Title
Visualization of Video Sequence in 3D Space

Author
Wang, Chenhui

Publication Date
2014

Peer reviewed|Thesis/dissertation

University of California
Los Angeles

Visualization of Video Sequence in 3D Space

A thesis submitted in partial satisfaction
of the requirements for the degree
Master of Science in Statistics

by

Chenhui Wang

2014

Abstract of the Thesis

Visualization of Video Sequence in 3D Space

by

Chenhui Wang

Master of Science in Statistics

University of California, Los Angeles, 2014

Professor Song-Chun Zhu, Chair

Understanding of a incoming video sequence is one of the key parts of modern computer vision. Visualization of all the details that a computer understands about a certain video sequence is a very important tool to evaluate a vision system. In this paper, we propose a method for visualizing a 2D video sequence in a 3D space, which is based on the data of DARPA project.

The visualization started with the reconstruction of 3D space depended on the 2D video sequence data information. The use of homographic transformation for a structured light system reduced the computational cost of the system, since it only needed one image of the underlying environment. The smoothing of 3D trajectories of moving person was calculated from detection results. All the human skeleton models were from a pre-created 3D skeleton library. Aligning all the skeletons was a challenging problem in this project. We used the human-object interaction relationship and temporal information of the trajectories, such as motion information, trying to put the skeletons in the correct place. The results showed that visualization was a good way to express the understanding results of a computer vision system.

The thesis of Chenhui Wang is approved.

Nicolas Christou

YingNian Wu

Song-Chun Zhu, Committee Chair

University of California, Los Angeles

2014

To my mother . . .

who—among so many other things—

saw to it that I learned to touch-type

while I was still in elementary school

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank my advisor, Song-Chun Zhu, for the help of giving very insightful guidance through all my work. He was patient with me as I worked. Thanks also to my other committee members, Yingnian Wu and Nicolas Christou, for reading and commenting on it. Thanks to Glenda Jones, the department administrator, who helped through out the filing process. At last, I would like to thank my parents who have always been the greatest support of my life.

I would like to further thank the support of grant: ONR MURI N00014-10-1-0933, DARPA MSEE project FA 8650-11-1-7149, 973 Program 2012CB316402

CHAPTER 1

Introduction

Modern computer vision has shown more and more favor on the spacial relationship between people and surrounding environment. However, the information that we can get from 2D images is always very vague and the 2D trajectories from tracking tell nothing in this area. Reconstructing the 3D environment and visualizing the 3D movements and interaction between people and man-made objects is very useful. In addition to expressing the learning results of a computer vision system, visualization is also very important, especially in revealing the relationship, such as *sitting on a chair, standing on the ground*.

1.1 Motivation

Typical 2D computer vision methods can largely solve detection and tracking problems, however, when dealing with occlusion and human-object interactions, such methods often lose their efficiency. The goal of this research is to find out the concealed information under 2D representation and make the detection and tracking results meaningful and reliable. Moreover, this visualization will further help to solve problems, such as *reasoning and intention prediction*.

Dataset. In order to test the performance of DARPA project results, we used the data collected by SIG, which contained both indoor and outdoor human daily scene. The complexity of these typical scenes can largely represent normal human daily activities and the interactions between each other and man-made

objects. The dataset contains 3 categories, 8 scenes and 26 different viewpoints in total. What is more, there are also 14 action classes and 6 interacting object classes. We test our visualizing method on this dataset and the visualization results demonstrate the goodness of the DARPA system.

CHAPTER 2

3D Environment Reconstruction

Much work has been done in camera calibration, since it is a very important step in 3D computer vision. The techniques can roughly be classified into two categories: photogrammetric calibration and self-calibration. To begin this chapter, we would like to first introduce the notations used.

2.1 Notation

We dente a 2D point by $\mathbf{m} = [u, v]^T$, and a 3D point by $M = [X, Y, Z]^T$. $\tilde{x}$ is used to denote the augmented vector by adding 1 as the last element: $\tilde{m} = [u, v, 1]^T$ and $\tilde{M} = [X, Y, Z, 1]^T$. A camera is modeled by the usual pinhole: the relationship between a 3D point M and its image projection $\mathbf{m}$ is given by

$$s\tilde{\mathbf{m}} = \mathbf{A} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \tilde{M}, \tag{2.1}$$

where s is an arbitrary scale factor. $(\mathbf{R}, \mathbf{t})$, called the extrinsic parameters, is the rotation and translation which relates the world coordinate system to the camera coordinate system, and $\mathbf{A}$, called the camera intrinsic matrix, is given by

$$\mathbf{A} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

with $[u_0, v_0]$ the coordinates of the principal point, α, and β the scale factors in image u and v axes, and γ the parameter describing the skewness of the two image axes.

2.2 Homography between the model plane and its image

Without loss of generality, we assume the model plane is on $Z = 0$ of the world coordinate system. Let us denote the i^{th} column of the rotation matrix $\mathbf{R}$ by $\mathbf{r}_i$. From (2.1), we have

$$s \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{A} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 & \mathbf{t} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix}$$
$$= \mathbf{A} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

Since Z is assumed equal to 0, we can abuse the notation and $M = [X, Y]^T$. In turn, $\tilde{M} = [X, Y, 1]^T$. Therefore, a model point M and its image $\mathbf{m}$ is related by a homography $\mathbf{H}$:

$$s\tilde{\mathbf{m}} = \mathbf{H}\tilde{M} \qquad \text{with} \qquad \mathbf{H} = A \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{t} \end{bmatrix}. \tag{2.2}$$

Thus, the 3×3 matrix $\mathbf{H}$ is defined up to a scale factor.

2.3 Estimation of the Homography Between the Model Plane and its Image

The technique we used to estimate the homography between the model plane and its image is based on maximum likelihood criterion. Let M_i and $\mathbf{m}_i$ be the model and image points, respectively. Ideally, they should satisfy (2.2). In practice, they do not because of noise in the extracted image points. Let us assume that $\mathbf{m}_i$ is corrupted by Gaussian noise with mean $\mathbf{0}$ and covariance matrix $\mathbf{\Lambda}_{\mathbf{m}_i}$. Then, the maximum likelihood estimation of $\mathbf{H}$ is obtained by minimizing the following

functional

$$\sum_i (\mathbf{m}_i - \hat{\mathbf{m}}_i)^T \Lambda_{\mathbf{m}_i}^{-1} (\mathbf{m}_i - \hat{\mathbf{m}}_i),$$

where

$$\hat{\mathbf{m}}_i = \frac{1}{\bar{\mathbf{h}}_3^T M_i} \begin{bmatrix} \bar{\mathbf{h}}_1^T M_i \\ \bar{\mathbf{h}}_2^T M_i \end{bmatrix} \qquad \text{with } \bar{\mathbf{h}}_i, \text{the } i^{\text{th}} \text{ row of } \mathbf{H}$$

In practice, we simply assume $\Lambda_{\mathbf{m}_i} = \sigma^2 \mathbf{I}$ for all i. This is reasonable if points are extracted independently with same procedure. In this case, the above problem becomes a nonlinear least-squares one, i.e., $\min_{\mathbf{H}} \sum_i \|\mathbf{m}_i - \hat{\mathbf{m}}_i\|^2$. The nonlinear minimization is conducted with the Levenberg-Marquardt Algorithm. This requires an initial guess, which can be obtained as follows.

Let $\mathbf{x} = [\bar{\mathbf{h}}_1^T, \bar{\mathbf{h}}_2^T, \bar{\mathbf{h}}_3^T]^T$. Then equation (2.2) can be rewritten as

$$\begin{bmatrix} \tilde{M}^T & \mathbf{0}^T & -u\tilde{M}^T \\ \mathbf{0}^T & \tilde{M}^T & -v\tilde{M}^T \end{bmatrix} \mathbf{x} = \mathbf{0}$$

When we are given n points, we have n above equations, which can be written in matrix equation as $\mathbf{L}\mathbf{x} = \mathbf{0}$, where $\mathbf{L}$ is a $2n \times 9$ matrix. As $\mathbf{x}$ is defined up to a scale factor, the solution is well known to be the right singular vector of $\mathbf{L}$ associated with the smallest singular value (or equivalently, the eigenvector of $\mathbf{L}^T\mathbf{L}$ associated with the smallest eigenvalues).

In $\mathbf{L}$, some elements are constant 1, some are in pixels, some are in world coordinates, and some are multiplication of both. This makes $\mathbf{L}$ poorly conditioned numerically. Much better results can be obtained by performing a simple data normalization prior to running the above procedure.

2.4 Constrains on the intrinsic parameters

Given an image of the model plane, an homography can be estimated. Let us denote it by $\mathbf{H} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \mathbf{h}_3 \end{bmatrix}$. From (2.2), we have

$$\begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_2 & \mathbf{h}_3 \end{bmatrix} = \lambda \mathbf{A} \begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & t \end{bmatrix}$$

where λ is an arbitrary scalar. Using the knowledge that $\mathbf{r}_1$ and $\mathbf{r}_2$ are orthonormal, we have

$$\mathbf{h}_1^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_2 = 0 \tag{2.3}$$

$$\mathbf{h}_1^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_1 = \mathbf{h}_2^T \mathbf{A}^{-T} \mathbf{A}^{-1} \mathbf{h}_2 \tag{2.4}$$

These are the two basic constraints on the intrinsic parameters, given one homography. Because a homography has 8 degrees of freedom and there are 6 extrinsic parameters (3 for rotation and 3 for translation), we can only obtain 2 constraints on the intrinsic parameters.

2.5 Real Data Calibration

Using the proposed technique to estimate each homography, we can simply reconstruct all the DARPA indoor and outdoor scenes. The following figures show the reconstruction results. To make the scene not so messy, we only draw one chair in each indoor scene. In real calibration data, we have all the chairs' 3D location information in our system.

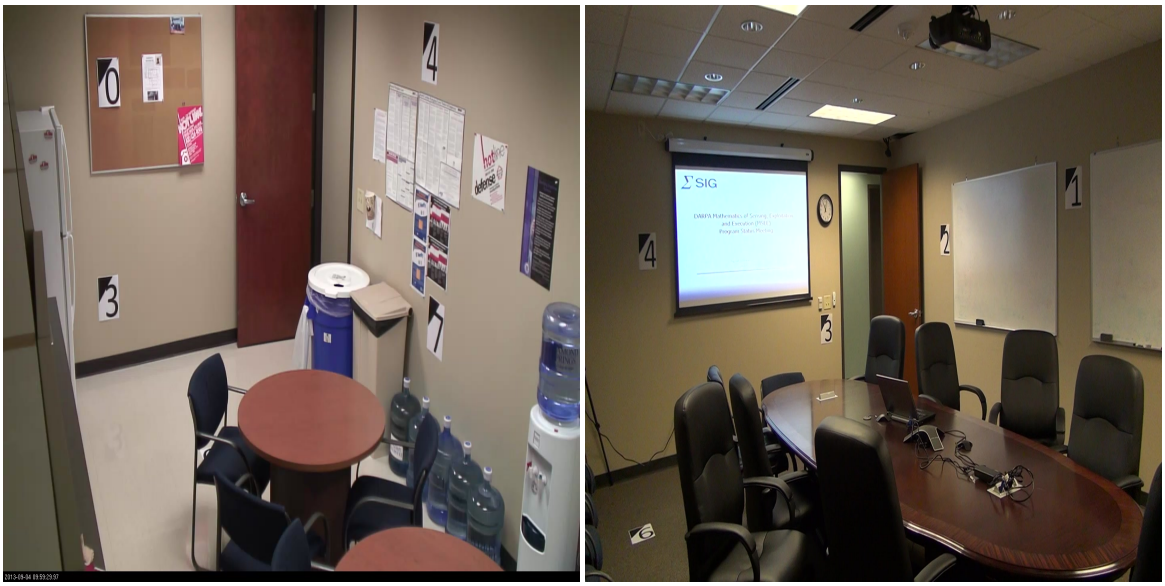

Figure 2.1: Two indoor scene

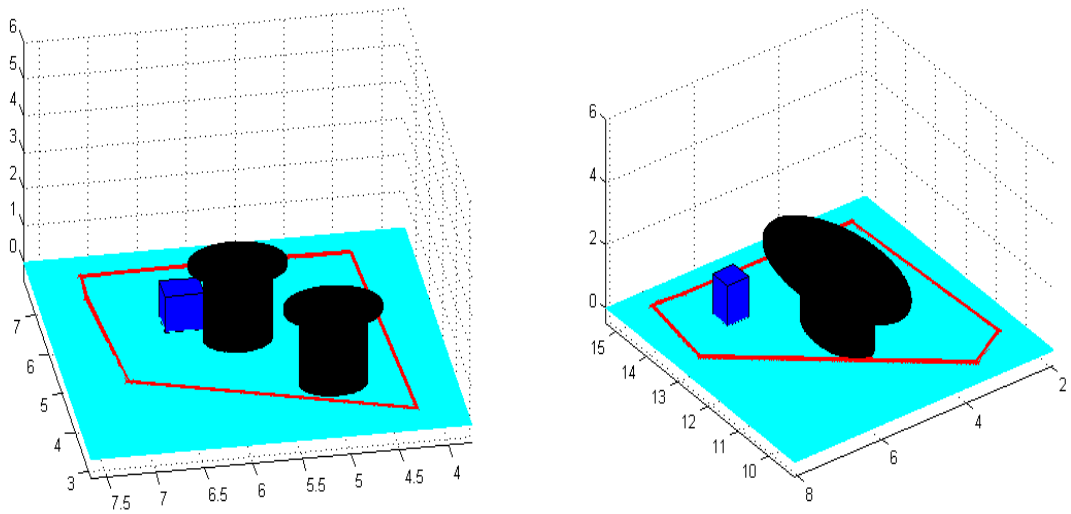

Figure 2.2: Two reconstructed scene respectively

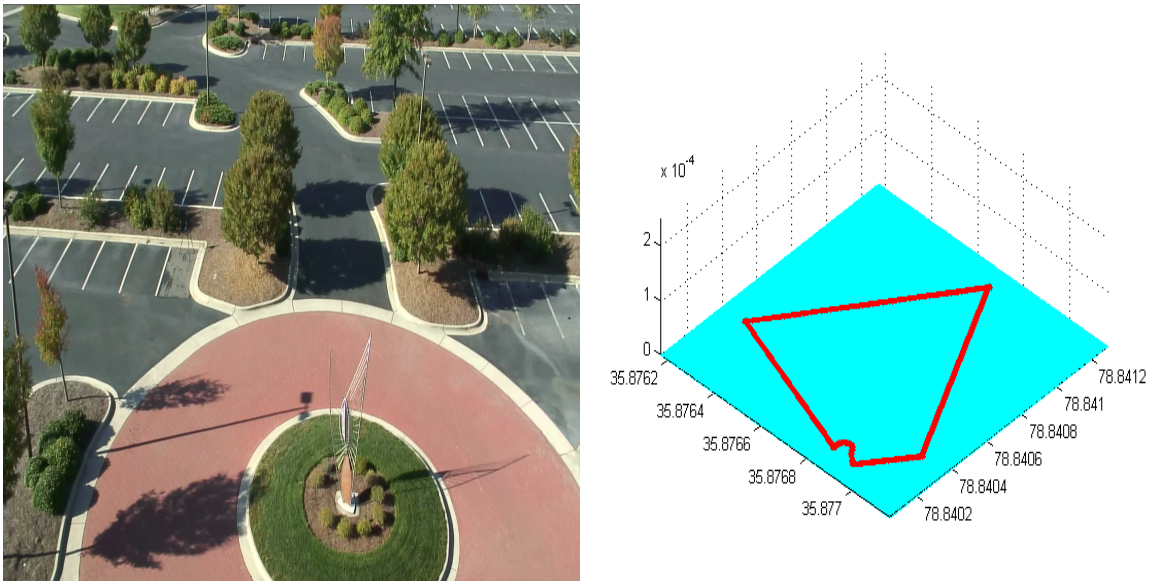

Figure 2.3: One outdoor scene and its reconstructed scene

CHAPTER 3

3D Skeleton Library

In order to visualize all the people actions, we built a 3D human action skeleton library using Kinect camera, and all the skeletons are captured from RGBD video. It includes 2 categories, 14 action classes, including stationary actions, such as *standing, sitting*, and non-stationary actions, like *walking, throwing*. It has totally 937 action video sequences and 50,132 RGBD frames. Each action class contains about 30 video sequences in stationary action classes, and around 80 in moving action classes. These action videos were shot from three different viewpoints, hence when we are dealing with skeleton visualization in a complicated scene, an appropriate skeleton could be placed in the right place.

The human skeleton model shot by Kinect camera each have 20 3D joints as shown in Table 3.1. After shooting the video data, we perform normalization by aligning all the skeletons to a reference pose so that the torso and shoulder of all skeleton models have the same location, size and direction. The alignment makes all the skeleton poses stand at $(0, 0, 0)$ with roughly the same size $3 \times 1 \times 10$, facing the direction of $(1, 0, 0)$.

3.1 Stationary Action Skeleton

Stationary actions mainly contains standing and sitting. These two classes will be found frequently in human daily action events. Standing is usually a transition status, it connects two walking paths in most cases. One random selected 3D

Table 3.1: Action intrinsic parameters

Number	Joint	Number	Joint
1	hip center	11	wrist left
2	spine	12	hand left
3	shoulder center	13	hip right
4	head	14	knee right
5	shoulder right	15	ankle right
6	elbow right	16	foot right
7	wrist right	17	hip left
8	hand right	18	knee left
9	should left	19	ankle left
10	elbow left	20	foot leftt

skeleton model of action *standing* is shown in Figure 3.1.

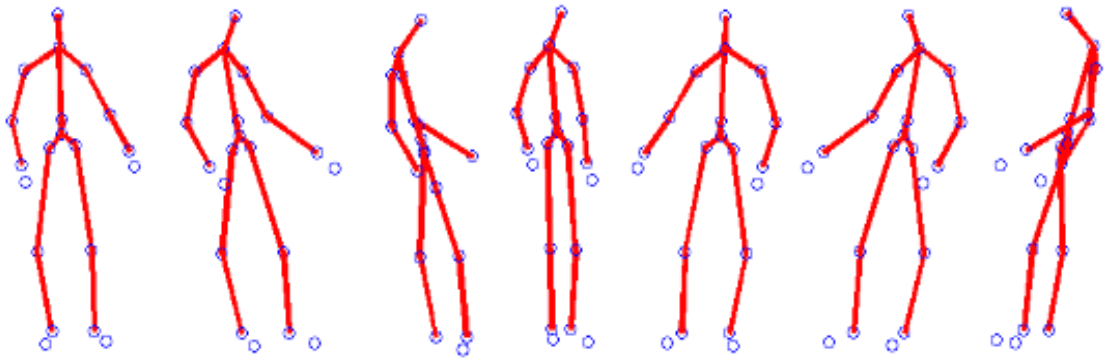

Figure 3.1: 3D skeleton for standing

Sitting is another stationary action that can be mostly used in the visualization of indoor environments. Since chair is one of the most important components of a indoor scene, the action of sitting could be found through out the whole DARPA project data. People sat down and talked, played cards, or ate food. Figure 3.2 shows a typical sitting action 3D model in 3D space.

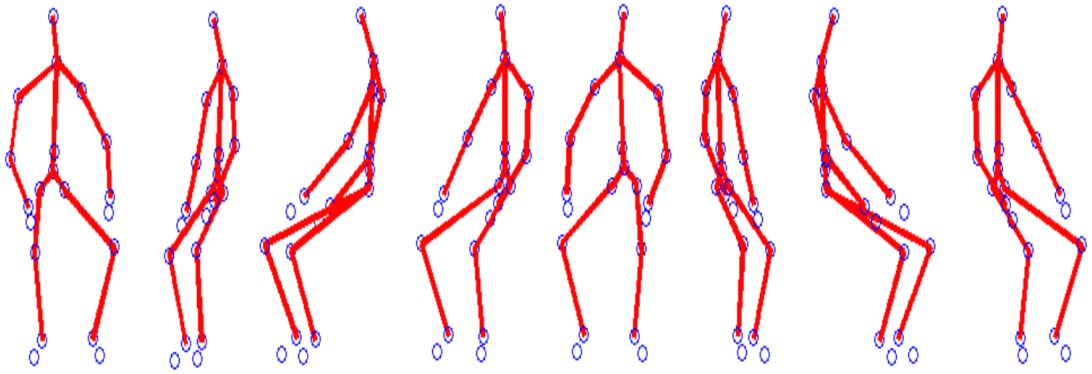

Figure 3.2: 3D skeleton for sitting

3.2 Non-stationary Action Skeleton

Moving actions form most of human daily action events. For example, walking can be everywhere in both indoor and outdoor scenes. People need to walk to do different sort of things, such as *picking up and throwing something*.

Here, we represent a non-stationary action by a sequence of 3D skeleton models. This kind of representation contains the temporal information of the underlying action, since in real world, people preform these actions in a continuous way, not a discrete way.

When dealing with outdoor actions, such as riding a bike, our library not only contains the model of the moving skeleton itself, but also the 3D bounding box of the interactive object, say *bicycle*. In this way, further information about this action can be learnt by our system. Because when the system detects a moving bicycle, it will search our library and get the corresponding 3D model. The model will provide the knowledge of riding a bike, like *human on the bike, feet above the ground*.

In our library, we define several such relationship, like *people in a car, people walking a bicycle*. Similar human-object interactive actions such as *throwing, picking*, also contain the object 3D bounding box alongside with human skeleton

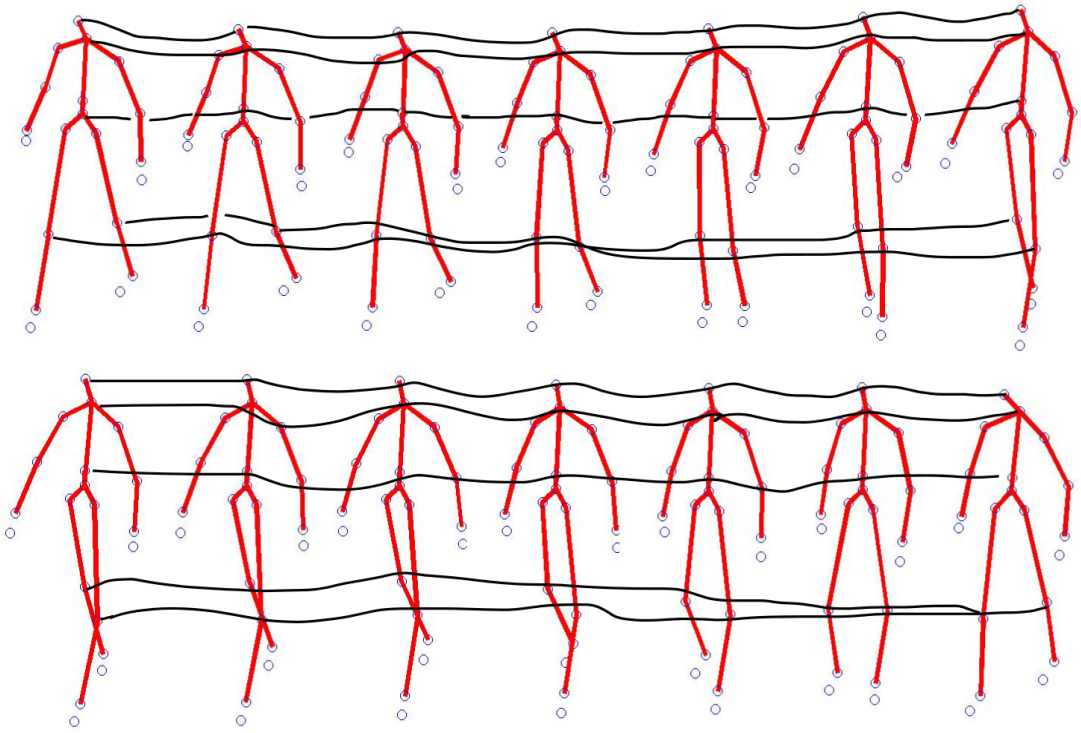

Figure 3.3: 3D skeleton sequence for walking

model.

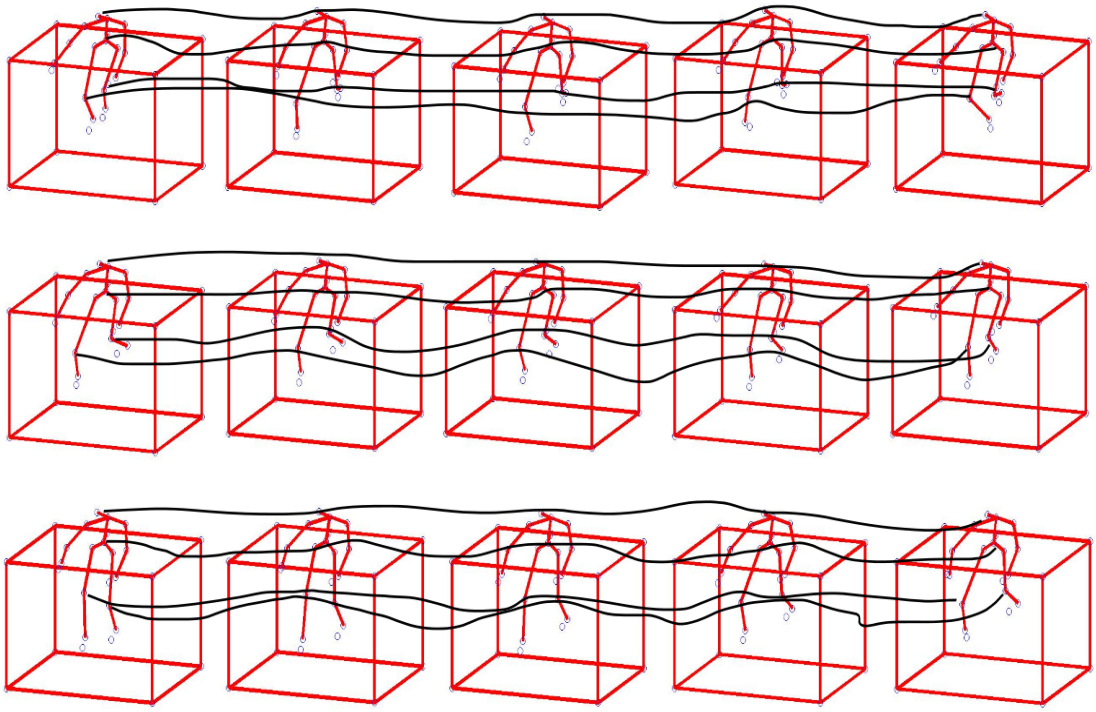

Figure 3.4: 3D skeleton sequence for riding

3.3 Information Provide by Our Library

As mentioned in 3.2, our system can get information about the human-object interaction from this 3D skeleton library, such as *on the bike, feet above the ground*. In out model, we define these human-object relationships as built in model labels. Walking, for example, would provide the intrinsic model label of feet on ground. The intrinsic model label for throwing would be the ownership of a small object changing from "yes" to "no". Each 3D model is associated with at least one reasonable model label.

With the help of these information, our system would be able to answer questions such as "Is there a person with his feet above the ground?" or "How many

people are there throwing a disk (small object)?"

Table 3.2: Action intrinsic parameters

	Intrinsic label				
Action	feet on ground	on chair	on bike	in car	ownership
carrying	1	0	0	0	$0 \rightarrow 1$
crawling	1	0	0	0	0
doffing	1	0	0	0	$1 \rightarrow 0$
donning	1	0	0	0	$0 \rightarrow 1$
in car	0	0	0	1	0
picking	1	0	0	0	$0 \rightarrow 1$
riding	0	0	1	0	0
sitting	1	1	0	0	0
sit down	1	$0 \rightarrow 1$	0	0	0
standing	1	0	0	0	0
stand up	1	$1 \rightarrow 0$	0	0	0
throwing	1	0	0	0	$1 \rightarrow 0$
walking	1	0	0	0	0
walking bike	1	0	0	0	0

Our library also describes the variation of key parts of human performing different actions. We assume that each joint follows a multivariate Gaussian distribution in space. Figure 3.5 shows the exact distribution of hip center (left) and head (right) in space. We can tell that the same human joint will not vary a lot during one action. More empirical distributions are shown in the following page.

Further, our library can generate scatter plot of the 2D locations of the moving joints given the locations of the still joints. Figure 3.6 shows what our library will produce. In the action of standing up, if we set the two ankles fixed, the variation

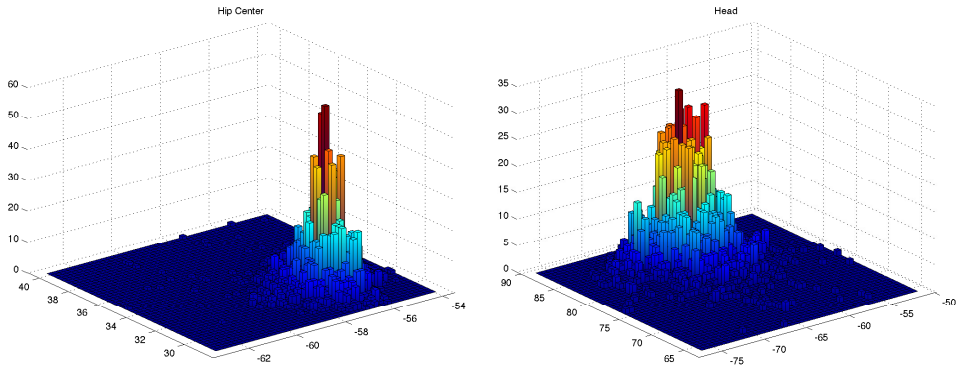

Figure 3.5: Empirical distribution of joints in action of sit down

of head location is the largest and two shoulders' are smaller, while hip center has the smallest variation. This observation fits our intuition.

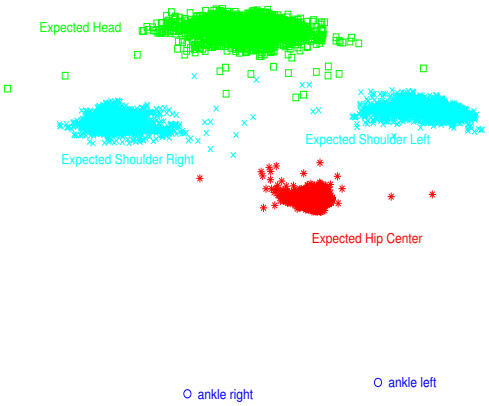

Figure 3.6: Relative 2D location map of standing up

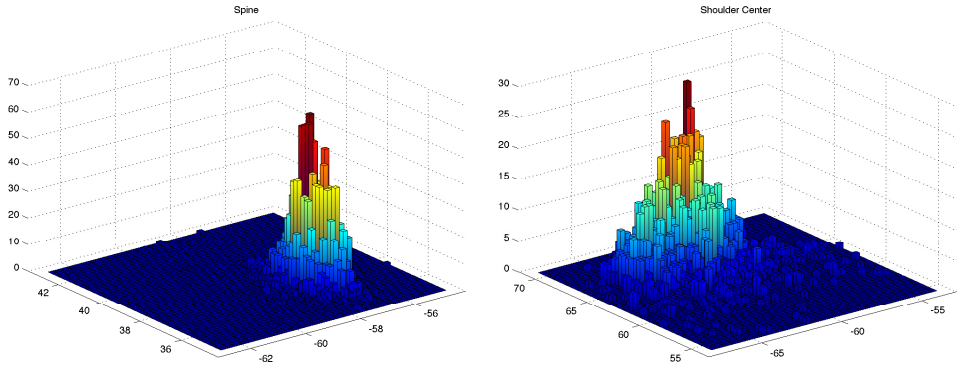

Figure 3.7: Empirical distribution of joints in action of sit down

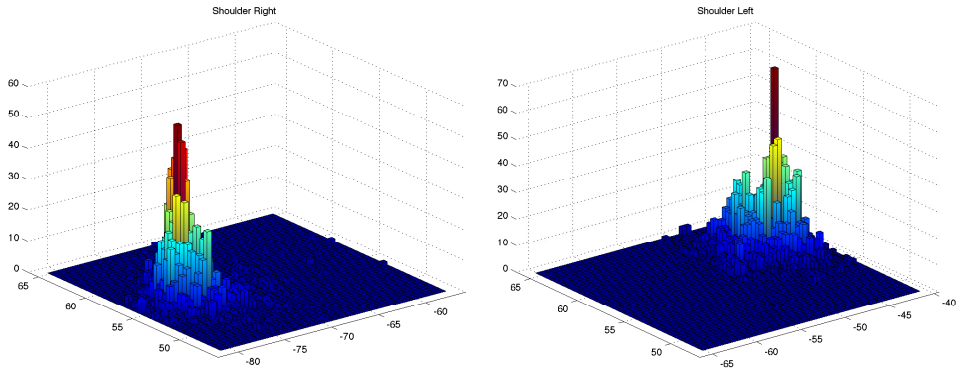

Figure 3.8: Empirical distribution of joints in action of sit down

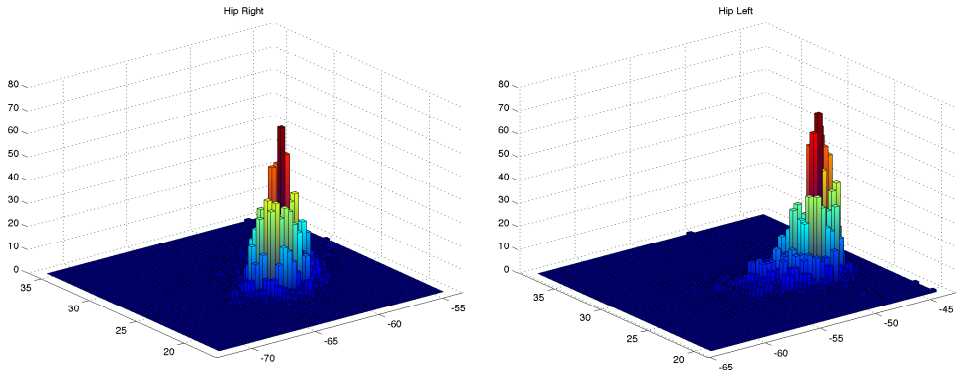

Figure 3.9: Empirical distribution of joints in action of sit down

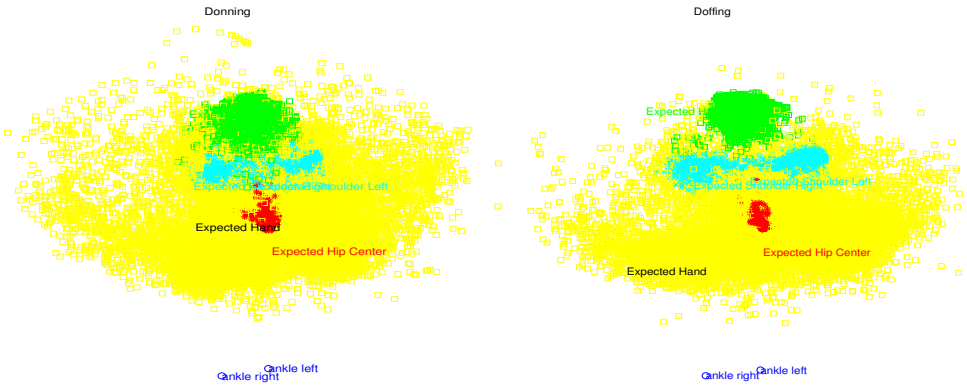

Figure 3.10: Relative 2D location map

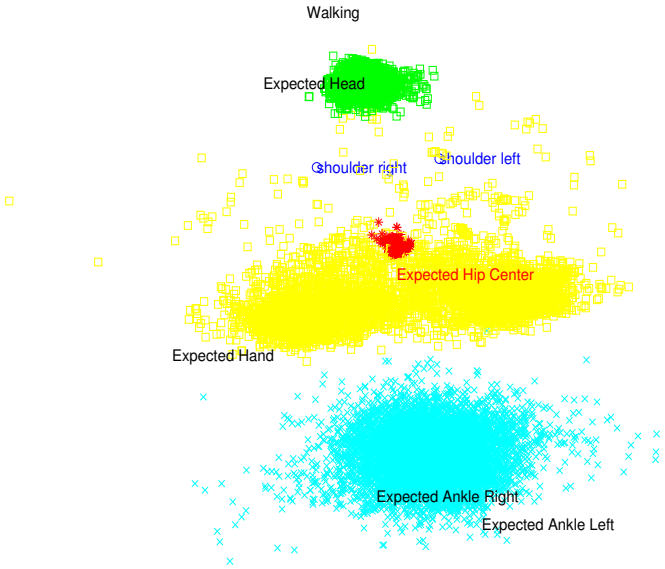

Figure 3.11: Relative 2D location map

CHAPTER 4

Visualization

In this chapter, we perform the final visualization of this project. First, we introduce the 3D trajectories used in visualizing the movements of person. Then, the method for aligning the skeletons is addressed, hence skeletons will not be facing in a wrong direction or seen at a unrealistic position. At last, visualization of the whole scene both skeletons and the surrounding environment is used to show the goodness and effectiveness of our method.

4.1 3D Trajectories

The 3D trajectories we used here are pre-computed through 2D tracking. However, these trajectories are pretty noisy, which are resulted in different size of head bounding box, occlusion, etc. Thus smoothing is needed.

We tried two ways of smoothing.

1). Simple moving average:

$$P_t = \frac{1}{n} \sum_{i=t}^{t+n} P_i \tag{4.1}$$

where $P_t = (x_t, y_t, z_t)$ is the trajectory position at time t. This method is easy and fast to implement, but it lags the actual position;

2). Smoothing spline:

Let $(x_i, Y_i); x_1 < x_2 < \cdots < x_n$ be a sequence of observations, modeled by

the relation $Y_i = \mu(x_i)$. The smoothing spline estimate $\hat{\mu}$ of the function μ is defined to be the minimizer of

$$\hat{\mu} = \min \left(\sum_{i=1}^{n} \left((Y_i - \hat{(}\mu(x_i)) \right)^2 + \lambda \int_{x_i}^{x^n} \hat{\mu}''(x)^2 \mathrm{d}x \right) \tag{4.2}$$

This method will manually balance accuracy and smoothness, but the choosing of smoothing parameter λ is a challenge.

Combing both methods, we get some good smoothed 3D trajectories results. Figure 4.1 and Figure 4.2 show the difference before and after the smoothing of 3D trajectories.

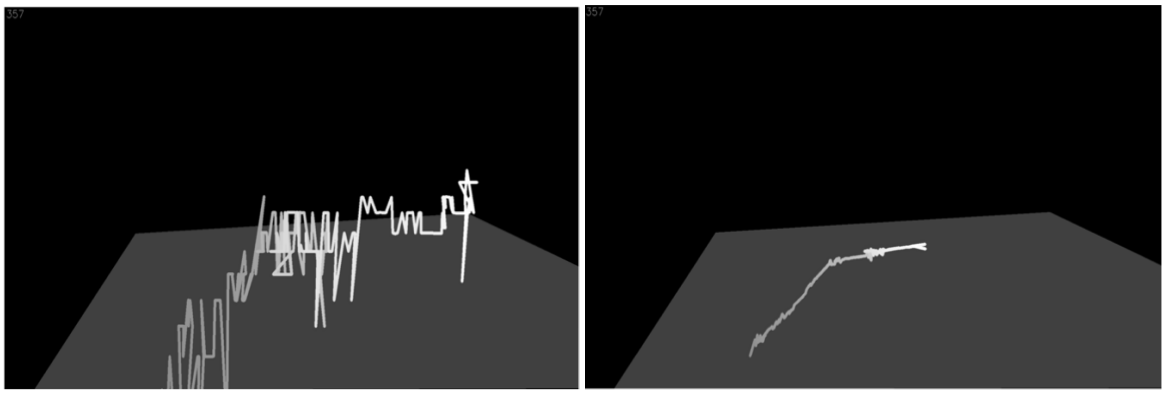

Figure 4.1: Three different viewpoints of break room

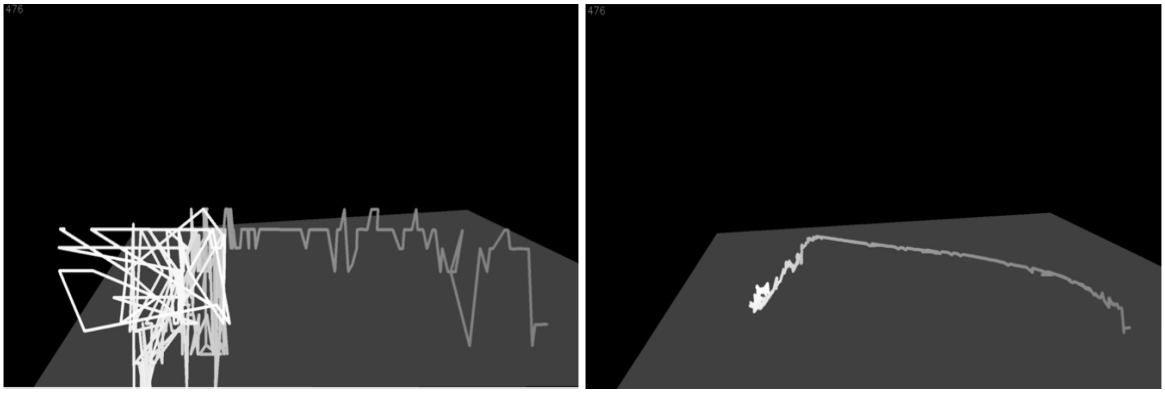

Figure 4.2: Three different viewpoints of break room

4.2 Aligning the Skeletons

Since we live in a 3D spacial world, the relationship between human and surrounding objects is especially important in the visualization of a space. The spacial constraints is the first thing that we considered in this project.

Suppose $V = (I_1, \ldots, I_\tau)$ is an input video sequence in the time interval $[1, \tau]$, where I_t is the video frame at time t. Let $L = (l_1, \ldots, l_\tau)$ be the sequence of frame labels. $l_t = (h_t, j_t, o_t, a_t)$ is the interpretation to the frame I_t. h_t is the human pose. j_t is the trajectory location. $o_t = (o_t^1, \ldots, o_t^{n^t})$ are the objects interacting with human, where n_t is the number of objects. Each object includes the attributes of class label and 3D location. a_t is the human action class like *walking, sitting, etc.*

Hence, the human-object interactions in 3D spacial domain can be described as

$$\Phi(I_t, l_t) = \phi_1(a_t, h_t, \beta_t) + \phi_2(a_t, j_t, o_t) \tag{4.3}$$

This function shows the geometric compatibility of each frame which describes the spatial constraint between human body and objects in 3D space.

Pose Model $\phi_1(a_t, h_t, \beta_t)$ is the human pose model. The human pose with 20 3D joints are from our 3D skeleton library. To normalize the data, we align all the skeletons to a reference pose so that the torso and shoulder of all poses have the same location, size and direction. To get the correct moving direction, we again used the simple moving average smoothing algorithm to catch the temporal information. Suppose we calculate the motion using only the footprint data, $z = 0$, thus $j_t = (x_t, y_t, 0)$, then

$$\beta_t = \frac{1}{n} \sum_{i=t}^{t+n} \arctan \frac{\Delta x_i}{\Delta y_i} \tag{4.4}$$

Further, as mentioned in 3.3, we assume that h_t follows a multivariate Gaussian distribution. Then we have $N(h_t; \mu_{a_t}, \Sigma_{a_t})$, where μ_{a_t} is the mean and Σ_{a_t} is the

covariance, thus each action will have a corresponding likelihood score.

3D Geometric Compatibility $\phi_2(a_t, j_t, o_t)$ measures the human-object geometric relations. Since human-object relationship in 3D space contains more information, the geometric relation in 2D image is not applicable as shown in Figure 4.3. We model this relationship in 3D space.

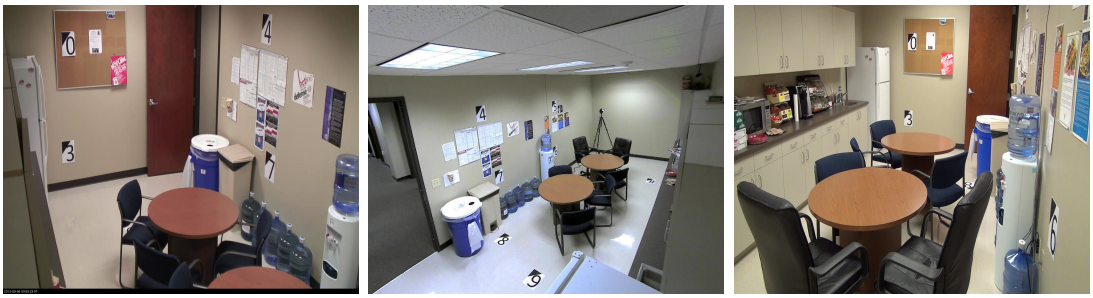

Figure 4.3: Three different viewpoints of break room

In any human-object interaction relationship, the location of an object is closely related to the locations and directions of some body parts, which we call the key parts. As when people sit, he must sit on a chair, hence his geometric location must be the same as the chair's. Suppose $\eta_{o_t^i}$ is the difference vector from the key parts center to the object geometric location center. In the visualization part we would like to minimize this distance. The 3D geometric relation is modeled as:

$$\phi_2(a_t, j_t, o_t) = \frac{1}{n_t} \sum_{j=1}^{n_t} \min_i \eta_{o_t^i} \tag{4.5}$$

where n_t is the number of objects. Dividing the function by n_t is to offset the influence of difference object number.

This minimum distance will also give the table facing information, since the minimum tells which table is the nearest one. Hence the facing problem can be easily solved.

4.3 Final Visualization

Combining all the above techniques, we can recover the 3D environment and visualize the movement of human and the interactive objects. Figure and Figure show the final visualization results.

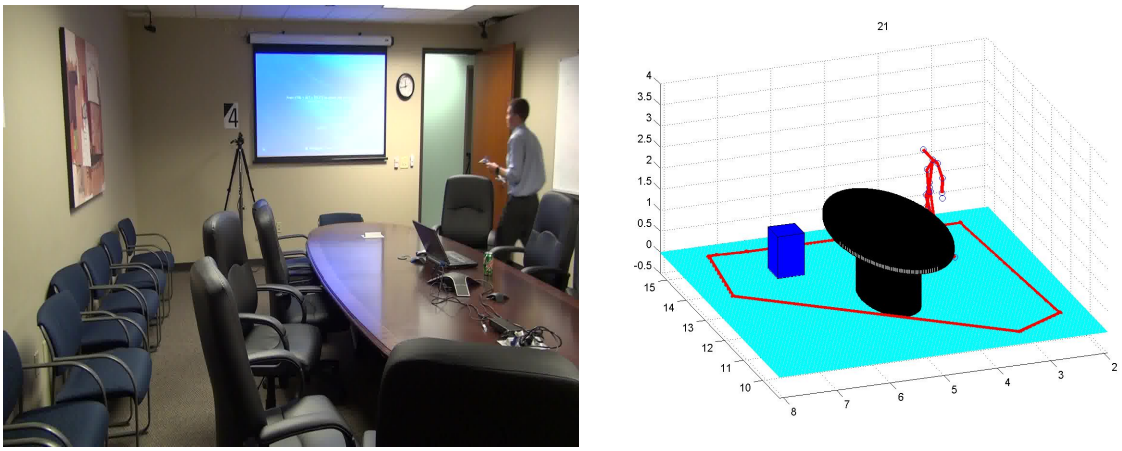

Figure 4.4: Final visualization of conference room: Walking

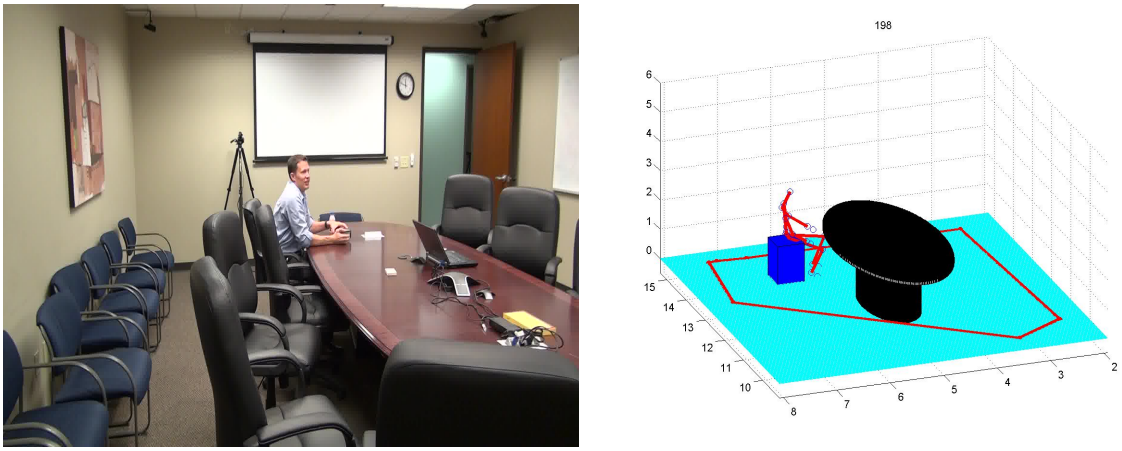

Figure 4.5: Final visualization of conference room: Sitting

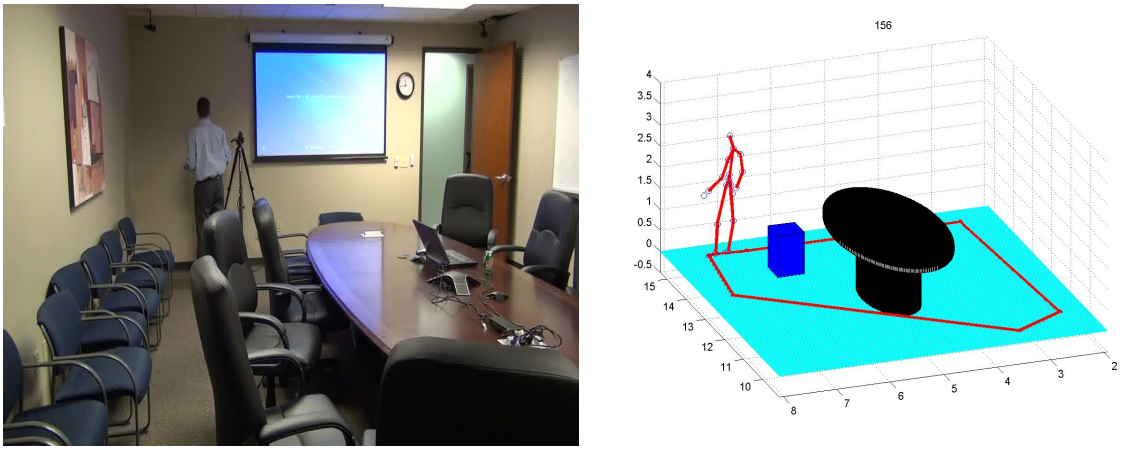

Figure 4.6: Final visualization of conference room: Standing

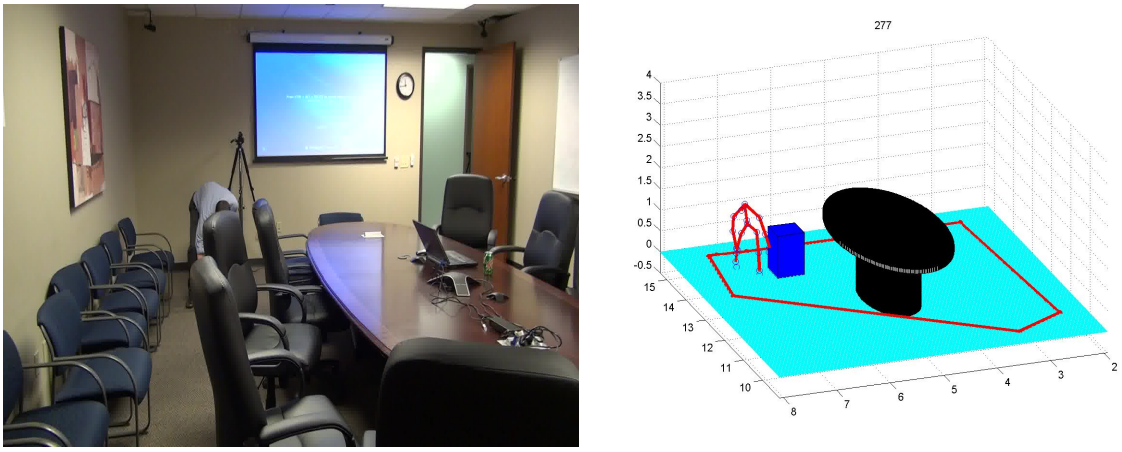

Figure 4.7: Final visualization of conference room: Picking

CHAPTER 5

Conclusion

We conducted a dynamic homographic transformation method to calibrate the scene and reconstruct the 3D environment of DARPA project. After the reconstruction, we visualized the 3D skeleton action movements in the synthesized scenes. The human-object interactions defined in 3D spacial domain boost the reliability on visualization of 3D events. Through these visualization experiments on DARPA dataset, we can prove the effectiveness of our method. Further using the intrinsic information we provided, the system would be able to answer simple queries.

References

[1] Z. Zhang. A Flexible New Technique for Camera Calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 22(11): 1330-1334, 2000.

[2] P. Wei, Y. Zhao, N. Zheng, and S.-C. Zhu Modeling 4D Human-Object Interactions for Event and Object Recognition. *International Conference on Computer Vision (ICCV)*, 2013.

[3] H. S. Koppula, R. Gupta, and A. Saxena. Learning human activities and object affordances from rgb-d videos. *The International Journal of Robotics Research*, 32(8), 2013.

[4] A. Gupta, A. Kembhavi, and L. Davis. Observing human-oject interactions: Using spatial and functional compatibility for recognition. *IEEE TPAMI*, 31(10), 2009.

[5] Y. Zhao and S.-C. Zhu Scene parsing by integrating function, geometry and appearance models. In *CVPR*, 2013.

[6] B. Yao and L. Fei-Fei. Recognizing human-object interactions in still images by modeling the mutual context of objects and human poses. *TPAMI*, 34(9), 2012.

[7] J.Gall, A. Fossat, and L. van Gool. Functional categorization of objects using real-time markerless motion capture. In *CVPR*, 2011.

[8] M.Pei, Y.Jia, and S.-C. Zhu. Parsing video events with goal inference and intent prediction. In *ICCV*, 2011.

[9] S. Sadanand and J.J. Corso. Action bank: A high-level representation of activity in video. In *CVPR*, 2012.

[10] J. Sung, C. Ponce, B. Selman, and A. Saxena. Unstructured human activity detection from rgbd images. In *ICRA*, 2012.

[11] M. Marszalek, I. Laptev, and C. Schmid. Actions in context. In *CVPR*, 2009.

[12] B. Packer, K. Saenko, and D. Koller. A combined pose, object, and feature model for action understanding. In *CVPR*, 2012.

[13] L. Bourdev, J. Malik. Poselets: Body Part Detectors Trained Using 3D Human Pose Annotations. In *ICCV*, 2009