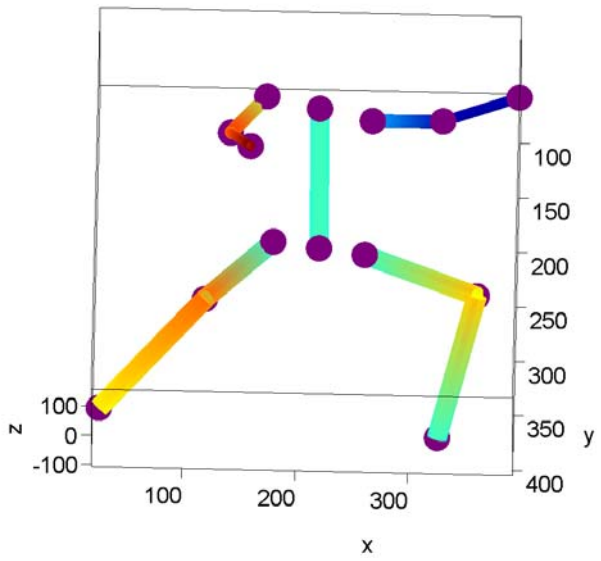
Each pose *j* of frame *n* is given a score based on equation 5. The only change is that the score from the previous frame's incoming pose is added as well. After the final frame is reached, determining the sequence of poses is simply a matter of backtracking through the graph starting with the minimum *S* value in the final frame.

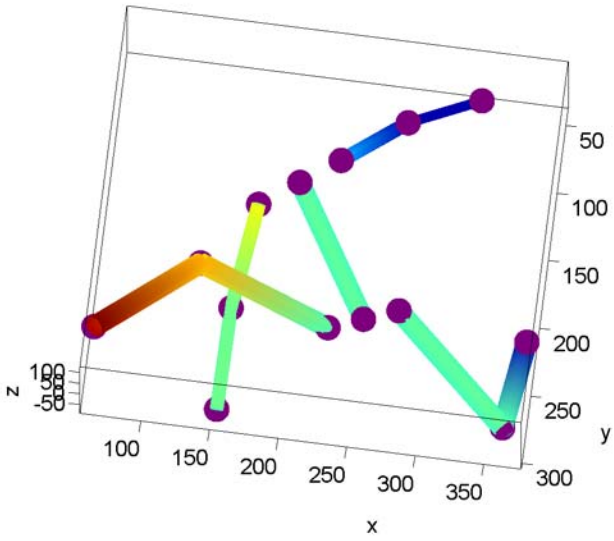
5. Results

Single Frame Reconstruction

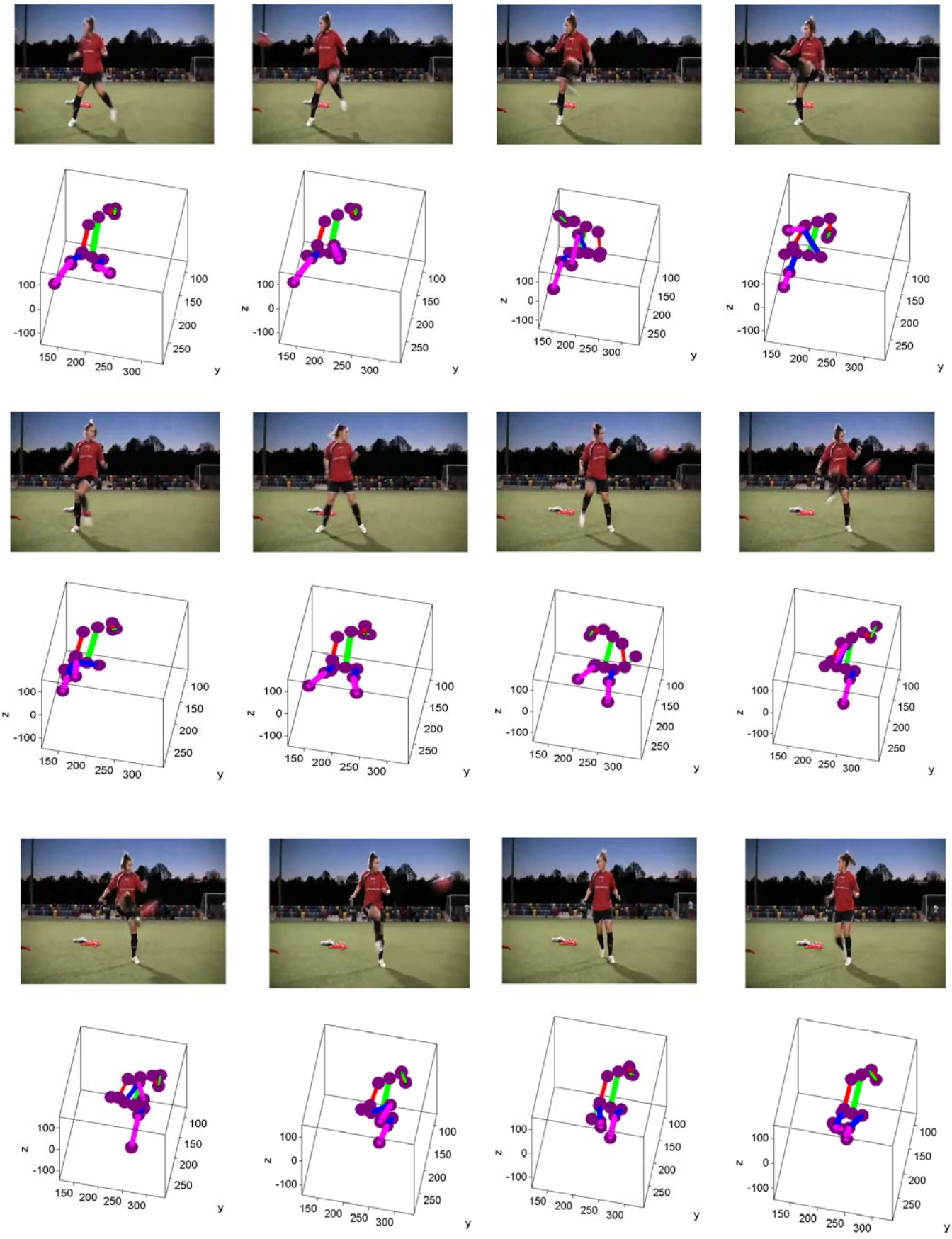


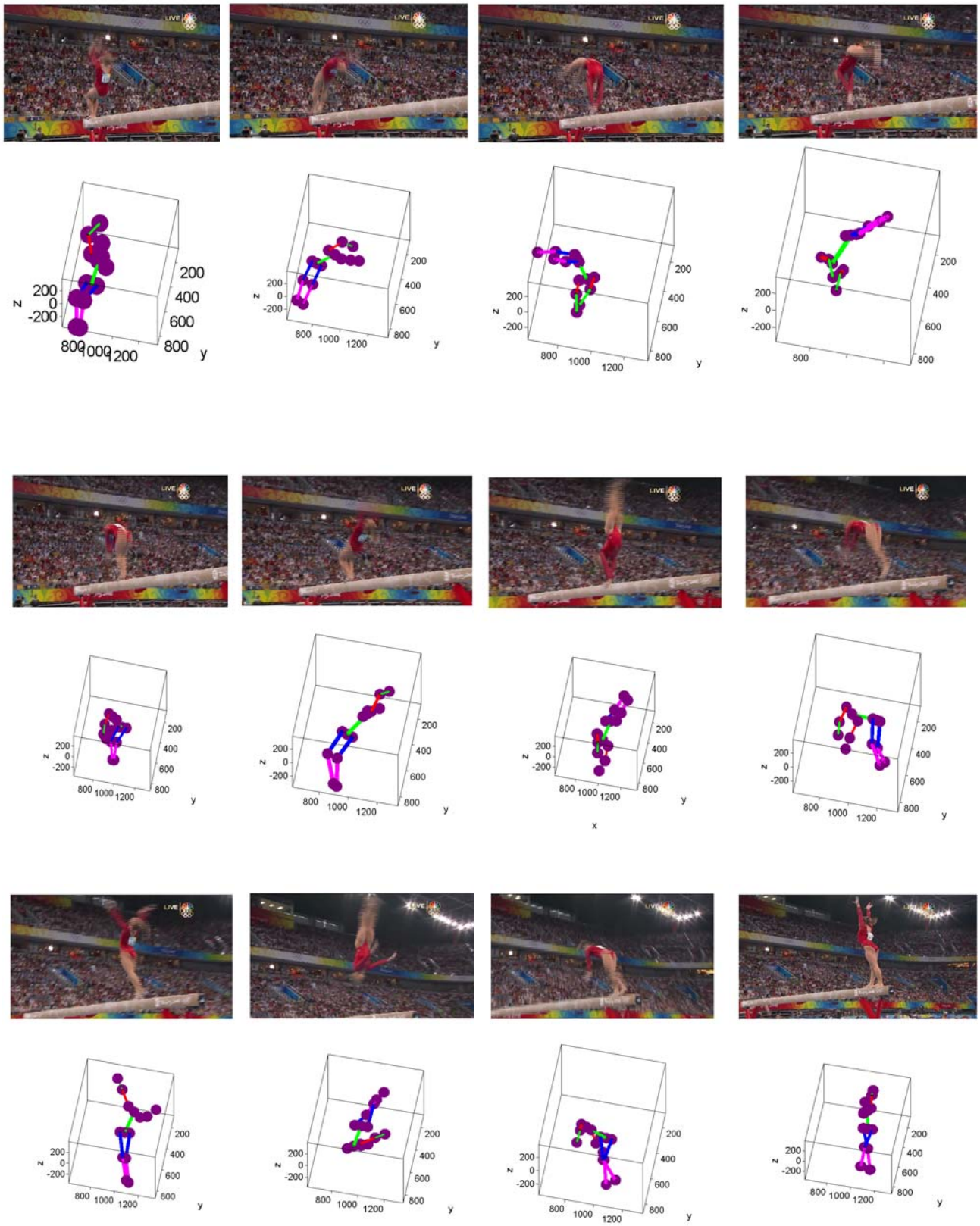



z
100
0
-100
100
200
300
x
100
150
200
250
300
350
400
y


z
100
50
-50
100
150
200
250
300
350
x
50
100
150
200
250
300
y

Multiple Frame Reconstruction




z
y
x
200
0
-200
400
600
800
1000
1200

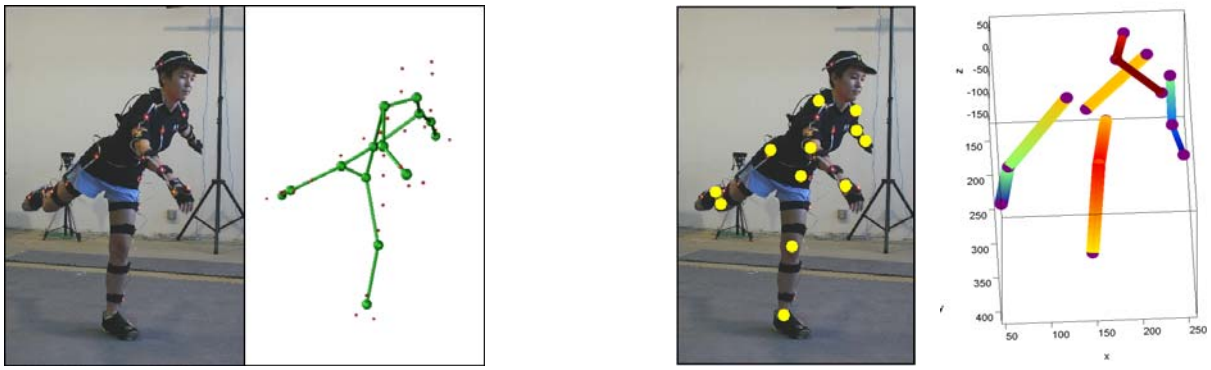
6. Discussion

As the above results show, this method of 3D human pose reconstruction is indeed possible in both single images as well as video sequences. Although this system is not 100% accurate, the results are good enough to be both meaningful and useful.

6.1 Applications

This thesis examines two applications directly related to 3D human pose reconstruction. The first is character animation and the second is sports performance analysis.

Traditionally, creating movement data for character animation is very limited. Individuals wearing suits with tracking markers were constrained to moving within small areas where special cameras monitor their actions. Because of this, trackable actions are limited in many ways. This 3D pose reconstruction offers an unconstrained form of motion capture that requires no special equipment.



(a) (b)

Figure 6. (a) shows person wearing trackable markers with the resulting computer generated pose next to it. (b) shows the same image with user marked joints. The resulting 3D pose is shown to the right.

As you can see in figure 6b, without any special cameras or use of the tracking markers, a similar 3D pose is generated. In effect, this means that any video can be used to generate a series of 3D human poses. The joint locations of the poses can then simply be plugged into any kind of character animation software and the actions of the original video can be acted out by any range of characters.

The second application of this type of software is for sports performance analysis. A computer is capable of conveying large amounts of data after recording and monitoring an action performed by an athlete. By simply tracking the individual points of the pose, the speed and direction of each individual joint can be calculated.

Figure 7. The speed and direction of all four extremities are displayed through this gymnast's balance beam performance.

Even with such a simple setup, the velocity of various body parts can be analyzed, multiple frames can be used to calculate their respective accelerations, and overall efficiency can be hypothesized. In addition to all of this, by simply including additional information with the pose, other aspects of the action can be examined. Estimating the weight of the individual could yield information such as the amount of work done by different limbs or torque experienced by different joints.

6.2 Related Work

Within the scope of this thesis, there has recently been some work done by other individuals, the most notable of which is Jinxiang Chai. Professor Chai currently teaches in the Department of Computer Science and Engineering at Texas A&M University. His work includes generating human poses from still images [9] as well as video sequences [10]. The methods used are very similar, but differ in a few key points. Professor Chai's method still requires a user to mark key points within an image, however, 18 points of interest are necessary. As a result, the human pose is defined by 17 line segments. This system makes no assumptions about the lengths of the line segments, but as a tradeoff, five images are needed of a single pose in order to remove any ambiguity.

In addition to the above publications, Professor Chai has a publication describing a system that generates human poses based on a collection of millions of example poses [11]. This approach is very similar to the Gaussian mixture model prior used in this project.

6.3 Future work

There are two major potential areas of improvement for this project. The first is to have some way of automatically tracking 2D joint positions. This was a problem for two reasons. First of all, many of the videos used were not very high quality. Trying to find key points in a motion blurred image is a difficult task for a person, let alone a computer. The 3D reconstruction is very much dependant on the accuracy of the joint locations. A single outlier would almost certainly change a pose tremendously. The second problem is occlusion. This can either be the result of an object appearing between the camera and the person being recorded, or the person turning in such a way that they are only partially visible. If some joint locations are

not tracked, then a 3D reconstruction would become meaningless because the human body would not be complete when inspected from different angles.

The next major improvement that should be made is some form of collision detection. This collision detection can refer to either the body as it relates to touching itself or other objects in its immediate environment. As it stands, the collection of points in a generated 3D human pose know little else than how to connect in order to form a stick figure. Because of this, poses that have legs or arm crossed sometimes have one limb passing through another. Problems like this could be avoided by adding the constraint that two limbs cannot simultaneously occupy some area in space.

Overall, this software represents a good starting point for human pose reconstruction. In addition to the possible applications described above, other future directions for this project include performance analysis, tracking multi-person interaction, or efficiently creating computer simulations involving people.

7. Acknowledgements

Professor Jiang

Your guidance in completing this project was invaluable. There were many times when the task at hand seemed too overwhelming, yet you always had time to sit down with me and set me in the right direction. This project would not have nearly approached the scope that it did had it not been for your continued support.

Professor Yu

I regard working in your research lab as being one of the most beneficial and influential experiences of my college career. Thank you for taking me under your wing and teaching me the rigors of academic research.

Professor Martin

Your computation photography class was by far the most difficult class I have taken at Boston College. That being said, I would not be here as a computer science major nor involved with computer vision had your teaching not made me fall in love with both subjects.

8. References

[1] Ahmad, M. and Lee, S. HMM-based Human Action Recognition Using Multiview Image Sequence. In Proceedings of the 18th International Conference on Pattern Recognition, 2006.

[2] Brand, M., Oliver, N., Pentland, A. Coupled Hidden Markov Models for Complex Action Recognition. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 1996.

[3] Bobick, A. F. and Ivanov, Y. A. Action Recognition using Probabilistic Parsing. In Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, 1998.

[4] Chen, Y. and Chai, J. 3D Reconstruction of Human Motion and Skeleton from Uncalibrated Monocular Video. In Proceedings of IEEE Asian Conference of Computer Vision, 2009.

[5] Dimitrijevic, M., Lepetit, V., Fua, P. Human Body Pose Recognition Using Spatio-Temporal Templates. In Workshop on Modeling People and Human Interaction, Beijing, China, Oct 15–21, 2005.

[6] Efros, A., Berg, A., Mori, G., Malik, J. Recognizing Action at a Distance. In Proceedings of IEEE International Conference on Computer Vision, 2003.

[7] Jiang, H. and Martin D. R. Global Pose Estimation Using Non-Tree Models. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, 2008.

[8] Johansson, G. Visual Perception of Biological Motion and a Model for its Analysis. Perception and Psychophysics. 14, pp. 201–211, 1973.

[9] Wei, X. and Chai, J. Modeling 3D Human Poses from Uncalibrated Monocular Images. In Proceedings of IEEE International Conference on Computer Vision, 2009.

[10] Wei, X. and Chai, J. VideoMocap: Modeling Physically Realistic Human Motion from Monocular Video Sequence. To Appear in ACM Transactions on Graphics, 2010.

[11] Wei, X., and Chai, J. Intuitive Interactive Human Character Posing with Millions of Example Poses. To Appear in IEEE Computer Graphics and Applications, 2009.

[12] Yamato, J., Ohya, J., Ishii, K. Recognizing Human Action in Time-Sequential Images using Hidden Markov Model. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 379–385, 1992.

[13] Yilmaz, A. and Shah, M. Actions Sketch: A Novel Action Representation. In Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005.

9. Other Resources

[14] Biomotion Lab. http://www.biomotionlab.ca/

[15] CMU Graphics Lab Motion Capture Database, mocap.cs.cmu.edu, developed with funding from NSF EIA-0196217