Upper Body Motion Analysis Using Kinect for Stroke Rehabilitation at the Home

by

Tingfang Du

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved November 2012 by the
Graduate Supervisory Committee:

Pavan Turaga, Chair
Andreas Spanias
Thanassis Rikakis

ARIZONA STATE UNIVERSITY

December 2012

ABSTRACT

Motion capture using cost-effective sensing technology is challenging and the huge success of Microsoft Kinect has been attracting researchers to uncover the potential of using this technology into computer vision applications. In this thesis, an upper-body motion analysis in a home-based system for stroke rehabilitation using novel RGB-D camera – Kinect is presented. We address this problem by first conducting a systematic analysis of the usability of Kinect for motion analysis in stroke rehabilitation. Then a hybrid upper body tracking approach is proposed which combines off-the-shelf skeleton tracking with a novel depth-fused mean shift tracking method. We proposed several kinematic features reliably extracted from the proposed inexpensive and portable motion capture system and classifiers that correlate torso movement to clinical measures of unimpaired and impaired. Experiment results show that the proposed sensing and analysis works reliably on measuring torso movement quality and is promising for end-point tracking. The system is currently being deployed for large-scale evaluations.

ACKNOWLEDGEMENTS

I would like to thank those who have contributed to my education and life at Arizona State University during the past two years.

First and foremost, I would like to thank my advisor, Pavan Turaga, for your support and guide thorough the past one year. Pavan gave me so much freedom to explore and discover my own interest. His approach to problem formulation and presentation has greatly shaped my way of thinking.

I would like to thank my application advisor, Thanassis Rikakis. Thank you so much for providing me an amazing opportunity to work in the stroke rehab group in AME. You are a person full of personal charm. Your sharp insights has inspired me a lot. I would also like to thank Andreas Spanias for kindly agreeing to serve on my thesis committee.

I would also like to thank my former advisor, Yinpeng Chen. I've learnt a lot from you, especially on your research attitude and approach to solve problems. You are a mentor both in research and life. I would also like to thank other members of the home system group for making my stay in the group fun and memorable - Michael Baran, Nicole Lehrer, Long Cheng, Diana Siwiak, Margaret Duff, Todd Ingalls, Loren Olsen, Rajarem Singaravelu.

Many thanks to my friends who shared my life. Without them my graduate study experience would not be as pleasant and colorful - Hui Zou, Yang Yang, Fengyu Wang, Yujia Zhu, Ming Zhou.

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION

1.1 THE RESEARCH PROBLEM

Every year, about 795,000 people in the United States suffer from stroke [40], and about 60% of stroke patients experience minor to severe upper extremity motor deficits, resulting in a decline quality of post-stroke life [41]. Stroke rehabilitation is the process that helps stroke survivors return to normal life as much as possible by regaining and relearning the skills of everyday living, which lasts from immediately after stroke to over a year. Physical therapy (PT) is one of the important aspects of stroke rehabilitation which focuses on regaining motor functionality by performing exercises and relearning functional tasks [26].

Conventional rehabilitation train motor function using labor-intensive (therapist) and expensive facilities. It is dependent on patient compliance and also suffers from limited availability depending on geography [42]. Further, clinical intervention alone is not effective for activities at a home [28][49][18][19]. Virtual reality (VR) is a computer-based technology that allows users to interact with a multisensory simulated environment and receive "real-time" feedback on performance [42]. Compared to the conventional rehabilitation, VR rehabilitation applied relevant concepts based on neuroplasticity leading to benefits in motor function improvement [42]. Also, it can be tailored to the needs of the patient, by providing feedback that fits the individual's cognitive and physical impairments, in order to promote positive learning experience while being fun and motivating [25].

VR has been widely applied in designing novel rehab systems for physical therapy. Based on the types of VR systems [17], they can be divided into two groups: 1) immersive VR rehab systems [21][56]; 2) nonimmersive VR rehab systems [41][34]. Research has shown that the use of VR systems may have improved

motor function, although the results is still not universally accepted, it is worthy and promising to further develop the VR-based stroke rehabilitation. In order for therapy to be effective, there is a need for tools that a patient can take home after they leave the clinic [5][46]. In recent years, there has been increasing interest to devise mixed-modal interventions that can assist a person at their home [46][6][14], for encouraging reflection on one's movement with the goal of supplementing traditional therapy.

An adaptive mixed reality rehabilitation (AMRR) [13][20] and motor learning theories [43][53] with motion capture and activity analysis technologies, and multimedia feedback, can result in effective and portable rehabilitation systems to be deployed at one's home. Over the past a few years, an ASU research team has investigated the benefits of an AMRR system, and shown its efficacy in helping improve the kinematic and functional performance of upper extremity [20]. However, this system was designed for a clinical setting, with high-end motion capture technologies with various markers and rigid-bodies attached to the wrist, arm, shoulder, torso etc, resulting in very rich data about the activities. However, this marker-based solution is unrealistic in a home-based environment. First, the heavy duty camera system and the complexity of marker setup inhibits participants to start up a physical session daily without assistance. Second, AMRR is not affordable to at home therapy. Third, the long-term at home therapy aims to transfer the training and assessment of clinician-led therapy sessions into daily experience at home. Thus, less constrained physical tasks and a multi-layered feedback hierarchy call for a simpler motion capture, which makes the old marker-based solution cumbersome. Therefore, a low-cost motion capture system should be employed in the home-based system. What low-cost sensing devices could be served as an ideal solution for our application? Is the low-cost motion capture module reliable enough for the motion analysis of impaired patients?

1.2 CHALLENGES AND MOTIVATION

A number of pressing challenges are yet to be addressed for designing low-cost motion capture module of home-based stroke rehab system. First, patients are expected to run the whole session of tasks unassisted. As a result, the system should be easy to setup, and user friendly. In particular, it is unrealistic to rely heavily on a marker-based solution due to their cumbersomeness. As well, inaccurate placement of markers can negatively effect the activity analysis modules. Second, in stroke rehab, many calculations of kinematic features requires a high tracking accuracy and high sampling rates. For example, computing deviation from expected speed profiles requires higher accuracy because speed is more sensitive to tracking errors than the trajectories. Also, the relative low sample rates (20-30Hz) is hard to provide a very detailed representation of movements. Third, the reduction of data requires a remodeling motion analysis in terms of proper kinematic representation and evaluation.

The recent advent of low-cost motion capture systems such as the Microsoft Kinect [1] emerge as excellent solutions. Kinect is a motion sensing device by Microsoft which enables hands-free control by tracking and interpreting user's body movement in three dimension using an infrared projector and RGB camera [1]. It enables a 3D presentation of the object, as well as off-the-shelf skeleton tracking algorithm, which greatly facilitates tracking of human body movement. Also, its low-cost, easy to use and natural human computer interaction leads us to throw the discussion on whether and how it could be applied as an effective solution for our at home system.

1.3 CONTRIBUTIONS

There are three key contributions in this thesis.

1. We conduct a comprehensive analysis of Kinect sensing technology and then discuss the usability of the Kinect in the stroke rehabilitation system. Also, we show how different sensing components are integrated in order to provide reliable and accurate data for further motion analysis.

2. Discuss the effect of accuracy for endpoint (wrist), torso and arm tracking during reaching physical tasks. A hybrid tracking approach using RGB-D camera is presented. We explicitly explain how we use the benefit of off-the-shelf skeleton tracking algorithm for torso and arm tracking. Also, an depth-fused mean shift tracking approach is described for endpoint tracking.

3. Propose an approach on evaluating torso compensatory movement quality on torso and endpoint (wrist) kinematic function during long-term therapy. The proposed evaluation framework evaluation of torso movement quality on reaching tasks or the progress of a certain session, and also provides quantitative measures for long-term therapy adaptation.

1.4 ORGANIZATION

The rest of the thesis is organized as follows: In chapter 2, related work and theoretical background are introduced. In chapter 3, we present the design of multi sensory motion capture system and present a novel upper body tracking approach using Kinect. In chapter 4, we introduce torso motion analysis for quantitative kinematic evaluation. We present system implementation and experimental results in Chapter 5, and concluding remarks in chapter 6.

Chapter 2

BACKGROUND AND RELATED WORK

2.1 HOME-BASED ADAPTIVE MIXED REALITY SYSTEM

Home-based adaptive mixed reality system (HAMRR) [6], integrates rehabilitation and motor learning theories, motion capture and activity analysis technologies, and multimedia feedback. HAMRR aims to provide a purposeful, engaging, hybrid (visual, auditory and physical) scene, which encourages patients to improve their performance on constraint induced repetitive tasks in stroke rehabilitation and promote learning of generalized movement strategies [12]. The system uses low-cost multimodal sensing components to track patients' upper body movement and provides a dynamic feedback environment to help stroke survivors self-assess their movement and improve the motor function in long-term adaptive task-specific therapy at home [6]. Below, we provide a brief introduction of HAMRR.

The HAMRR system integrates five computational subsystems: (a) multimodal Sensing; (b) motion analysis; (c) multimodal feedback; (d) archiving; and (e) adaptation. All these five subsystems are controlled by a media center computer. Figure 2.1 shows the system structure.

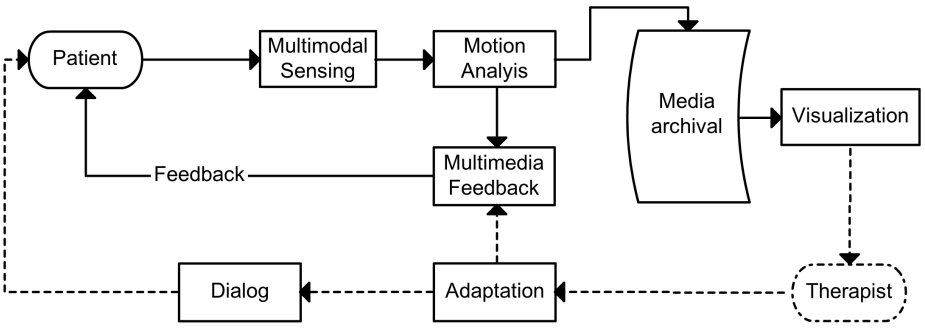


Figure 2.1: HAMRR System Architecture.

2.1.1 SENSING AND MOTION ANALYSIS

The HAMRR system utilizes multiple sensing modalities to extract kinematic features of a participant's movement, providing a cost-effective and robust sensing solution for unsupervised, private home training [6]. The physical environment includes a chair, a table, tangible objects, a 27-inch imac, two Bose speakers, and four Natural Point Opti-Track cameras. The sensing module includes:

- Opti-track camera system: the Opti-Track cameras, which run at 100fps, and tracking tools, which are used to track participants' joint 3D positions by tracking the reflective markers mounted on participants' body. The current Opti-track system is a scaled-down tracking solution as compared to the high-end camera system in our system at hospital.
- Chair: The chair is used in torso movement tracking.
- Tangible objects: The object interaction is sensed throughout different sensors setup on different objects, such as button object (used in reach-to-touch tasks), cone object (used in reach-to-grasp tasks), and lift object (used in reach-to-lift-to-transport tasks).

The motion analysis module calculates kinematic features based on the tracking data from sensing module. The kinematic features are used to train a variety of classifiers that evaluate the movement quality and then gives descriptive results to generate multimedia feedback, and also sent to adaptation framework for the selection of future tasks.

2.1.2 MULTIMEDIA FEEDBACK

HAMRR is designed to provide long-term at-home training and rehabilitation during the 12-24 months after clinic therapies. Thus, the basic idea of multimedia

feedback system is to provide a dynamic environment which helps a stroke survivor restore motion function through self-assessment and distanced supervision by therapists. A multi-layer feedback hierarchy is proposed to help stroke surveyor evolve over time and regain self-confidence. Details can be seen in [29].

2.1.3 ADAPTATION

HAMRR is designed to provide a long-term, distanced semi-supervised therapy. During the weekdays, the participants are expected to conduct physical tasks at home by themselves, and in the weekend, a therapist reviews the participant progression and adjust tasks and goals dynamically. The types of tasks, tangible objects and locations, feedback streams, and feedback sensitivities and all designed to be adaptable in order to offer challenging and engaging tasks based on the progress of specific participants. HAMRR employs a utility function to determine the sequence of sets and parameters based on 1) a prior established week-long sequence of tasks; 2) the history of foci and tasks for each set; and 3) the participant's performance. Details for adaption framework can be found in [10].

2.2 KINECT SENSING TECHNOLOGY

Novel motion-sensing technology has been leading a revolutionary change in the gaming industry by creating an engaging and interactive environment. During the past few years, the remarkable success of Nintendo Wii and Microsoft Kinect has been attracting researchers to uncover the potential of using these technologies into applications. The idea of applying these motion-sensing devices to the development of home-based stroke rehab system is intuitive. This is because motion-sensing devices are designed for at home video game applications, low-cost and easy-setup are prerequisites. Also, they use human motion as one of the inputs in this game. This requires a reliable motion sensing for real-time human body representation.

Table 2.1: Kinect for Windows Specifications

Kinect	Specifications
Sensor	Color and Depth Cameras
	IR projector
	Voice microphone array
	Tilt motor for sensor adjustment
Field of View	
Angle Ranges	(Horizontal) 57 degrees
	(Vertical) 43 degrees
	(Physical tilt range) +/- 27 degrees
Distance Ranges	(Default Mode) 0.8 to 4 m
	(Near Mode) 0.4 to 3m
Resolution	320×240 or 640×480 Depth
	320×240 or 640×480 or 1280×960 Color
FrameRate	30 fps Depth
	30 fps @ 320×240 , 640×480 Color
	15fps @ 1280×960
Skeleton Tracking System	Tracks up to 6 players (2 active players)
	(Default Mode) 20 joints per active player
	(Seat Mode) 10 joints per active player

2.2.1 DEVICE SPECIFICATIONS

Kinect contains a USB hub with three different devices: A camera device with an IR projector, a depth camera and a RGB camera; an audio device equipped with a multi-array microphone; a motor/LED device. In this system, the datastream obtained from the depth and RGB camera is used as input. Table 2.1 shows the specification for the Kinect device [4][2].

Depth maps are created by continuously projecting an infrared 'static pseudorandom' pattern onto a 3D environment and further using stereo triangulation [27]. Body parts are then inferred from depth maps using random decision forest classifiers, which are trained from one million training samples [45].

2.2.2 PROS AND CONS OF KINECT

The reasons why Kinect is beneficial for motion capture applications are three-fold:

1. Kinect sends out RGB and depth data with the resolution of 640 $\times$ 480 at each 30 ms, which provides rich data for real-time applications.

2. Kinect can track up to two skeletons without markers and for each skeleton 20 joints can be tracked, which greatly simplified the motion capture setup. Compared to the marker-based system, the markerless upperbody solution significantly enhances the participant's experience.

3. Kinect is easy to set up and use. Compared to the multi-camera system, Kinect is a portable single camera. It doesn't require stereo imaging among different cameras. The hands-free control could provide also possibilities for designing an engaging and interactive environment.

However, Kinect sensing also has certain limitations and problems which are discussed as follows.

1. Lighting: Lighting is important for image quality because high illumination makes depth tracking less reliable while low illumination works for depth but degrade the RGB. Since the depth image is generated by 'light coding' using IR projector, it works poorly when items or clothing materials are reflective.

2. Distance: Depth Camera works well within very limited range of distances. It will lead to unstable and incorrect skeleton representations and slow calibrations at out-of-range distances.

3. Image Quality: Depth Image contains many noises on the edges between the background and user body contours. This requires preprocessing work on data smoothing and denoising.

2.2.3 RELATED WORK

Kinect [30][45], as an inexpensive motion capture device, has impacted many computer vision applications, such as tracking [38][35], activity and gesture recognition [48][36][23]. The applications of Kinect in rehabilitation and related healthcare applications have recently been investigated [37][8]. However, these investigations were focused on the accuracy of tracking alone, and they found that the Kinect offered reasonable accuracy as measured in terms of pure trajectory level error. They did not, however, report whether trajectory errors have any impact on higher-level movement quality classifiers that form the core of any rehabilitation application.

2.3 MEANSHIFT TRACKING

Object tracking is an important task in the computer vision domain. The rapid growth of computing capability, along with the emergence of high quality and inexpensive camera and the increasing need for automated object tracking algorithms, has been attracting researchers to conduct a variety of research. There are primarily three key steps in activity and motion analysis: the detection of moving objects, or frame-to-frame object tracking, and analysis of object tracks to recognize their behaviors [57]. From a bottom-up perspective, an object tracking problem starts from how to represent and model interesting object. The next step is to segment the object from its background. The last step is to locate the object frame-to-frame. The first step is called target representation, while the second step is called target localization.

An object can be represented in different ways, such as points, geometry shapes, silhouette, contour and skeletal models, depending on its applications. For example, single small objects can be regarded as points, while for tracking articulated objects, skeletal models are commonly applied. Colors, edges, optical flows and textures are most common features that are selected to build object models.

Most tracking approaches combine different types of features for object detection.

Object tracking is a difficult problem because of: 1) the loss of information caused by projection of the 3D world on a 2D image; 2) complex object motion; 3) object shape deformation; 4) real-time processing requirements; and 5) partial and full object occlusions.

Mean shift tracking [16] uses feature histogram-based target representations regularized by spatial masking with an kernel. The tracking problem is formulated by finding the local maxima, or mode in the feature space. A Bhattacharyya coefficient as similarity measure is used for algorithm optimization. Mean shift is a fast, efficient tracking approach and has been widely used in different tracking applications. However, the mean shift is sensitive to background noises and rotations, neither the global optimality is guaranteed [44]. A large amount of research has been proposed to improving the approach by combining other tracking algorithms, such as Kalman filter [60], Particle filter [44], or better object detection solutions [59].

2.3.1 BACKGROUND

Define a set of normalized pixel locations $\{\mathbf{x}_i^\star\}_{i=1\ldots m}$ centered at 0 as the target model. An kernel function $k(x)$ is applied to assign smaller weights to pixels farther from the center. The function $b : \mathbb{R}^2 \to \{1\ldots m\}$ associates to the pixel at location $\mathbf{x}_i^\star$ the index of $b(\mathbf{x}_i^\star)$ of its bin in the quantized feature space. The probability distribution of the feature $u = 1\ldots m$ in the target model is then computed as [16]

$$\hat{q}_u = C\sum_{i=1}^{n} k(\| \mathbf{x}_i^\star \|^2)\delta\left[b(\mathbf{x}_i^\star) - u\right], \tag{2.1}$$

where δ is the Kronecker delta function, and C is the normalization constant which is derived by imposing the condition $\sum_{u=1}^{m}\hat{q}_u = 1$. Let the $\{\mathbf{x}_i\}_{i=1\ldots n_h}$ be the normalized pixel locations of the target candidate, centered at $\mathbf{y}$ in the current frame.

The probability distribution of the feature $u = 1 \ldots m$ in the target candidate is given by

$$\hat{p}_u(\mathbf{y}) = C_h \sum_{i=1}^{n_h} k(\| \frac{\mathbf{y} - \mathbf{x_i}}{h} \|)^2 \delta[b(\mathbf{x_i} - u)] \tag{2.2}$$

The maxima of the similarity function is achieved by minimizing the Bhattacharyya coefficient between $\mathbf{p}$ and $\mathbf{q}$:

$$\rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(\mathbf{y})\hat{q}_u}. \tag{2.3}$$

Using Taylor expansion around the values $\hat{p}_u(\hat{\mathbf{y}}_\mathbf{0})$, the Bhattacharyya coefficient could be represented by

$$\rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}] \approx \frac{1}{2} \sum_{u=1}^{m} \sqrt{\hat{p}_u(\hat{\mathbf{y}}_\mathbf{0})\hat{q}_u} + \frac{C_h}{2} \sum_{i=1}^{n_h} w_i k(\| \frac{\mathbf{y} - \mathbf{x}_i}{h} \|^2), \tag{2.4}$$

where

$$w_i = \sum_{u=1}^{m} \sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\mathbf{y}}_\mathbf{0})}} \delta[b(\mathbf{x}_i) - u]. \tag{2.5}$$

To minimize, mean shift procedure is employed, the kernel is recursively moved from the current location $\hat{\mathbf{y}}_0$ to new location $\hat{\mathbf{y}}_1$ according to the relation

$$\hat{\mathbf{y}}_1 = \frac{\sum_{i=1}^{n_h} \mathbf{x}_i w_i g(\| \frac{\hat{\mathbf{y}}_0 - \mathbf{x}_i}{h} \|^2)}{\sum_{i=1}^{n_h} w_i g(\| \frac{\hat{\mathbf{y}}_0 - \mathbf{x}_i}{h} \|^2)}, \tag{2.6}$$

where $g(x) = -k'(x)$, assuming that the derivative of $k(x)$ exists for all $x \in [0, \infty)$, except for a finite set of points.

2.3.2 KALMAN FILTER

Kalman Filter [52] is commonly applied in mean shift approach to solve the problem of losing tracking caused by fast motion and occlusions. The Kalman filter model assumes the space state of a discrete-time k evolves from the state at $k-1$ by the linear stochastic difference equation [52]

$$x_k = Ax_{k-1} + Bu_{k-1} + wk_1, \tag{2.7}$$

with a measurement z_k that is

$$z_k = Hx_k + v_k, \tag{2.8}$$

where w_k and v_k are independent, white, and with normal probability

$$p(w) \sim N(0,Q), p(v) \sim N(0,R), \tag{2.9}$$

where A is the state transition model applied to the previous state x_{k-1}, B is the control-input model, and H is the observation model which maps the true state space into the observed space. The models A,B,H may change between states, but they are assumed to be stable here. The Kalman filter estimates a process by using a form of feedback control - the filter estimates the process state at some time and then obtains feedback in the form of measurements. The Kalman filter algorithm includes two updates stages:

- time update (Predict):

$$\hat{x}_k^{'} = A\hat{x}_{k-1} + Bu_{k-1} \tag{2.10}$$

$$P_k^{'} = AP_{k-1}A^T + Q \tag{2.11}$$

- measurement update (Correct):

$$K_k = P_k^{'}H^T(HP_k^{'}H^T + R)^{-1} \tag{2.12}$$

$$\hat{x}_k = \hat{x}_k^{'} + K_k(z_k - H\hat{x}_k^{'}) \tag{2.13}$$

$$P_k = (I - K_kH)P_k^{'} \tag{2.14}$$

2.3.3 RELATED WORK

A number of methods have been addressed to overcome the limitations mentioned above. There are primarily two lines of research based on the different steps of the object tracking problem.

Target Representation: Extensive work on target representation can be divided into two groups. In the first line of research, a handful of research has aimed to modify the feature models or kernel formulation to improve the tracker's performance. Traditional mean shift method requires a symmetric kernel and assumes constancy of the objet scale and orientation during tracking. Asymmetric kernel based on mean shift methods is presented to improve the robustness in terms of scale and orientation changes [57][51]. A Difference of Gaussian (DOG) mean-shift kernel enables efficient tracking of blobs through scale space [15]. Others attempt to modify the feature models. In [31], the author presented an adaptive binning color model for mean shift tracking in order to chi eve the number of subspaces automatically. This was different from the conventional mean shift which lacked a systematic way to determine bin number. In [50], an online updating appearance generative mixture model for mean shift tracking is proposed. A new spatial color histogram is applied in [54]. In the second line, efforts have been devoted to replacing the color histogram model with other features and objection detectors. In [59][9], they combined the benefits of SIFT features and color features based mean shift and evaluate them in an expectation-maximization scheme in order to achieve a maximum likelihood estimation of similar regions. This was similar to the approach used In [55].

Target Localization: Extensive work on target localization has generally involved adding Kalman filters or particle filters to improve the tracking robustness when partial or full collusion of the objects occurs. In [44], a mean shift embedded particle filter method is proposed. This approach produces reliable tracking while effectively handling rapid motion and distraction. In [60], a real time eye tracking method combing Kalman filter and mean shift tracking is presented. The experiment shows that the robustness has significantly been improved in term of handling occlusion.

2.4 3D CARTESIAN COORDINATE SYSTEM AND CAMERA CALIBRATION

3D coordinate system uses a geometric 3-parameters model to represent three-dimensional space [3]. Since our physical universe is three- dimensional, the 3D coordinate system is used to represent the locations in real world. Cartesian coordinate system describes every point in 3D space by means of three orthogonal axes labeled x, y, and z. In the motion analysis domain, *xandy* are used to represent the image plane and z to represent the vertical or depth. A coordinate system is comprised of an origin O, and three orthogonal unit vectors $\boldsymbol{i}, \boldsymbol{j},$ and $\boldsymbol{k}$. The direction of these three vectors follows the *right-handed rule* [3].

2.4.1 COORDINATE REPRESENTATION

In this section, the 3D representations of some basic geometric objects are provided using the Cartesian coordinate system. These basic geometric objects are as follows:

Point: A point P in the coordinate system F is represented by the (signed) lengths of the orthogonal projections of the vector $\overrightarrow{OP}$ onto the vector $\boldsymbol{i}, \boldsymbol{j},$ and $\boldsymbol{k}$, with

$$\begin{cases} x = \overrightarrow{OP} \cdot \boldsymbol{i} \\ y = \overrightarrow{OP} \cdot \boldsymbol{j} \\ z = \overrightarrow{OP} \cdot \boldsymbol{k} \end{cases} \iff \overrightarrow{OP} = x\boldsymbol{i} + y\boldsymbol{j} + z\boldsymbol{k} \tag{2.15}$$

Distance between points: Cartesian coordinate system follows Euclidean space and thus the distance D between two points $P_1(x_1, y_1, z_1), P_2(x_2, y_2, z_2)$ is

$$D = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \tag{2.16}$$

Plane: The points lying in $\prod$ are characterized by

$$\overrightarrow{AP} \cdot \boldsymbol{n} = 0 \tag{2.17}$$

, where the coordinates of point P is x, y, z and the coordinates of $\boldsymbol{n}$ are a, b, and c, and a more general representation of plane $\prod$ is

$$ax + by + cz + d = 0, \tag{2.18}$$

Distance from a point to a plane: the shortest distance from a point P_1 to a plane $\prod : ax + by + cz + d = 0$ is

$$D = \frac{|ax_1 + by_1 + cz_1 + d|}{\sqrt{a^2 + b^2 + c^2}} \tag{2.19}$$

Intersection Angles between Planes: The intersection angle θ between two planes $\prod_1, \prod_2$ is defined by

$$\theta = arccos(\frac{a_1a_2 + b_1b_2 + c_1c_2}{\sqrt{a_1^2 + b_1^2 + c_1^2}\sqrt{a_2^2 + b_2^2 + c_2^2}}). \tag{2.20}$$

2.4.2 COORDINATE SYSTEM CHANGES AND RIGID TRANSFORMATION

It is common to change coordinate system in order to get different representations of a point, line or plane. In motion analysis system, the computation module gets 2D representations of a real-world object from a camera, and transform the position into a calibrated 3D global coordinate system, while the visualization module projects the 3D position into a 2D image plane. Also, using different coordinates are needed in order to compute the space correlation between points, lines or planes. Any coordinate system can be considered the production of rigid transformations from another coordinate system [22]. Two transformations that preserve distances between points - translations and rotations - are particularly helpful in this study.

Consider 2 coordinate systems: $(A) = (O_A, \boldsymbol{i}_A, \boldsymbol{j}_A, \boldsymbol{k}_A)$, $(B) = (O_B, \boldsymbol{i}_B, \boldsymbol{j}_B, \boldsymbol{k}_B)$. If $(\boldsymbol{i}_A, \boldsymbol{j}_A, \boldsymbol{k}_A)$ and $(\boldsymbol{i}_B, \boldsymbol{j}_B, \boldsymbol{k}_B)$ are parallel to each other, and O_B can be described as

$$\overrightarrow{O_BP} = \overrightarrow{O_BO_A} + \overrightarrow{O_AP} \tag{2.21}$$

, the two systems thus are separated by a *pure translation*; If O_B and O_A are identical and the two systems thus are separated by a *pure rotation*. The *rotation matrix* ${}^{A}R^{B}$ as the 3×3 is defined as the array of numbers

$$ {}^{A}R^{B} = \begin{pmatrix} \boldsymbol{i}_A \cdot \boldsymbol{i}_B & \boldsymbol{j}_A \cdot \boldsymbol{i}_B & \boldsymbol{k}_A \cdot \boldsymbol{i}_B \\ \boldsymbol{i}_A \cdot \boldsymbol{j}_B & \boldsymbol{j}_A \cdot \boldsymbol{j}_B & \boldsymbol{k}_A \cdot \boldsymbol{j}_B \\ \boldsymbol{i}_A \cdot \boldsymbol{k}_B & \boldsymbol{j}_A \cdot \boldsymbol{k}_B & \boldsymbol{k}_A \cdot \boldsymbol{k}_B \end{pmatrix} \tag{2.22} $$

. The rotation matrix ${}^{A}R^{B}$ is computed as follows:

$$ {}^{A}R^{B} = [M_x^B, M_y^B, M_z^B], \tag{2.23} $$

, where M_x^B, M_y^B, M_z^B are orthogonal unit vectors of the O_B. They can be easily computed as follows:

$$ M_x^B = \frac{v^{12}}{\|v^{12}\|}, M_y^{B'} = \frac{v^{23}}{\|v^{23}\|}, \tag{2.24} $$

$$ M_z^B = M_x^B \times M_y^{B'}, M_y^B = M_x^B \times M_z^B, \tag{2.25} $$

$$ v^{12} = (x_2 - x_1)\vec{i} + (y_2 - y_1)\vec{j} + (z_2 - z_1)\vec{k}, \tag{2.26} $$

$$ v^{23} = (x_3 - x_2)\vec{i} + (y_3 - y_2)\vec{j} + (z_3 - z_2)\vec{k} \tag{2.27} $$

, where $(x_1, y_1, z_1), (x_2, y_2, z_2)$ and (x_3, y_3, z_3) are 3D coordinates of any three points that are not in a line in coordinate system O_A. Without loss of generality, we assume (x_2, y_2, z_2) as the origin of O_B. The translation vector ${}^{A}\mathbf{t}^{B}$ is computed by

$$ {}^{A}\mathbf{t}^{B} = [-x_2, -y_2, -z_2] \tag{2.28} $$

2.4.3 EULER JOINT ANGLE COMPUTATION

The rotation matrix ${}^{A}R^{B}$ can be considered a sequence of three rotations, corresponding to three axises, respectively. The three rotation matrices are defined as $R_x(\psi), R_y(\theta)$, and $R_z(\phi)$, where ψ, θ, and ϕ represent the rotation radians. These three angles are called Euler angles [47].

The rotation matrix varies across different rotation orders. If rotate first from A to B around x-axis, then the y-axis, and finally the z-axis, the rotation matrix can be represented as,

$$
\begin{aligned}
{}^{A}R^{B} &= R_z(\phi)R_y(\theta)R_x(\psi) \\
&= \begin{pmatrix} \cos\theta\cos\phi & \sin\psi\sin\theta\cos\phi - \cos\psi\sin\theta & \cos\psi\sin\theta\cos\phi + \sin\psi\sin\theta \\ \cos\theta\sin\phi & \sin\psi\sin\theta\sin\phi + \cos\psi\cos\theta & \cos\psi\sin\theta\sin\phi - \sin\psi\cos\theta \\ -\sin\theta & \sin\psi\cos\theta & \cos\psi\cos\theta \end{pmatrix}
\end{aligned} \tag{2.29}
$$

, then the three angles can be computed using the algorithm in [47].

2.4.4 CAMERA CALIBRATION

The goal of camera calibration is to find the transformation matrix that transforms a 2D point position in pixel coordinates into a defined 3D point position in world coordinates. Homogeneous coordinates are used to represent the transformation

$$
\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = sMW \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{2.30}
$$

, where s is an arbitrary scale factor, M is an intrinsic camera matrix which is represented by

$$
M = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{2.31}
$$

, where f_x and f_y are focal lengths on x and y axes, and c_x and c_y is the center of the image plane. W is the extrinsic camera matrix which is represented by

$$
W = \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} \tag{2.32}
$$

, where R is a 3×3 rotation matrix from the 2D pixel coordinate to the 3D world coordinate, and $\mathbf{t}$ is a 3×1 translation vector from the origin of 2D pixel coordinate to the origin of 3D world coordinate. Without loss of generality, we assume the plane on which all the points satisfy $Z = 0$ as the global coordinate. Then

$$\begin{bmatrix} x_p \\ y_p \\ 1 \end{bmatrix} = sM \begin{bmatrix} \mathbf{r_1} & \mathbf{r_2} & \mathbf{t} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} \tag{2.33}$$

, we denote $\mathbf{H} = \begin{bmatrix} \mathbf{h_1} & \mathbf{h_2} & \mathbf{h_3} \end{bmatrix}$, and then we have

$$\begin{bmatrix} \mathbf{h_1} & \mathbf{h_2} & \mathbf{h_3} \end{bmatrix} = \lambda M \begin{bmatrix} \mathbf{r_1} & \mathbf{r_2} & \mathbf{t} \end{bmatrix} \tag{2.34}$$

Because the rotation vectors are orthonormal, thus

$$\mathbf{r_1^T r_2} = 0 \tag{2.35}$$

$$\| \mathbf{r_1} \| = \| \mathbf{r_2} \| \tag{2.36}$$

. From we can get

$$\mathbf{r_1} = \lambda M^{-1} \mathbf{h_1}, \mathbf{r_2} = \lambda M^{-1} \mathbf{h_2} \tag{2.37}$$

and 2.35 can be written as

$$\mathbf{h_i}^T M^T M^{-1} \mathbf{h_j} = 0, i \neq j \tag{2.38}$$

$$\mathbf{h_i}^T M^T M^{-1} \mathbf{h_i} = \mathbf{h_j}^T M^T M^{-1} \mathbf{h_j}, i \neq j \tag{2.39}$$

Set $B = M^T M^{-1}$, since M is the intrinsic matrix, B can be represented as

$$B = \begin{bmatrix} \frac{1}{f_x^2} & 0 & \frac{-c_x}{f_x^2} \\ 0 & \frac{1}{f_x^2} & \frac{-c_y}{f_y^2} \\ \frac{-c_x}{f_x^2} & \frac{-c_y}{f_y^2} & \frac{c_x^2}{f_x^2} + \frac{c_y^2}{f_y^2} + 1 \end{bmatrix} \tag{2.40}$$

Because B is symmetric, then $\mathbf{h_i^T} B \mathbf{h_j} = v_{ij}^T b$, and the 2.38 can be written as

$$\begin{bmatrix} v_{12}^T \\ (v_{11} - v_{22})^T \end{bmatrix} b = 0 \tag{2.41}$$

this linear equation can be solved if 2 images of chessboards together, and all the parameters in both the intrinsic and extrinsic matrices can be solved. Please see [58] for details.

Chapter 3

A LOW-COST DESIGN OF MOTION CAPTURE USING KINECT FOR AT-HOME MIXED REALITY REHABILITATION SYSTEM

In this chapter, the design of the upper body motion capture using inexpensive RGB-D camera is introduced for home-based adaptive mixed reality rehabilitation system (HAMRR). HAMRR aims to help restore the motion function of stroke survivors by providing an engaging rehab physical therapy at home at a low cost [46]. It is a scaled-down version based on the theories and results obtained from the Adaptive Mixed Reality Rehabilitation System (AMRR) [12]. HAMRR tracks movement of the wrist and torso and provides real-time, post-trial, and post-set feedback to encourage stroke survivors to self-assess their movement and to engage in active learning of new movement strategies.

Motion capture plays an important role in mixed reality system, as continuously providing reliable and accurate information on joint trajectories of upper body for feature calculation and feedback control. In the AMRR system [13], a commercial tracking system called Opti-Track is used as a solution for motion capture. 12 reflective markers are equipped on patient's upper body, tracked by 6 infrared cameras with 100 frames per seconds. The Opti-Track system provides rich information on movements of hands, wrists, arms, and torso. However, when the system is required to be transferred from hospital environment to home environment, motion capture needs to be remodeled in order to find in a lower cost and easy-to-setup solution.

It is challenging to design a low-cost but reliable motion capture module for the mixed reality rehabilitation applications because:

- Compared to high quality sensors, low-cost sensors provide noisy and unreli-

able data with a lower sampling rate which may lead to problems on motion analysis. For example, when the movement quality is evaluated in terms of its speed profile and segmentation in previous system, about 200 frames are sampled for the calculation of the features, but only 70 frames are collected to represent the same reaching in low-cost system. The loss of data may cause inconsistency in representing a fast movement.

- In order to provide comprehensive kinematic representations of upper extremity, different types of sensors are required to be integrated. The system aims to help participants get rid of complex assistive robot arms and markers to bring long-term supervised therapy into daily experience, thus it is difficult to get motion data for the whole upper body using any single sensing component. Sensor selection and integration are challenging.

In this chapter, we discuss the design of a multimodal sensing module to address the problem mentioned above. Specifically, it includes: 1) How to integrate different types of sensors in order to combining the benefits of high-end and inexpensive motion capture technologies, and 2) a presentation of a hybrid upper body tracking approach as well as a study of the effect of accuracy for endpoint and torso tracking during reaching and grasping tasks.

3.1 MULTIMODAL SENSING SYSTEM

The HAMRR system utilizes multiple sensing modalities to extract kinematic features of a participant's movement, providing a cost-effective and robust sensing solution for unsupervised, private home training [6]. In this section, we first introduce how different sensing components integrate to provide reliable movement data, and then discuss the usability of Kinect camera. Previously, the end-point tracking was achieved by tracking a reflective marker wearing on a wristband through Opti-Track

camera system. Torso tracking was achieved by sensors from chair, while tangible sensors are applied for generating tangible feedback.

Practically, several problems are found with the existing motion capture solution:

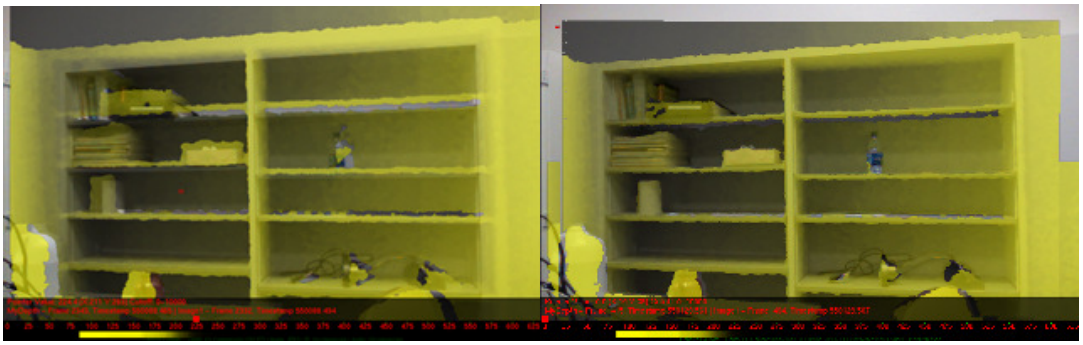
- Although the Opti-Track camera system is reduced to a 4 camera setup, it is still expensive.
- The chair sensors are so noisy and sensitive that it is difficult to evaluate the torso movement quality. Additionally , the chair sensors do not work when a participant's torso is off the chair. Thus, using a chair is not a feasible solution for tracking torso compensatory movements. Opti-track can track torso with rigid-body markers, but additional cameras are needed.
- There is no efficient way for elbow and shoulder joint tracking.

A conclusion is drawn from the pros and cons mentioned in section 2.2, that is, of great value to conduct a systematic analysis of the tradeoffs encountered in the richness and accuracy of the acquired data by Kinect as compared to a high-end multi-camera motion capture system.

In the next section, the usability of Kinect in upper body tracking is discussed. We also study the effect of tracking of different segments using Kinect and then determine the integration of different sensing components.

3.2 CAMERA CALIBRATION

Camera calibration is an important pre-stage of tracking. We want to send out joint positions in a 3D space while the input of a camera is in a projective 2D plane. Camera calibration can be divided into two stages: intrinsic calibration and extrinsic



(a) Unaligned Image (b) Aligned Image

Figure 3.1: The images are made by overlaying depth image on color image. We can see a clear offset between depth and color image on the bottles and books in (a). Note the field of view of depth camera is smaller than color camera. So the depth imaging is not available on the edge of the color image.

calibration [7]. Intrinsic calibration refers to calculating the intrinsic matrix parameters that enable the transformation from a 2D pixel coordinate to a 3D global coordinate. Extrinsic calibration refers to finding rotation matrix and translation vector that represent the transformation from the 3D global coordinate to user-defined 3D local coordinate. Next we describe how these calibrations are conducted.

Calibration for multi-camera system is complex. First, it needs to find the intrinsic and extrinsic parameters for each camera. Second, image registration work is required to map each pixel in one camera with the corresponding one in another camera. The alignment work of the depth camera and color camera is done using functions in the SDKs. The coordinate of point A in image based the 2D coordinates of color camera is (x_p, y_p), and then the coordinate of point A in aligned images can be represented as $(x_p, y_p, D(x_p^{'}, y_p^{'}))$, where $D(x_p^{'}, y_p^{'})$ refers to the depth value of aligned pixel of the color image point (x_p, y_p) in depth image. Figure 3.1 illustrates the depth and color images before and after the image alignment. After the image alignment, the multi-camera calibration is then transferred into single camera calibration. Next, the calibration is conducted to find the intrinsic and extrinsic parameters using the color camera.

3.2.1 INTRINSIC PARAMETERS

Intrinsic calibration refers to calculating the intrinsic parameters that enable the transformation from a 2D coordinate to a 3D global coordinate. Zhang's calibration method [58] is applied here. We use a pattern of black and white squares (e.g. chessboard as a calibration object), which ensures that there is no bias toward one side or the other in measurement. OpenCV has wrapped the calibration function so that we can directly apply it in the system. The chessboard has 6×4 corners and in practice we rotate the chessboard to obtain a rich set of views. The program could

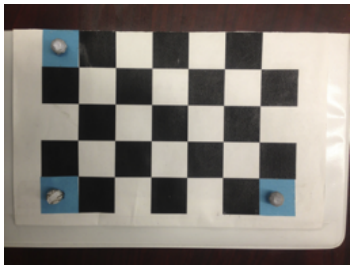


Figure 3.2: Calibration Object, the three reflective markers are the L-frame for determining base coordinate system.

automatically detect the corners on the chessboard. Then we used these corners to fix the unknown parameter in intrinsic matrix according to (2.38), (2.41). In order to obtain high-quality results, we did the experiment 10 times, and then averaged across the results. The results are provided in table 3.1.

After the intrinsic calibration, a 3D position is obtained for each corresponding point in the 2D projective plane. Since both the rotation and translation preserve the euclidean distance between points, we can test the accuracy of intrinsic parameters by taking the opti-track result as ground truth, and compute the distance errors between two points in Kinect 3D global coordinate.

Table 3.1: Intrinsic parameters for Kinect color camera

Para Name	Value
f_x	527.96
f_y	530.62
c_x	315.94
c_y	249.10

3.2.2 EXTRINSIC PARAMETERS

Extrinsic calibration refers to finding rotation matrix and translation vector that represent the transformation from the 3D global coordinate to user-defined ground zero coordinate. Because Opti-Track and Kinect system are both used for endpoint tracking, a unified coordinate system needs to be defined. As shown as figure, three reflective markers can be seen on the chessboard, which is represented as a L-frame. The L-frame is used to set up a marker-based coordinate system which takes the *legs* and *catheti* of the marker-based triangular as the x and y axes. The 3D positions of the three markers labeled A, B, and C in marker-based global coordinate O_i can be represented as A, B, and C are $(x_A^i, y_A^i, z_A^i), (x_B^i, y_B^i, z_B^i), (x_C^i, y_C^i, z_C^i)$, respectively.

In the next step, 3D positions are located in global camera coordinate system O_c based on the Kinect color camera. The markers' 2D positions are labelled as $(x_A^p, y_A^p), (x_B^p, y_B^p), (x_C^p, y_C^p)$ in the image plane. Then, the 3D coordinates in O_c are calculated by

$$\begin{bmatrix} x^c \\ y^c \\ z^c \end{bmatrix} = \begin{bmatrix} \frac{1}{f_x} & 0 & 0 \\ 0 & \frac{1}{f_y} & 0 \\ -\frac{c_x}{f_x} & -\frac{c_y}{f_y} & 1 \end{bmatrix} \begin{bmatrix} x^p \\ y^p \\ D(x^{p'}, y^{p'}) \end{bmatrix} \tag{3.1}$$

Then we compute the rotation matrix ${}^cR^i$ and translation vector ${}^cT^i$ based on (2.23)

and (2.28) . The extrinsic parameters are :

$$ {}^{c}R^{i} = \begin{bmatrix} 0.043411 & -0.992503 & -0.114250 \\ 0.209391 & 0.115218 & -0.921350 \\ 0.927606 & 0.016074 & 0.212823 \end{bmatrix}, \tag{3.2} $$

$$ {}^{c}T^{i} = \begin{bmatrix} -164.863846 & 139.966827 & 1108.0000 \end{bmatrix} \tag{3.3} $$

3.3 TRACKING

The motion capture module plays an important role in our HAMRR [12] by exporting accurate and robust motion data during the repetitive physical therapy. These motion data consist of a set of important joints that represent the articulated human body. The motion analysis module evaluates patient's kinematic representation by analyzing these joint trajectories and angles. Human upper body consists of several segments: head, torso, arms and hands. The movements of these segments are captured by tracking the joint positions and angles on upper body, such as shoulders, elbows, wrists, neck, and hips.

Both OpenNI and KinectSDK use skeleton tracking algorithms, which can track up to 20 joints in human body without wearing any markers. Moreover, they are invariant to scale, rotation, occlusion of the body and light changes. Thus, they are both sound alternative solutions for tracking. However, most physical rehab tasks, such as reaching, grasping and lifting, require high accuracy on endpoint tracking. Therefore, a better endpoint tracking algorithm is needed. Figure 3.3 shows an overview of the tracking approach.

3.3.1 DEPTH BASED TORSO AND ARM TRACKING

The assumption underlying skeleton tracking algorithms is described as follows: When a participant enters the scene, several consecutive frames are collected for segmenting the 'participant pixels' from 'background pixels' in depth images. Then

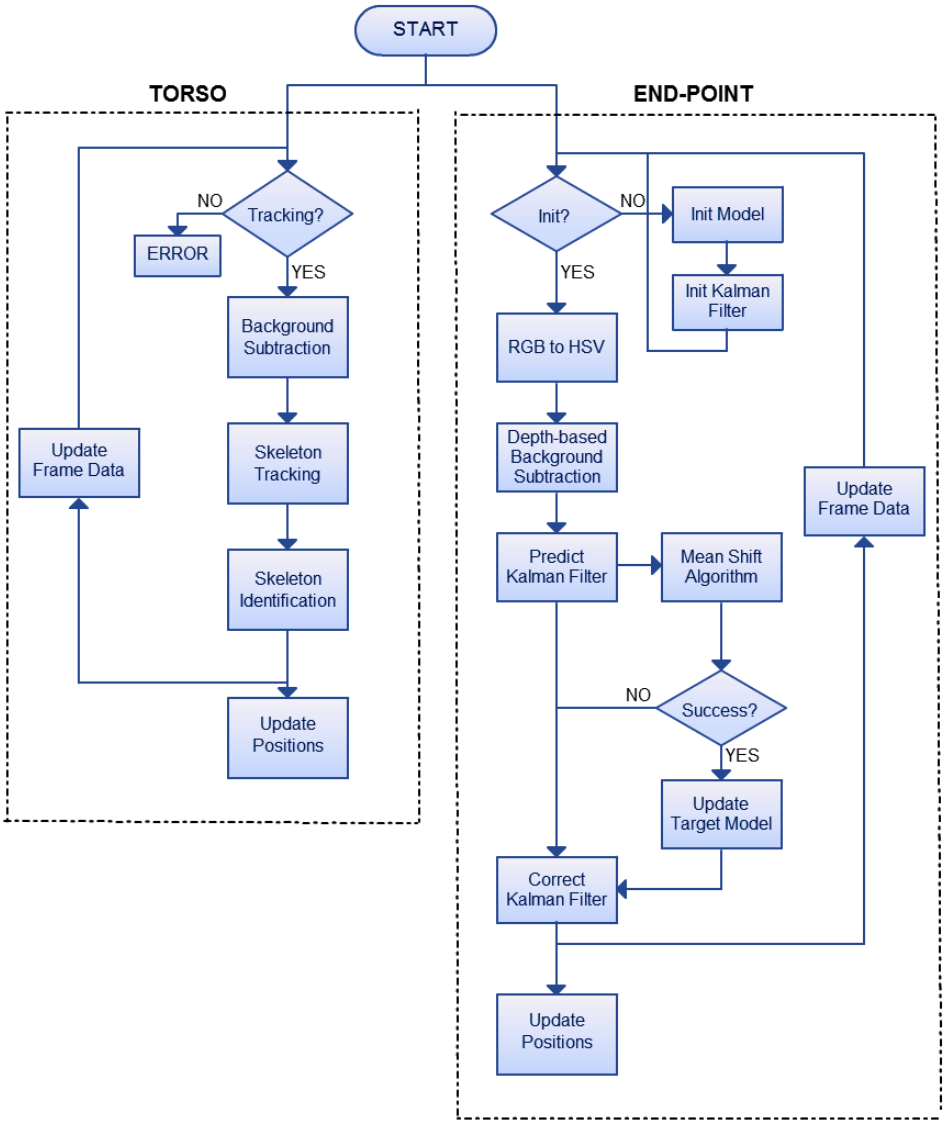


Figure 3.3: Tracking approach flowchart. Torso tracking is achieved by using off-the-shelf skeleton tracking algorithm in Kinect SDKs, and endpoint tracking is achieved by using a depth-fused mean shift tracking algorithm.

a specific label is assigned to a connected region that is considered the i_{th} participant's body. Classification algorithm is used to estimate the centroids of each segmented body parts which construct a skeleton representation of human body.

Both the skeleton tracking algorithms from Kinect SDK and OpenNI can track up to 20 joint positions and orientations running at 30Hz, providing rich information on analyzing torso and arm movements. We use the joints named

SKEL_LEFT_SHOUDLER, SKEL_RIGHT_SHOUDLER, SKEL_TORSO to represent the torso movement, and *SKEL_RIGHT_SHOULDER, SKEL_RIGHT_ELBOW, SKEL_RIGHT_HAND* to represent right arm movement. Next we compare the two algorithms from the concerned aspects as follows:

- **Skeleton Calibration**: Skeleton calibration rather than camera calibration is used to refer to the process through which human body skeleton is estimated after a participant is tracked. The skeleton calibration is a pre stage of skeleton tracking, which is used to estimate the participant's body postures based on image sequences. The latest version of OpenNI and Kinect SDK both complete the calibration process automatically, which means neither of them require the participant to do a 'T' pose for calibration as before. In practice, however, the calibration in OpenNI takes longer time. Since the system is expected to be deployed into a patient's home, it may cause inconvenience on interaction between the system and the participant.

- **Joint tracking accuracy and stability**: The skeleton tracking precision drops in the HAMRR system as compared to normal applications, because during the motion capture process, the lower limbs are occluded and only partial skeleton can be tracked. Results show both OpenNI and Kinect SDK provide robust tracking on torso. However, OpenNI works poorly on tracking arms while seating.

- **Software Compatibility**: Kinect SDK only works on natural Windows, while our Kinect system is running on a virtual machine.

As a result, we choose OpenNI as the better solution for torso tracking.

The process of torso and arm tracking is straightforward. The skeleton algorithm first detect all the active users $\{P_1, P_2, \ldots, P_n\}$. Because the patient is sitting

in front of the camera. We can find the patient's userID P_{user} from the following:

$$\sum_{i=1}^{M} Z_{P_{user}}(i) = \min_{j=1,2,\ldots,n} \sum_{i=1}^{M} Z_{P_j}(i) \tag{3.4}$$

, where $Z_{P_j}(i)$ denotes the depth value of the ith joint of user j, M is the max joints number. Then we update the joint positions of left shoulder, right shoulder, torso, right elbow, and right wrist for feature calculations in motion analysis module.

3.3.2 ENDPOINT TRACKING

In HAMRR, endpoint tracking quality is crucial for evaluating participant's motor function during repetitive physical tasks, since most feature computations are derived from the raw endpoint trajectories and movement speed [13]. However, results are not reliable using skeleton tracking. First, it is known that the skeleton tracking algorithm exhibits lower precision on limb joints than torso joints [45], and tracking robustness is even worse when it comes to seated skeleton mode a part of the body is occluded. Even with the recently released Microsoft SDK for upper-body tracking, low accuracy is observed in the end-point tracking compared to torso tracking. Secondly, endpoint tracking is very sensitive to articulations of the palm. The limitations mentioned above are illustrated in figures 3.4 and 3.5 respectively.

Hence, there is need to develop a more reliable endpoint tracking method. For real-time applications, it is desirable to keep the tracking complexity as low as possible in order to allocate system resources to other high-level processing manipulations [32], thus we want to simplify the tracking algorithm by adding constraints that are making sense for this specific problem.

We adopt a marker-based tracking because, although markerless hand tracking approaches are widely proposed, large computation on hand gesture recognition makes it difficult for real-time applications. Unlike normal hand tracking, marker-based tracking, through using color, shape or texture, can exhibit good clustering

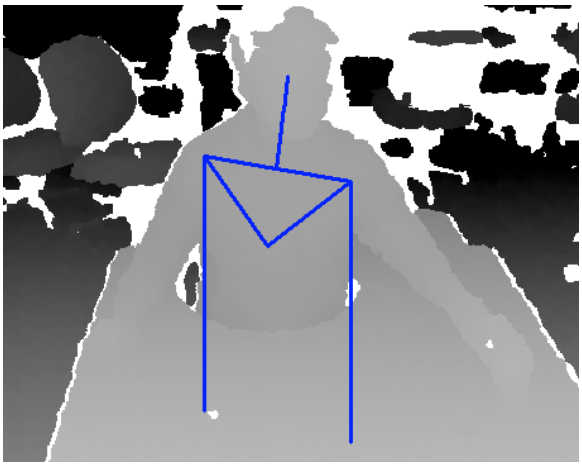
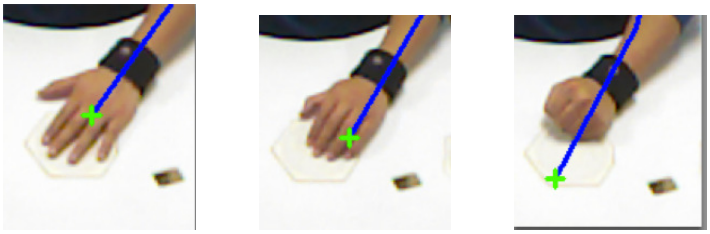


Figure 3.4: Illustration of using the OpenNI SDK skeleton tracking. In general, we observe lower accuracy in arm and end-point (wrist) tracking as compared to torso tracking which is more stable.



(a) Fully Open (b) Half Open (c) Close

Figure 3.5: Inaccurate end-point localization during articulations of the palm, as obtained from OpenNI SDK skeleton tracking. The green '+' marks show the estimated end-point position.

in feature space, which dramatically reduces the tracking complexity. Figure 3.6 shows the marker. We combine the color features extracting from the RGB image with the depth features from the Depth image.

Depth features significantly simplify the task of object detection. It enables the reconstruction of shape and appearance of real objects from the 2D projection plane. Depth value is a remarkable feature for representing objects since objects, foreground, and background have different distances from the camera. This greatly reduces the segmentation errors when the background cluster is similar to the foreground in feature space using only color camera. In addition, background models

Figure 3.6: A cyon marker is adhered on the top of the wristband.

based on color are often influenced by illumination changes, while depth image is not sensitive to light changes except extreme conditions and can work in complete darkness.

If the input RGB-D image frame is defined by P, the background subtraction is defined by a function $\mathbf{B} : P(x,y,z) \rightarrow I(x,y)$, where $I(x,y)$ represents the RGB image after background subtraction. The function $\mathbf{B}$ is achieved in two stages:

1. The first step aims to segment the potential participant's body regions from the background regions. Scene analysis algorithm ($\mathbf{L}$) in OpenNI labels the region of each participant as a unique integer and we then use (3.4) to select the participant's body regions. This operation is expressed as

$$P^{'}(x,y,z) = \begin{cases} P(x,y,z) & \text{if} \quad \mathbf{L}(P) = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

2. The second step aims to segment arm and torso parts. Based on our physical setup, the average depth values of arm pixels are smaller than ones of torso pixels when doing tasks. With the help of skeleton tracking algorithm, we could compute the average depth value of torso part $\overline{D}_{torso}$ by averaging the depth value of joints *SKEL_LEFT_SHOUDLER, SKEL_RIGHT_SHOUDLER, SKEL_TORSO*, and the average depth value of arm part $\overline{D}_{arm}$ by using the depth value of endpoint location in previous frame $\overline{D}_{endpoint}$. And the opera-

tion is expressed as

$$I(x,y) = \begin{cases} P^{'}(x,y) & \text{if} \quad P^{'}(z) > (\overline{D}_{torso} + \overline{D}_{arm})/2 \\ 0 & \text{otherwise} \end{cases} \tag{3.6}$$

We combine the Kalman filter and mean shift algorithm [32] to track the end-point. The complete tracking algorithm is presented below.

Given: The target model $\{\hat{q}_u\}_{u=1...m}$ and the location $\mathbf{y}_0$ in the previous frame and the kernel size $h_{prev} = (h_x, h_y)$:

1. Set the region of interest (ROI) at the location centering at $\mathbf{y}_0$, with the size $(2h_x, 2h_y)$.

2. Run the background subtraction process according to (3.5).

3. Time update (Predict) using Kalman filter. Update the location to $\mathbf{y}_1$ according to (2.10).

4. Initialize the location of the target in the current frame with $\mathbf{y}_0$, then compute $\{\hat{p}_u(\mathbf{y}_0)\}_{u=1...m}$, and compute the Bhattacharyya coefficient $\rho[\hat{\mathbf{p}}(\mathbf{y}), \hat{\mathbf{q}}]$ according to (2.3).

5. Derive the weights $\{w_i\}_{i=1...n_{h_{prev}}}$ from (2.5).

6. Find the next location of the target candidate according to (2.6).

7. Compute $\{\hat{p}_u(\mathbf{y}_2)\}_{u=1...m}$, and evaluate

$$\rho[\hat{\mathbf{p}}(\mathbf{y_2}), \hat{\mathbf{q}}] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(\mathbf{y}_2)\hat{q}_u}. \tag{3.7}$$

8. If $\| \mathbf{y}_2 - \mathbf{y}_0 \| > \varepsilon$, set $\mathbf{y}_0 \leftarrow \mathbf{y}_2$ and go to Step 5.

9. If $\rho[\hat{\mathbf{p}}(\mathbf{y_2}), \hat{\mathbf{q}}] < \delta$, set $\mathbf{y}_0 \leftarrow \mathbf{y}_1$. Otherwise, go to Step 10.

10. Measurement Update (Correct) using Kalman filter according to (2.12). Update $h_{cur} \leftarrow h_{prev}$. Stop.

Note RGB image is converted into HSV space, and only H and S channels are adopted. The color is quantified into 32×16 bins. The experiment results of the tracking accuracy and robustness will be provided in chapter 5.

Chapter 4

QUANTATIVE KINEMATIC EVALUATION ON TORSO COMPENSATION FOR IMPAIRED STROKE SURVIVORS

Stroke survivors usually use their torso to assist arm movements to compensate for their inadequacies in arm strength. It was recently suggested that excessive torso movement when reaching may affect their recovery of the 'normal' motor patterns of the arm [39], and torso-restraint method produced greater improvement in arm impairment [33]. Thus, analysis of a subject's compensatory movement is key to the evaluation of arm motor functionality.

The goal of the motion analysis module is twofold: 1) translate tracking results into kinematic features that represent patients' motor functionality during physical tasks. 2) translate kinematic features into quantitative kinematic evaluation, giving descriptive results for generating proper multimodal feedback. In response, we first introduce kinematic feature extraction process, and then describe how these features elicit multimodal feedback.

4.1 KINEMATIC FEATURES FOR TORSO MOVEMENT

Torso compensation, is usually found in the form of unacceptable levels of torso leaning forward, or torso twisting to the sides. These two kinds of compensatory movements are defined as including: 1) Leaning forward or backward, termed 'torso leaning'; and 2) twisting towards or away from the target, termed 'torso twisting'. Figure 4.1 illustrates the movements.

A variety of factors contribute to the complexity of the torso movement evaluation. Different physical tasks and target locations, as one crucial factor, are prescribed and give rise to different levels of compensation. For example, participants use more twisting when they are reaching a cone at midline than the one

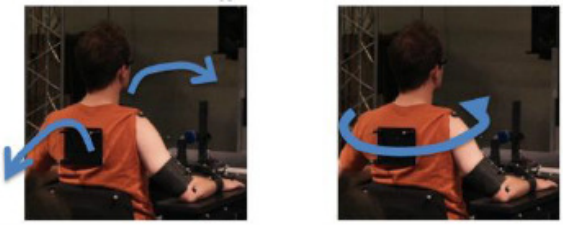


Figure 4.1: Illustration of two classes of compensatory movement that needs to be measured. (Left) Torso leaning (Right) Torso rotation. Details described in text.

at middle. Further, different subjects adopt different compensatory strategies. For example, some people perform significant leaning during initial movement, while others use leaning during the final phase of reaching a target.

Based on the idea of multi-layer feedback hierarchy, the feedback environments of HAMRR [6] are divided into three levels – real-time, post-trial, and post-set feedback. The dynamic feedback hierarchy calls for robust motion analysis strategies. Real-time feedback aims to give patients intuitive and immediate response on adjusting movement during the task, while the post-trial and post-set feedback focus on providing a comprehensive and reflective evaluation of the participant's movements that can be played back and help the participant self-assess their motor functionality and plan for improvement in future long-term therapy. As a result, we compute two kinds of features - real-time based features and trial based features - to generate different levels of feedback.

4.1.1 REAL-TIME BASED FEATURES

The first set of features are termed 'real-time features', as they are used to elicit multimodal feedback in real-time when the reaching action is performed.

Assume that torso segment is a plane on articulated skeleton body representation. All the movements related to the torso plane can be categorized into two major aspects: ${}^{r}R^{c}$ - Rotation from rest plane local coordinate system O_c to current

plane local coordinate system O_r; ${}^cR^a$ - Rotation from O_c to arm plane coordinate system O_a.

First, two angles are used to specifically measure the two compensatory movements described above:

- **L**eaning angle $\theta^L(t)$: Leaning angle measures how much a participant leans forward or backward. In the global coordinate system, leaning angle is the rotation angle between current torso plane and rest torso plane around x axis.
- **T**wisting angle $\theta^T(t)$: Twist angle measures how much patient twist towards or away from the target. In the global coordinate system, twisting angle is the rotation angle between the current torso plane and rest torso plane around y axis.

Below, we first focus on the first group of movements. Then we introduce how to calculate $\theta^L(t)$ and $\theta^T(t)$. If a point position is rotated by θ, it is equal to a $-\theta$ rotation of coordinate. $\theta^L(t)$ and $\theta^T(t)$ are both Euler joint angles from rotation matrix. Thus, they can be calculated by calculating the rotation matrix ${}^rR^c$ from the rest torso local coordinate system O_r to the current torso local coordinate system O_c. ${}^rR^c$ can be calculated by

$$ {}^rR^c = {}^rR^g({}^cR^g)^{-1}, \tag{4.1} $$

where ${}^rR^g$ is the rotation matrix from O_r to global coordinate system, ${}^cR^g$ is the rotation matrix from current torso local coordinate to global coordinate. We compute the rotation matrix ${}^rR^g$ according to (2.22). The torso local coordinate system is established by 3D coordinates of the three joints, which are *SKEL_LEFT_SHOUDLER, SKEL_RIGHT_SHOUDLER*, and *SKEL_TORSO* in global coordinate system. Then, we compute the rotation matrix ${}^cR^g$ in the same manner. According to [47], we can

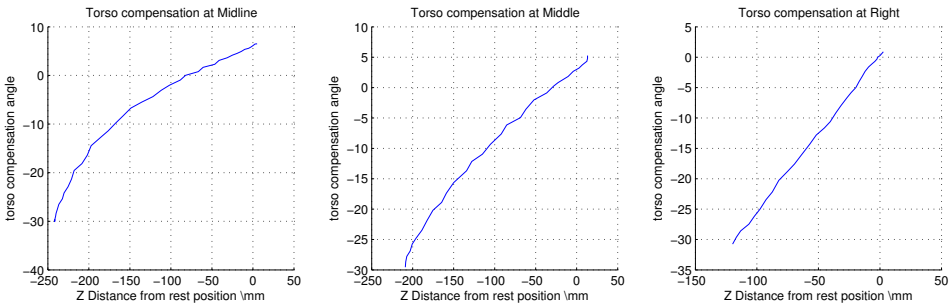


Figure 4.2: Illustration of correlation between torso leaning angles and the end-point distance in Z-axis away from the rest position

compute the Euler joint angles θ, ψ, ϕ from the rotation matrix ${}^{c}R^{g}$. As a result, we get the two angles by $\theta^{L}(t) = \phi, -90^{\circ} < \phi < 90^{\circ}$, $\theta^{T}(t) = \theta, -90^{\circ} < \theta < 90^{\circ}$.

A 'reference' movement is trained for each target. Figure 4.2 shows the correlation between torso leaning angles, and the end-point in Z axis away from the rest position.

In conclusion, they are approximately linear. The reference angles are described as

$$\theta^{L}_{ref}(z^{'}(t)) = z^{'}(\theta^{L}_{ref}(z^{'}_{tar}) - \theta^{L}_{ref}(z^{'}_{rest})) + \theta^{L}_{ref}(z^{'}_{rest}) \tag{4.2}$$

$$\theta^{T}_{ref}(z^{'}(t)) = z^{'}(\theta^{T}_{ref}(z^{'}_{tar}) - \theta^{T}_{ref}(z^{'}_{rest})) + \theta^{T}_{ref}(z^{'}_{rest}) \tag{4.3}$$

where $\theta^{L}_{ref}(z^{\star}(t)), \theta^{T}_{ref}(z^{'}(t))$ refer to the reference leaning and twisting angles at $z^{'}(t)$. $z^{'}(t)$ refer to the normalized distance from rest position in z axis of a rotated 2D coordinate O_r, which is generated by first projecting O_{base} in XZ plane and then rotating it so that the rest-to-target direction as the z axis. z_{tar}, z_{rest} refer to the z values of target and rest position in coordinate O_r. Thus, $z_{tar} = 1, z_{rest} = 0$.

4.1.2 TRIAL BASED FEATURES

The second set of features are termed 'trial features', which are computed post-trial. The trial features are dedicated to providing a comprehensive quality evaluation of movement and also helping the participant see the progress of repetitive tasks and

plan movement in future tasks. Based on real-time features, we proposed the following trial features, for evaluating the quality of compensatory movements: 1) mean leaning angle, $\overline{\theta}^L$; 2) mean twisting angle, $\overline{\theta}^T$; 3) max leaning angle difference, θ^L_{max}; 4) max twisting angle difference, θ^T_{max}; 5) standard deviation of leaning angle, σ_L; 6) standard deviation of twisting angle, σ_T; 7) maxOffsetX, $\Delta_{X,max}$; and 8) maxOffsetZ, $\Delta_{Z,max}$.

Where $\Delta_{X,max}$, $\Delta_{Z,max}$ are the offsets from rest positions to current endpoint positions in X and Z axes. $\overline{\theta}^L$, $\overline{\theta}^T$ are calculated by

$$\overline{\theta}^L = \frac{\sum_{t=t_{start},\ldots,t_{end}}^{N} \theta^L(t) K(z^{'}(t))}{N} \tag{4.4}$$

$$\overline{\theta}^T = \frac{\sum_{t=t_{start},\ldots,t_{end}}^{N} \theta^T(t) K(z^{'}(t))}{N} \tag{4.5}$$

where $K(z^{'}(t))$ is an exponential kernel which is added based on the fact that compensatory movements are most likely to be initiated either at initial or near the target. $K(z^{'}(t))$ is given by

$$K(z^{'}(t)) = e^{\alpha|z^{'}(t)-0.5|} \tag{4.6}$$

4.2 MAPPING FEATURES TO FEEDBACK

4.2.1 REAL-TIME FEEDBACK

In order to generate real-time feedback, a descriptive result needs to be provided on excessive leaning or twisting actions are detected and the overall compensation profile for each sample. We use normalized angle values to calculate the confidence scores to trigger feedback. The normalized angles are given by:

$$\hat{\theta^L}(t) = \frac{|\theta^L(t) - \theta^L_{ref}(z^{'}(t))|}{TH^L}, \tag{4.7}$$

$$\hat{\theta^T}(t) = \frac{|\theta^T(t) - \theta^T_{ref}(z^{'}(t))|}{TH^T}, \tag{4.8}$$

where TH^L and TH^T are thresholds. If the variance is larger than TH^L, TH^T , the normalized angles are set at 1. The real-time torso compensation score is given by

$$C^{TC}(t) = w^L \hat{\theta}^L(t) + w^T \hat{\theta}^T(t) \tag{4.9}$$

where w^L, w^T are two weights. The values of thresholds and weights are shown in the table.

4.2.2 POST TRIAL FEEDBACK

In order to get post trial feedback, confidence scores for the whole trial are calculated by applying off-the-shelf classification technologies. The feature selection process is explained on Section 4.1.2, and the classification results will be provided in Chapter 6.

Chapter 5

IMPLEMENTATIONS AND EXPERIMENT RESULTS

In experiments, a system is tested along two primary dimensions. One is to measure the tracking accuracy and robustness of the torso and end-point using Kinect. We first compare the torso and arm tracking accuracy and robustness between OpenNI and Kinect SDK. Then compare end-point tracking accuracy and robustness between our approach and results from OpenNI and Kinect SDK. Both of the evaluation consider the Opti-Track results as ground truth data. The other is the classification accuracy in measuring anomalies in torso compensation using Kinect.

Next, we first introduce the system setup, and then give the evaluation results for both the end-point tracking performance and also torso movement.

5.1 SYSTEM SETUP

Figure 5.1 shows the physical setup of HAMRR system [6]. The media center includes 1) a 27 inch iMac with 3.4GHz Intel i7 CPU, 20GB memory, and 320GB SSD hard drive; and 2) Two Bose Companion 2 speakers. They are utilized for computing, system GUI, and providing audio and visual feedback. Four Natural Point Opti-Track Infrared cameras and Kinect for Windows are mounted on the media center, supported by an aluminum frame.

A table is utilized to give support for the hand and arm during movements. The location of three predetermined target slots (midline, Ipsilateral Straight, and Ipsilateral Right) are designed according to the [5]. Different kinds of objects can be plugged in or removed using a button: a) Virtual and button objects - designed for reaching tasks; b) cone objects - designed for reaching-to-grasping tasks; and c) transport objects - designed for reaching-to-lifting-to-transporting tasks. The table also houses a contact switch rest position pad, ensuring the reaching task is

Figure 5.1: Physical setup of HAMRR system.

initiated from approximately the same location, and two capacitive touch buttons, for interaction between participants and media center. A chair covered with 1.5 inch square FSRs is applied for providing alternative torso information, especially for determining if the participant is in the rest position.

For the software setup, the main system control program runs in Mac OSX and the Kinect sensing program and Tracking Tools, which is a commercial software for Opti-Track both runs under Windows 7 in virtual machine. Parallels are employed to get the environments running simultaneously, and the cross-platform communication is achieved by Multicast. OpenNI v1.5.4.0, SensorKinect v0.93, and NITE v1.5.21 are installed for driving Kinect sensor, and OpenCV v2.3.1 is also applied in basic image processing functions.

Various experimental parameters, thresholds, and constants are shown in

Table 5.1: Threshold and parameters for various features computation

Para Names	Value
α	-0.5
TH^T	15°
TH^L	13°
w^T	0.4
w^L	0.6
δ	0.6

table 5.1.

5.2 TRACKING PERFORMANCE EVALUATION

Tracking evaluation is to compare end-point tracking accuracy and robustness between the approach and results from OpenNI. Video sequences of 320×240 pixels are recorded, and during the sequences, reaching, grasping, and lifting tasks to different targets with normal or abnormal movement are recorded as well. The target was initialized with a preset rectangle region which referred to the rest position of size 26×26. Figure 5.2illustrates the tracking robustness to partial occlusion, rotation.

We also computed the tracking errors during reaching movements to different targets. For each target, we computed the x-axis and z-axis error separately. We recorded the data of four groups, covering different kinds of possible movements to three objects. The four groups were normal reaching, normal reaching with torso leaning, normal grasping, and curved reaching. We captured over 3500 Kinect frames and also over 10000 Opti-Track frames. The data were synchronized using Timestamp. The data included 36 trials, with 12 trials for each objects. We computed the maximum and mean tracking errors for each trial in X and Z axis, and gave the result in figure 5.3.

The proposed end-point tracking approach showed promising results on

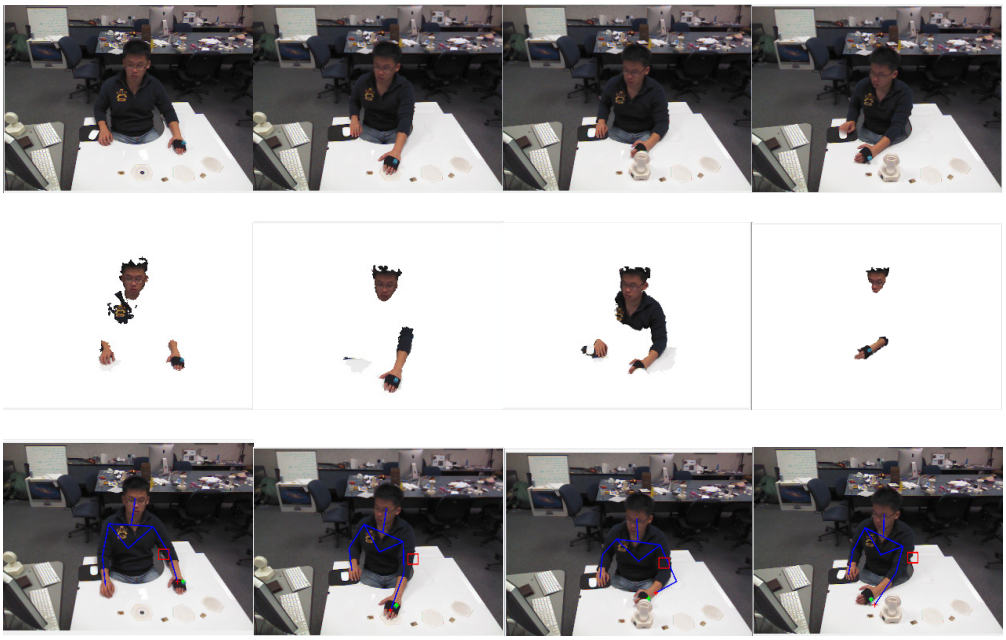


Figure 5.2: The top line shows the sequences of input RGB frames. The sequences show the tracking under different rotations, occlusions. The second line shows the segmentation results. The last line shows the tracking results. The green cross refers to the endpoint location obtained from our proposed approach, while the red cross refers to the result from OpenNI skeleton tracking.

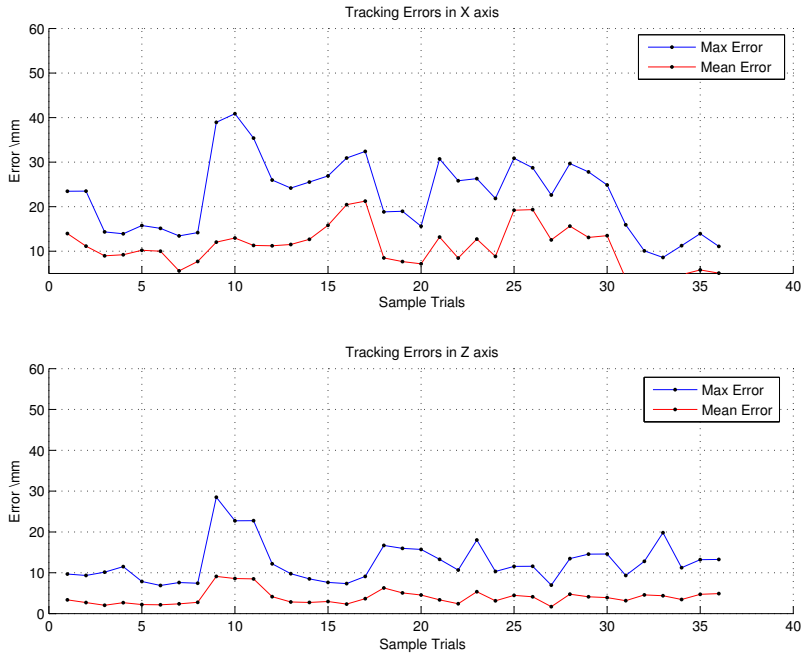


Figure 5.3: Tracking errors for end-point tracking approach.

tracking accuracy. In both the x-axis and z-axis, the mean error was under 2mm, which was proved to be practical in our application. Torso tracking showed solid accuracy and stability.

5.3 TORSO MOVEMENT EVALUATION

This evaluation was difficult to implement due to several reasons. The most important challenge lied at its extensive comparative testing with patients. The goal was to design a system that can be shipped to a patient's home for a long-term therapy. However, design decisions on motion capture (Optitrack vs. Kinect), had to be made without extensive data from use of the system by patients at the home. An extensive data of patients was obtained from a recently completed clinical study [11]. This data was used to construct the classifiers. However, this data was captured using an eight camera Optitrack system and a significant amount of body markers during supervised therapy. It is not yet clear how this data will be aligned with data from the simpler set up of the home system which is used without a therapist's supervision. The prior clinical trial did not use a Kinect, either. The home system is currently being deployed at patients' homes for a multisite pilot trial. Data from this trial will allow to further improve the classifiers and study more extensively the comparative performance of Kinect and marker based capture for the extraction of movement quality classifiers.

In this study, data was collected in controlled settings using simulated movements by experienced members of a research group. The acquired database of various reach movements for this experiment consists of 23 different sets, and each set contains several trials. The trials cover rehabilitation physical tasks such as grasping, reaching and lifting to different targets – midline, ipsilateral straight out, and ipsilateral at a right angle – which correspond to three different placements of the reaching target. Details of physical placement can be found in [5]. In the dataset,

we have a total of 134 trials, and for each trial the features proposed in section 4.1.2 are extracted.

We used Weka [24], to train and evaluate the classifiers in our experiments. We used Naive Bayes, nearest neighbors and support vector machines (SVM) as the classifiers to compare with. We use 10-fold cross validation to compare various classifiers. Classification accuracies for various choices of features and classifiers and individual tasks are shown in table 5.2. In table 5.3, we provide the overall confusion matrices for the various feature and classifier combinations.

Table 5.2: Results of cross-validation for classifying torso movements. Two types of movements 'Leaning' and 'Twisting' actions are classified into classes 'Normal' and 'Impaired'. Group I and Group II features are discussed in section 4.1.2.

Classifier	Features	Midline	Ipsilateral Straight	Ipsilateral Right	Total
Torso Leaning					
Naive Bayes	Group I	72%	100%	80%	87.31%
	Group II	72%	100%	82%	88.06%
1-NN	Group I	96%	83.01%	86%	86.56%
	Group II	100%	98.30%	98%	98.51%
SVM	Group I	78%	88.13%	96%	88.81%
	Group II	78%	88.13%	96%	88.81%
Torso Twisting					
Naive Bayes	Group I	84%	93.20%	84%	88.06%
	Group II	88%	93.20%	84%	88.81%
1-NN	Group I	88%	83.01%	86%	87.31%
	Group II	88%	93.20%	90%	91.05%
SVM	Group I	92%	88.13%	84%	87.31%
	Group II	96%	88.13%	84%	88.06%

It is arguably obvious that classification rates are stable across classifiers. However, an improvement was found in the result - when the extra end-point features i.e. $\{\Delta_{X,max}, \Delta_{Z,max}\}$ were added to Group I features - becoming Group II features. In the absence of accurate capture of the end-point, we would have relied solely on Group I features, which is still sufficiently reliable. These results indicate that the quality of data from Kinect combined with carefully crafted features and

Table 5.3: Confusion matrices for classifying torso movements into 'Normal' and 'Impaired'. Group I and Group II features are discussed in section 4.1.2.

Leaning Action						
	Naive Bayes		Nearest Neighbor		SVM	
Group I	43	9	40	12	38	14
	8	74	6	76	1	81
Group II	43	9	52	0	38	14
	7	75	2	80	1	81
Twisting Action						
	Naive Bayes		Nearest Neighbor		SVM	
Group I	75	7	74	8	77	5
	9	43	9	43	12	40
Group II	76	6	77	5	78	4
	9	43	7	45	12	40

classifiers is sufficient for torso compensation analysis of the home-based rehabilitation system.

Chapter 6

CONCLUSION AND FUTURE WORK

In this thesis, we presented a motion/activity analysis for a home-based stroke rehab system, with a detailed analysis of the pros and cons of choosing a high-end motion capture technology (Opti-Track) versus an inexpensive one (Kinect). While it is possible to obtain reasonable tracking accuracy of various joints in terms of tracking errors, it does not necessarily translate to robust activity classification for measuring impairment. Although torso movement classifiers were able to produce robust results using the Kinect, the end-point tracking did not show satisfactory confidence score for robust end-point kinematics. There, thus, is need to combine the use of a four-camera Opti-Track setup with a single marker on the wrist for end-point tracking, with the use of low-cost depth camera Kinect for torso tracking.

This research points to several interesting directions of future work. From a sensor fusion perspective, one can explore the utility of multiple Kinect sensors and study its effect on obtaining high fidelity tracking results. Accuracies of such multi-Kinect systems and their efficacy for rehabilitation systems are still unknown. For the computer vision and machine learning communities, this application area raises several interesting questions related to robust features and classifiers for movement analysis. Significant research in computer vision has focused on activity and gesture recognition and not much on measures of 'quality' of the movement. While this problem is traditionally addressed in the bio-mechanics community, the tools developed in that community are based on precise clinical measurements of motion or expensive equipment, such as EMG and pressure sensors. Thus, one needs to rely on large datasets and advanced feature selection and machine learning tools to devise quality measures. This can form the basis of several interesting research questions in the future.

BIBLOGRAPHY

[1] Kinect. http://en.wikipedia.org/wiki/Kinect. Online.

[2] Kinect for Windows Sensor Components and Specifications. http://msdn.microsoft.com/en-us/library/jj131033.aspx. Online; accessed 02/11/2012.

[3] Three Dimensional Space. http://en.wikipedia.org/wiki/Three-dimensional_space. Online; accessed 11/09/2012.

[4] What you need to know about the Kinect for Xbox 360? http://www.gizmowatch.com/entry/what-you-need-to-know-about-the-kinect-for-xbox-360/. Online; accessed 02/11/2012.

[5] Suneeth Attygalle, Margaret Duff, Thanassis Rikakis, and Jiping He. Low-cost, at-home assessment system with Wii Remote based motion capture. In *Virtual Rehabilitation*, pages 168–174, Aug 2008.

[6] M. Baran, N. Lehrer, D. Siwiak, Y. Chen, M. Duff, T. Ingalls, and T. Rikakis. Design of a home-based adaptive mixed reality rehabilitation system for stroke survivors. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pages 7602–7605. IEEE, 2011.

[7] G. Bradski and A. Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. O'Reilly Media, Incorporated, 2008.

[8] C.Y. Chang, B. Lange, M. Zhang, P. Requejo, N. Somboon, A. Sawchuk, and A. Rizzo. Towards pervasive physical rehabilitation using microsoft kinect. In *6th International Conference on Pervasive Computing Technologies for Healthcare*, 2012.

[9] A. Chen, M. Zhu, Y. Wang, and C. Xue. Mean shift tracking combining sift. In *Signal Processing, 2008. ICSP 2008. 9th International Conference on*, pages 1532–1535. IEEE, 2008.

[10] Y. Chen, M. Baran, H. Sundaram, and T. Rikakis. A low cost, adaptive mixed reality system for home-based stroke rehabilitation. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pages 1827–1830. IEEE, 2011.

[11] Y. Chen, M. Duff, N. Lehrer, S. M. Liu, P. Blake, S. L. Wolf, H. Sundaram, and T. Rikakis. A novel adaptive mixed reality system for stroke rehabilitation:

Principles, proof of concept, and preliminary application in 2 patients. *Topics in stroke rehabilitation*, 18(3):212–230, -01 2011.

[12] Y. Chen, N. Lehrer, H. Sundaram, and T. Rikakis. Adaptive mixed reality stroke rehabilitation: system architecture and evaluation metrics. In *Proceedings of the first annual ACM SIGMM conference on Multimedia systems*, pages 293–304, 2010.

[13] Yinpeng Chen. *Constraint-aware computational adaptation framework to support realtime multimedia applications*. PhD thesis, Arizona State University, Tempe, AZ, USA, 2009. AAI3371193.

[14] Yinpeng Chen, Weiwei Xu, Richard Isaac Wallis, Hari Sundaram, Thanassis Rikakis, Todd Ingalls, Loren Olson, and Jiping He. A real-time, multimodal biofeedback system for stroke patient rehabilitation. In *ACM Multimedia*, pages 501–502, 2006.

[15] R.T. Collins. Mean-shift blob tracking through scale space. In *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, volume 2, pages II–234. IEEE, 2003.

[16] Dorin Comaniciu, Visvanathan Ramesh, and Peter Meer. Kernel-based object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.*, 25(5):564–575, 2003.

[17] P.J. Costello. *Health and safety issues associated with virtual reality: a review of current literature*. Citeseer, 1997.

[18] M. de Niet, J. B. Bussmann, G. M. Ribbers, and H. J. Stam. The stroke upper-limb activity monitor: Its sensitivity to measure hemiplegic upper-limb activity during daily life. *Archives of Physical Medicine and Rehabilitation*, (88):1121–1126, 2007.

[19] A. W. Dromerick, C. E. Lang, R. Birkenmeier, M. G. Hahn, S. A. Sahrmann, and D. F. Edwards. Relationships between upper-limb functional limitation and self-reported disability 3 months after stroke. *Journal of Rehabilitation Research and Development*, (43):401–408, 2006.

[20] M. Duff, Y. Chen, S. Attygalle, J. Herman, H. Sundaram, G. Qian, J. He, and T. Rikakis. An adaptive mixed reality training system for stroke rehabilitation. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 18(5):531–541, 2010.

[21] H.C. Fischer, K. Stubblefield, T. Kline, X. Luo, R.V. Kenyon, and D.G. Kamper. Hand rehabilitation following stroke: a pilot study of assisted finger extension training in a virtual environment. *Topics in Stroke Rehabilitation*, 14(1):1–12, 2007.

[22] D.A. Forsyth and J. Ponce. *Computer vision: a modern approach*. Prentice Hall Professional Technical Reference, 2002.

[23] I. Guyon, V. Athitsos, P. Jangyodsuk, B. Hamner, and H.J. Escalante. Chalearn gesture challenge: Design and first results. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2012 IEEE Computer Society Conference on*, pages 1–6. IEEE, 2012.

[24] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *SIGKDD Explorations Newsletter*, 11(1):10–18, Nov 2009.

[25] A. Henderson, N. Korner-Bitensky, and M. Levin. Virtual reality in stroke rehabilitation: a systematic review of its effectiveness for upper limb motor recovery. *Topics in stroke rehabilitation*, 14(2):52–61, 2007.

[26] H.T. Hendricks, J. van Limbeek, A.C. Geurts, M.J. Zwarts, et al. Motor recovery after stroke: a systematic review of the literature. *Archives of physical medicine and rehabilitation*, 83(11):1629–1637, 2002.

[27] S. Kean, J. Hall, and P. Perry. *Meet the Kinect: An Introduction to Programming Natural User Interfaces*. Apress, 2011.

[28] G. Kwakkel, B. Kollen, and E. Lindeman. Understanding the pattern of functional recovery after stroke: Facts and Theories. *Restorative Neurology and Neuroscience*, 22(3-5):281–299, 2004.

[29] N. Lehrer, Y. Chen, M. Duff, S.L. Wolf, and T. Rikakis. Exploring the bases for a mixed reality stroke rehabilitation system, part ii: Design of interactive feedback for upper limb rehabilitation. *Journal of neuroengineering and rehabilitation*, 8(1):54, 2011.

[30] Tommer Leyvand, Casey Meekhof, Yichen Wei, Jian Sun 0001, and Baining Guo. Kinect identity: Technology and experience. *IEEE Computer*, 44(4):94–96, 2011.

[31] P. Li. An adaptive binning color model for mean shift tracking. *Circuits and Systems for Video Technology, IEEE Transactions on*, 18(9):1293–1299, 2008.

[32] P. Meer. Kernel-based object tracking. *IEEE Transactions on pattern analysis and machine intelligence*, 25(5), 2003.

[33] S.M. Michaelsen, R. Dannenbaum, and M.F. Levin. Task-specific training with trunk restraint on arm recovery in stroke randomized control trial. *Stroke*, 37(1):186–192, 2006.

[34] A. Mirelman. Effects of virtual reality training on gait biomechanics of individuals post-stroke. *Gait and posture*, 31(4):433, 2010. doi: pmid:.

[35] Richard A. Newcombe, Shahram Izadi, Otmar Hilliges, David Molyneaux, David Kim, Andrew J. Davison, Pushmeet Kohli, Jamie Shotton, Steve Hodges, and Andrew Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Proceedings of the 2011 10th IEEE International Symposium on Mixed and Augmented Reality*, ISMAR '11, pages 127–136, Washington, DC, USA, 2011. IEEE Computer Society.

[36] B. Ni, G. Wang, and P. Moulin. Rgbd-hudaact: A color-depth video database for human daily activity recognition. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 1147–1153. IEEE, 2011.

[37] S. Obdrzalek, G. Kurillo, F. Ofli, R. Bajcsy, E. Seto, H. Jimison, and M. Pavel. Accuracy and robustness of kinect pose estimation in the context of coaching of elderly population. In *International Conference of the Engineering in Medicine and Biology Society (EMBC)*, Aug 2012.

[38] I. Oikonomidis. Efficient model-based 3d tracking of hand articulations using kinect. *Computer Vision and Image Understanding*, 108(1-2):52, 2011.

[39] A. Roby-Brami, A. Feydy, M. Combeaud, EV Biryukova, B. Bussel, and MF Levin. Motor compensation and recovery for reaching in stroke patients. *Acta neurologica scandinavica*, 107(5):369–381, 2003.

[40] V.L. Roger, A.S. Go, D.M. Lloyd-Jones, E.J. Benjamin, J.D. Berry, W.B. Borden, D.M. Bravata, S. Dai, E.S. Ford, C.S. Fox, et al. Heart disease and stroke statisticsÑ2012 update a report from the american heart association. *Circulation*, 125(1):e2–e220, 2012.

[41] G. G. Saposnik. Effectiveness of virtual reality using wii gaming technology in stroke rehabilitation: A pilot randomized clinical trial and proof of principle. *Stroke (1970)*, 41(7):1477–1484, -07 2010. doi:10.1161/STROKEAHA.110.584979 pmid:.

[42] G. G. Saposnik. Virtual reality in stroke rehabilitation: A meta-analysis and implications for clinicians. *Stroke (1970)*, 42(5):1380–1386, -05 2011. doi:10.1161/STROKEAHA.110.605451 pmid:.

[43] M. T. Schultheis and A. A. Rizzo. The application of virtual reality technology in rehabilitation. *Rehabilitation psychology*, 46(3):296–311, 2001.

[44] C. Shan, T. Tan, and Y. Wei. Real-time hand tracking using a mean shift embedded particle filter. *Pattern Recognition*, 40(7):1958–1970, 2007.

[45] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from single depth images. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1297–1304, Washington, DC, USA, 2011. IEEE Computer Society.

[46] Diana Siwiak, Nicole Lehrer, Michael Baran, Yinpeng Chen, Margaret Duff, Todd Ingalls, and Thanassis Rikakis. A home-based adaptive mixed reality rehabilitation system. In *ACM Multimedia*, pages 785–786, 2011.

[47] G. G. Slabaugh. Computing euler angles from a rotation matrix. *Tech Report*, 1999.

[48] J. Sung, C. Ponce, B. Selman, and A. Saxena. Human activity detection from RGBD images. In *AAAI workshop on Pattern, Activity and Intent Recognition (PAIR)*, 2011.

[49] E. Taub, G. Uswatte, and R. Pidikiti. Constraint-induced movement therapy: A new family of techniques with broad application to physical rehabilitation – A clinical review. *Journal of Rehabilitation Research and Development*, (36):237–251, 1999.

[50] J. Tu, H. Tao, and T. Huang. Online updating appearance generative mixture model for meanshift tracking. *Machine vision and applications*, 20(3):163–173, 2009.

[51] J. Wang, B. Thiesson, Y. Xu, and M. Cohen. Image and video segmentation by anisotropic kernel mean shift. *Computer Vision-ECCV 2004*, pages 238–249, 2004.

[52] G. Welch and G. Bishop. An introduction to the kalman filter.

[53] S. S. L. Wolf. Effect of constraint-induced movement therapy on upper extremity function 3 to 9 months after stroke: The excite randomized clinical trial. *JAMA : the journal of the American Medical Association*, 296(17):2095–2104, -11 2003. doi:10.1001/jama.296.17.2095 pmid:.

[54] D. Xu, Y. Wang, and J. An. Applying a new spatial color histogram in mean-shift based tracking algorithm. In *Image and Vision Computing New Zealand*, 2005.

[55] C. Yang, R. Duraiswami, and L. Davis. Efficient mean-shift tracking via a new similarity measure. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 176–183. IEEE, 2005.

[56] G. Yavuzer, R. Selles, N. Sezer, S. Sütbeyaz, J.B. Bussmann, F. Köseoğlu, M.B. Atay, H.J. Stam, et al. Mirror therapy improves hand function in subacute stroke: a randomized controlled trial. *Archives of physical medicine and rehabilitation*, 89(3):393–398, 2008.

[57] A. Yilmaz, O. Javed, and M. Shah. Object tracking: A survey. *Acm Computing Surveys (CSUR)*, 38(4):13, 2006.

[58] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, November 2000.

[59] H. Zhou, Y. Yuan, and C. Shi. Object tracking using sift features and mean shift. *Computer Vision and Image Understanding*, 113(3):345–352, 2009.

[60] Zhiwei Zhu, Qiang Ji, Kikuo Fujimura, and Kuangchih Lee. Combining kalman filtering and mean shift for real time eye tracking under active ir illumination. In *IEEE Intl. Conference on Pattern Recognition (ICPR) (4)*, page 318, 2002.