Assessment of Human Kinematic Performance with Non-contact Measurements for Tele-rehabilitation

by

Saiyi Li

Submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

Deakin University

August, 2015
I am the author of the thesis entitled

Assessment of Human Kinematic Performance with Non-contact Measurements for Tele-rehabilitation

submitted for the degree of Doctor of Philosophy

This thesis may be made available for consultation, loan and limited copying in accordance with the Copyright Act 1968.

'I certify that I am the student named below and that the information provided in the form is correct'

Full Name: ..........Saiyi Li.................................................................
(Please Print)

Signed: Signature Redacted by Library

Date: .................25/11/2015.................................................................
I certify the following about the thesis entitled (10 word maximum)

Assessment of Human Kinematic Performance with Non-contact Measurements for Tele-rehabilitation

submitted for the degree of Doctor of Philosophy

a. I am the creator of all or part of the whole work(s) (including content and layout) and that where reference is made to the work of others, due acknowledgment is given.

b. The work(s) are not in any way a violation or infringement of any copyright, trademark, patent, or other rights whatsoever of any person.

c. That if the work(s) have been commissioned, sponsored or supported by any organisation, I have fulfilled all of the obligations required by such contract or agreement.

I also certify that any material in the thesis which has been accepted for a degree or diploma by any university or institution is identified in the text.

'I certify that I am the student named below and that the information provided in the form is correct'

Full Name: ..................Saiyi Li.................................................................
(Please Print)

Signed: .................................................................

Date: ..................25/11/2015.................................................................

Deakin University CRICOS Provider Code 00113B
Acknowledgements

First and foremost, I would like to express my sincerest gratitude to my supervisor, Associate Professor Dr. Pubudu Pathirana. Being an excellent advisor, his patience, motivation, enthusiasm and more importantly, his immense knowledge, support me throughout the past few years. In addition to that, he also is an admirable mentor and a good friend. I have learnt a lot from him, not only about how to be a researcher, but also about how to be a well-rounded person. His supervision during my time at Deakin University is invaluable, without which, I have never been able to finish my dissertation.

Apart from my supervisor, my thanks also goes to Professor Terry Caelli. His ideas and feedbacks inspired me and encouraged me to have more investigation in this important research field.

I would also like to thank NICTA for the financial support and giving me this opportunity to be a PHD student.

Besides, I thank Professor Mario Ferraro, Professor Mary P. Galea, Associate Professor Goetz Ottmann and Professor Fary Khan for their help and constructive feedbacks on my papers.

Furthermore, thanks all the members of the Deakin Network and Sensing Group and Deakin staff for sharing me valuable experience of both study and life.

Last but not the least, I thank my parents for supporting me spiritually and materially during my study in Australia, as well as throughout my life. At the same time, I would like to thank my wife who dedicated her time to take care of my life and cheer me up when I was depressed. This dissertation is dedicated for them.
Publications

- Published: Saiyi Li; Ferraro, M.; Caelli, T.; Pathirana, P.N., "A Syntactic Two-Component Encoding Model for the Trajectories of Human Actions," Biomedical and Health Informatics, IEEE Journal of , vol.18, no.6, pp.1903,1914, Nov. 2014


- Accepted: Hai Trieu Pham, Pubudu N. Pathirana, Yonggwan Won and Saiyi Li,"A Summative Scoring System for Evaluation of Human Kinematic Performance", Biomedical Signal Processing and Control

- Accepted: Pubudu N. Pathirana, Saiyi Li, Hieu M. Trinh and Aruna Seneviratna, "Robust real-time Bio-Kinematic Movement tracking using Multiple Kinects© for Tele-rehabilitation," Industrial Electronics, IEEE Transactions on


Published: Saiyi Li; Pathirana, P.N., "Cloud-based non-invasive tele-rehabilitation exercise monitoring," Biomedical Engineering and Sciences (IECBES), 2014 IEEE Conference on, vol., no., pp.385,390, 8-10 Dec. 2014


Published: Lee, Y.S.; Pathirana, P.N.; Caelli, T.; Saiyi Li, "Further applications of Doppler radar for non-contact respiratory assessment," Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE, vol., no., pp.3833,3836, 3-7 July 2013

Accepted: Saiyi Li, Pubudu N. Pathirana and Mary P. Galea., "Motion trajectory analysis for evaluating functional upper extremity tasks in daily living: a pilot study," Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE

Table of Contents

Acknowledgements i
Publications ii
Table of Contents iv
Abstract vii

1 Introduction 1
1.1 Motivation ................................ 1
1.2 Background ................................ 2
1.3 Musculoskeletal Injuries and Neurological Movement Disorders ... 6
  1.3.1 Musculoskeletal injuries .................... 6
  1.3.2 Movement disorders ....................... 6
1.4 Sensors in Tele-rehabilitation ..................... 10
  1.4.1 Kinect.............................. 11
  1.4.2 RGB camera and microphone ................. 15
  1.4.3 Inertial measurement unit (IMU) ............. 17
  1.4.4 Summary and challenges .................... 20
1.5 Human Motion Encoding in Tele-rehabilitation ....... 21
  1.5.1 Human motion encoders in action recognition .... 21
  1.5.2 Human motion encoders in physical tele-rehabilitation ... 22
  1.5.3 Summary and challenges .................... 26
1.6 Patients’ Performance Evaluation ................... 27
  1.6.1 Questionnaire based assessment scales ......... 27
  1.6.2 Automated kinematic performance assessment ... 28
  1.6.3 Summary and challenges .................... 29
1.7 Contributions ................................ 30
1.8 Thesis Outline ................................ 31

2 Multi-Kinect Skeleton Fusion 33
2.1 Introduction ................................ 33
2.2 Linear Model of Human Motion Multi-Kinect System .......... 37
  2.2.1 Rotation and translation of the Kinect© .......... 40
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.3</td>
<td>Model-based State Estimation</td>
<td>40</td>
</tr>
<tr>
<td>2.3.1</td>
<td>Kalman filter (KF)</td>
<td>41</td>
</tr>
<tr>
<td>2.3.2</td>
<td>Particle filter (PF)</td>
<td>42</td>
</tr>
<tr>
<td>2.3.3</td>
<td>Robust linear filtering (RLF)</td>
<td>44</td>
</tr>
<tr>
<td>2.4</td>
<td>Fusion of Information</td>
<td>48</td>
</tr>
<tr>
<td>2.5</td>
<td>Mitigation of Occlusions and Optimised Positioning</td>
<td>48</td>
</tr>
<tr>
<td>2.5.1</td>
<td>Occlusions and incomplete information</td>
<td>49</td>
</tr>
<tr>
<td>2.5.2</td>
<td>Kinect® optimal positioning</td>
<td>50</td>
</tr>
<tr>
<td>2.6</td>
<td>Computer Simulations and Hardware Implementation</td>
<td>52</td>
</tr>
<tr>
<td>2.6.1</td>
<td>Computer simulations</td>
<td>54</td>
</tr>
<tr>
<td>2.6.2</td>
<td>Hardware experiments</td>
<td>58</td>
</tr>
<tr>
<td>2.6.3</td>
<td>Discussion</td>
<td>64</td>
</tr>
<tr>
<td>2.6.4</td>
<td>Application in tele-rehabilitation</td>
<td>65</td>
</tr>
<tr>
<td>2.7</td>
<td>Summary</td>
<td>67</td>
</tr>
<tr>
<td>3</td>
<td>Human Motion Trajectory Encoder</td>
<td>69</td>
</tr>
<tr>
<td>3.1</td>
<td>Introduction</td>
<td>69</td>
</tr>
<tr>
<td>3.2</td>
<td>The Two-component Encoder Theory</td>
<td>72</td>
</tr>
<tr>
<td>3.3</td>
<td>Encoding Methods</td>
<td>74</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Temporal index</td>
<td>74</td>
</tr>
<tr>
<td>3.3.2</td>
<td>Spatial index</td>
<td>75</td>
</tr>
<tr>
<td>3.4</td>
<td>Dealing with Noise</td>
<td>77</td>
</tr>
<tr>
<td>3.4.1</td>
<td>Least-square Gaussian filter</td>
<td>77</td>
</tr>
<tr>
<td>3.4.2</td>
<td>The Savitzky-Golay filter</td>
<td>78</td>
</tr>
<tr>
<td>3.4.3</td>
<td>Kalman filter</td>
<td>79</td>
</tr>
<tr>
<td>3.5</td>
<td>Complex Motion Decomposition using Switching Continuous Hidden Markov Models</td>
<td>80</td>
</tr>
<tr>
<td>3.5.1</td>
<td>Building model for atomic motions</td>
<td>80</td>
</tr>
<tr>
<td>3.5.2</td>
<td>Complex motion decomposition</td>
<td>81</td>
</tr>
<tr>
<td>3.6</td>
<td>Canonical Actions and the Action Alphabet</td>
<td>82</td>
</tr>
<tr>
<td>3.7</td>
<td>Experiments and Results</td>
<td>84</td>
</tr>
<tr>
<td>3.7.1</td>
<td>Comparison between Savitzky-Golay filter and least-squares Gaussian filter</td>
<td>85</td>
</tr>
<tr>
<td>3.7.2</td>
<td>The computability of the two-component model</td>
<td>86</td>
</tr>
<tr>
<td>3.7.3</td>
<td>Comparisons between the combinations of various indexing schemes and filtering approaches</td>
<td>91</td>
</tr>
<tr>
<td>3.7.4</td>
<td>Motion decomposition using switching continuous hidden Markov model (SCHMM)</td>
<td>92</td>
</tr>
<tr>
<td>3.8</td>
<td>Summary</td>
<td>95</td>
</tr>
<tr>
<td>4</td>
<td>Kinematic Performance Evaluation</td>
<td>98</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>98</td>
</tr>
<tr>
<td>4.2</td>
<td>Methodology</td>
<td>101</td>
</tr>
</tbody>
</table>
Abstract

This thesis proposes approaches to address some open questions in developing an automated human kinematic performance assessment method for physical tele-rehabilitation using optoelectronic bio-kinematic motion capture devices (OBM-CDs). Challenges relating to achieving this goal can be classified into three categories: human motion representation and decomposition, improving tracking accuracy and performance assessment using rehabilitation exercises and daily living.

These three challenges are investigated in this thesis by applying the following methods. Firstly, robust linear filtering is proposed to fuse multiple OBMCDs improving the motion capture accuracy, while range of view of the multi-sensor system can be extended by reducing occlusions. Secondly, a two-component encoding model, including a shape and a dynamic model, is introduced to represent motion trajectories in sensor (joint) level. By using the shape model, complex motion trajectories can be decomposed into pre-defined atomic ones. Further, the approximate entropy of the shape and dynamic model of motion trajectories is utilised to assess the smoothness of these trajectories. This evaluation is then used with motion duration to primarily evaluate the kinematic performance of “patients with dyskinesia” in doing functional upper extremity tasks in their daily living.

Eventually, all these researches led to a prototype of mobile cloud-based physical tele-rehabilitation system for patients’ motion monitoring and evaluation, which can be deployed in the cloud while accessed from mobile and non-mobile devices. For mobile devices, a multi-level human motion encoding scheme is proposed to extend the duration of these devices so patients can access the rehabilitation service for a longer period of time.

Preliminary experiments conducted show that the approaches proposed in this thesis contribute to solving these challenges.
Chapter 1
Introduction

1.1 Motivation

The increasing number of aging population has been observed in recent decades. According to a study conducted by Australian government[1], the percentage of people with age over 65 has increased from 8% in 1970 to 13% in 2001, which will keep growing in the next 40 years. In addition, it is obvious that the prevalence of various movement disorders increases with age [2]. As a result, in the foreseeable future, rehabilitation services will be highly in demand. In recent years, tele-rehabilitation services have been considered to tackle some inherent issues (refer to Sec. 1.2) in traditional clinic-based rehabilitation services although assessing the performance of patients in tel-rehabilitation exercise sessions remains a problem. Motivated by this issue, in this dissertation, a number of open questions are investigated preliminarily to build an automated assessment tool for tele-rehabilitation by using non-contact measurements. To achieve this goal, this dissertation introduces novel approaches to fuse multiple OBMCDs to improve tracking accuracy and robustness, extract features and perform automated human kinematic assessment. Eventually, the design of a physical tele-rehabilitation system with an energy saving scheme is introduced.
1.2 Background

Human locomotor system is a complex system composing of two main components, namely a passive and an active system. The former further consists of skeleton (approximately 200 individual bones) and joints, while the latter includes muscle, tendons and so on[3]. A pictorial description of these two components is shown in Fig. 1.1.

Components in the active locomotor system are unlikely to perform voluntary movements without the control of our powerful brain. According to [4], one argument regarding the relationship between the neural system and human motions is that the former makes programs for the latter. Therefore, it is obvious that the deterioration in the neural system (shown in Fig. 1.2) is highly likely to negatively affect the performance in the locomotor system.

Such complex locomotor and neural systems are frequently influenced by various injuries and disorders. As a result, a number of approaches have been explored to
treat people suffering from conditions leading to abnormal movements and even
disabilities. Among these methods, physical rehabilitation is commonly utilised
to assist patients in recovery and regaining abilities to perform Activities of Daily
Living (ADLs).

Defined as “the treatment of disease, injury, or deformity by physical methods
such as massage, heat treatment, and exercise rather than by drugs or surgery”
[5], physiotherapy (also known as physical therapy) has been applied in clinics
throughout the therapeutic history. A number of therapeutic procedures are in-
cluded in physical therapy, such as mechanotherapy, hydrotherapy, balneotherapy
and so on[6], among which, mechanotherapy was documented as early as 1840s. In
recent decades, physical therapy has been applied extensively for various muscu-
loskeletal injuries and neurological movement disorders [7, 8]. The detailed examples
can be found in Sec. 1.3.

Although traditional physical therapy has shown its effectiveness for the reha-
bilitation of physical functions of patients with movement disorders[9], a series of
drawbacks can also be observed[10, 11]. They are summarised as follows.
• These rehabilitation programs are “boring” and patients are discouraged by these repetitive exercises;

• Computerised sensing techniques are not involved in these programs, which may lead to insufficient interpretation of observed data;

• One-to-one form of delivering rehabilitation services makes conventional rehabilitation very inefficient and costly;

• Costly equipment is required by traditional rehabilitation therapies;

• Insufficient funding for rehabilitation services results in the access to these services unaffordable;

• Workforce in rehabilitation field is inadequate in number;

• Rehabilitation centres are usually distributed in urban area, while a large number of people needing rehabilitation services live in rural regions;

In light of the above, in 1998, the term “tele-rehabilitation” was “first raised” in a scientific article, attempting to overcome the shortages in conventional rehabilitation services [12]. Co-existing with the opportunities for tele-rehabilitation, such as economy of scale, interactive and motivation, reduced healthcare cost, patients’ privacy prevention and so on [10], a number of challenges can be seen from the engineering point of view as well. These challenges are also closely relating to the aim of this thesis.

First of all, developing affordable, high-quality and robust hardware [13] is critical to capture human motions for further analysis. A high quality and effective hardware may provide more accurate monitoring of the movements, thereby offering better feedbacks to patients to correct their movements and valuable information to therapists to make further treatment decisions. In addition, similar to the fourth
issue in traditional rehabilitation mentioned above, costly equipment used in tele-
rehabilitation services may prevent patients with low economic status from accessing
them. Thus, the development of affordable devices is critical for the improvement
of tele-rehabilitation services.

Secondly, an advanced approach in representing human movement data [14] is
important after capturing human motions. As is pointed out by [15], one difficulty
in tele-rehabilitation is how to reduce information collected from sensors, thereby
producing meaningful results to therapists. Therefore, an emerging challenge is
identifying the features that can be utilised to represent human motions so that
their rehabilitation-related details can be well preserved while other irrelevant in-
formation can be reduced as much as possible.

Thirdly, developing an outcomes measurement scheme to quantitatively and
objectively represent the performance of patients accessing tele-rehabilitation ser-
ices is also a challenge [16]. As mentioned in [17], one of the goals of RERC
is to develop assessment tools to monitor the progress of patients accessing tele-
rehabilitation services. These tools should not only provide feedback to stimulate
the patients to perform exercises more effectively, but also provide therapists with
general information about their patients in terms of functional rehabilitation.

Last but not the least, enabling patients to access physical tele-rehabilitation
services regardless of their location and time is a challenge[16]. Since one purpose of
rehabilitation services is to assist the patients to regain the ability to perform ADLs,
-enabling them to perform rehabilitation exercises in their most familiar and natural
environment is very important. Due to the different preferences between patients,
developing tele-rehabilitation services that can be run on mobile devices is extremely
useful. By doing so, patients’ kinematic performance in tele-rehabilitation exercises
and daily living can be assessed pervasively.
Because of the importance of tele-rehabilitation, as well as the need for automation of these tools, a significant amount of effort has been made to improve the physical tele-rehabilitation, which is discussed in Sec. 1.4, 1.5 and 1.6.

1.3 Musculoskeletal Injuries and Neurological Movement Disorders

Being one of the major areas where physical rehabilitation is applied, movement disorders associated with various injuries and conditions that lead to abnormal movements and disabilities. There are a number of causes resulting in abnormal movements, which can be classified into two main categories, including musculoskeletal injuries and neurological movement disorders. In this section, some examples in these two categories are briefly discussed.

1.3.1 Musculoskeletal injuries

These injuries are normally observed in joints with certain degree of movements, such as shoulder, elbow and wrists in upper extremities, hips, knees and ankles in lower ones, which may eventually lead to abnormal movements or disabilities. Some examples of disorders are shown in Table 1.1 and 1.2.

1.3.2 Movement disorders

Different from musculoskeletal injuries, movement disorders are generally indicated by various symptoms and signs resulted from different neurological disorders and conditions. There are two main symptoms associating these disorders. In one symptom, the movements of the patients are much slower and less than that is for healthy people, which are classified as hypokinesias. On the other hand, some patients may experience excessive and abnormal involuntary movements, which belong to hyperkinesias [47]. According to [48], some common examples of hypokinesias include bradykinesia, freezing, rigidity and stiff muscles, while those belong to hyperkinesias
Table 1.1: Examples of musculoskeletal injuries in joints of upper extremities. “Inj” refers to injury and “ST” refers to studies using physical rehabilitation to treat the conditions or gain recovery (the same as that in Tale 1.2).

<table>
<thead>
<tr>
<th></th>
<th>Shoulder</th>
<th>Elbow</th>
<th>Wrist</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movements</td>
<td>flexion, extension, abduction, adduction, internal and external rotation[18]</td>
<td>“flexion and extension at the ulnohumeral and radiocapitellar articulations, while pronation and supination at the proximal radioulnar joint”[19]</td>
<td>flexion, extension, radial deviation and ulnar deviation [20]</td>
</tr>
<tr>
<td>Inj</td>
<td>shoulder impingement</td>
<td>tennis elbow</td>
<td>carpal tunnel syndrome</td>
</tr>
<tr>
<td>Description</td>
<td>It “occurs against the anterior edge and undersurface of the anterior third of the acromion, the coracoacromial ligament, and at times, the acromioclavicular joint” [21] and deemed as one of the factors that lead to shoulder disability [22].</td>
<td>Although it is not perfectly understood, it negatively influences “the attachment of the extensors of the forearm at the lateral side of the elbow”, thereby leading to pain [23].</td>
<td>It usually is caused by the pressure on the median nerve on a wrist and leads to various conditions, such as pain, paraesthesiae, hypoaestesia and so on[24].</td>
</tr>
<tr>
<td>ST</td>
<td>[25]</td>
<td>[26]</td>
<td>[27]</td>
</tr>
<tr>
<td>Description</td>
<td>adhesive capsulitis</td>
<td></td>
<td>scaphoid</td>
</tr>
<tr>
<td>ST</td>
<td>[30]</td>
<td></td>
<td>[31]</td>
</tr>
</tbody>
</table>

are chorea, dyskinesia, myoclonus, tics and tremor.

There are some commonly seen diseases and conditions that are associated with one or multiple movement disorders, examples of which are discussed as follow.
Table 1.2: Examples of musculoskeletal injuries in joints of lower extremities

<table>
<thead>
<tr>
<th>Movements</th>
<th>Hip</th>
<th>Knee</th>
<th>Ankle</th>
</tr>
</thead>
<tbody>
<tr>
<td>flexion, extension, abduction, adduction, internal and external rotation[32]</td>
<td>There are two ways to describe the degree of freedom (DOF) on a knee. One is with two DOFs (flexion-extension and axial rotation) [33] and the other is with six DOFs (flexion-extension, varus-valgus, internal-external rotation and mediolateral, anteroposterior and superoinferior translation around mediolateral, anteroposterior and superoinferior axis) [34].</td>
<td>extension, flexion, valgus and varus[35]</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Inj1</th>
<th>hamstring strain</th>
<th>patellofemoral pain syndrome</th>
<th>achilles tendonitis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>It usually is associated with lower extremity activities, like football, soccer, dancing and so on, while this condition occurs in different phases of motions in various types of activities[32].</td>
<td>It is an anterior knee pain and mainly resulted from “aberrant motion of the patella in the trochlear groove” [36].</td>
<td>The physical findings of this condition include soft tissue swelling, local tenderness and sometimes crepitus[37].</td>
</tr>
</tbody>
</table>

| Description                      | It contributes 2% to 5% of all sport injuries[41]. Vincent et al. [41] also mentioned that the diagnosis of this pain is hard because of its complex anatomy in affected region, as well as the coexistence of multiple injuries. | The causes are in two major categories, including non-contact (usually resulted from the sudden deceleration before changing direction or a landing motion) and contact (valgus collapse) [42]. | It can be deemed as the most common injury on ankles[43], which is usually cause by inversion of the foot[43]. |

<table>
<thead>
<tr>
<th>ST</th>
<th>[38]</th>
<th>[39]</th>
<th>[40]</th>
</tr>
</thead>
<tbody>
<tr>
<td>ST</td>
<td>groin pain</td>
<td>anterior cruciate ligament (ACL) injury</td>
<td>lateral sprain</td>
</tr>
</tbody>
</table>

| ST                               | [44]                                                                | [45]                                                                | [46]                                                                |
First of all, being the most common neurological disorder [49] and adult movement disorder, essential tremor (ET) is about 20 times as much prevalence as Parkinson’s disease. Due to the fact that patients with ET are highly likely to have tremor with 4 to 12 Hz, their ability to perform tasks in both workspace and daily living is adversely affected [50]. Three potential risks that may lead to ET include age, ethnicity and a family history [49]. As for the pathology of ET, although there are some controversial discussions, three hypothesis are tested. According to the conclusions made by [51], although more proof should be made, some evidence can be found for the neurodegeneration hypotheses. In addition, it is confirmed that GABAergic tone is reduced in the same location as the change of neurodegeneration. Lastly, some studies have been done to test the hypothesis that there is a oscillating network, rather than one oscillator leading to essential tremor. Though a number of medical approaches have been proposed to treat essential tremor [52], physical rehabilitation methods are also used [53].

The second condition that leads to a number of different movement disorders is Parkinson’s disease (PD), which is the second most common neurodegenerative disorder [54]. As for its prevalence, 160 out of 100000 people in Western Europe with age over 80[55] suffer from this condition. While in China, 1.7 % people with their age more than 65 and approximately 1.7 million people with age more than 55[56] are diagnosed with PD. The potential causes of PD can be generally divided into two categories, including non-genetic (environmental) and genetic risk factors [54]. The former includes but is not limited to endotoxin (lipopolysaccharide) resulted from Salmonella minnesota [57] and pesticide[58], while the latter involves causative genes and susceptibility genes [54]. According to [59], the movement disorders experienced by a PD patient can be classified into three stages. In the initial stage of PD, the patient may have forward stooped posture, festinating gait, rigidity and so on. Furthermore, during the first ten years of PD, phenomena, such as resting
tremor, hypokinesia and micrographic handwriting, are observed. Moreover, in the latter phase, patients may exhibit dyskinesia, akinesia, postural instability and etc. In terms of treatment, various kinds of medical therapies, such as levodopa, as well as surgical approaches and deep brain stimulation are utilised to control symptoms. Physical therapies, such as [60] are also considered.

Although previous two conditions are very common, they are not fatal conditions. In comparison, stroke is one of the most fatal conditions in the developed countries[61]. However, the majority of the stroke suffers may be alive after the initial injury. Thus, they are highly likely to lose some motor functions in the rest of their lives for a long period[62]. In [63], it is mentioned that, in 2005, there are 5.7 million death in low and middle income countries resulted from stroke, which may increase significantly to 6.5 million and 7.8 million in 2015 and 2030 if there is no proper intervention. As for the risk factors that may lead to stroke, it is concluded that age, gender, race, ethnicity and heredity are important markers of risk [64]. Additionally, hypertension, cardiac disease, diabetes, glucose metabolism, lipids, cigarette smoking, alcohol, illicit drug use, lifestyle may contribute to resulting in a stroke [64]. The main pathophysiology of ischemic stroke is tissue necrosis resulted from excitotoxic, inflammatory and microvascular mechanisms [65]. Similar to PD, a number of involuntary abnormal movements associate stroke. For example, Alarcon et al. [66] observed chorea, dystonia, tremor and parkinsonism in their patients, while more movement disorders are reviewed in [67]. Similar to other disorders, physiotherapies are also widely utilised to assist stroke patients to regain some physical functionality [68].

1.4 Sensors in Tele-rehabilitation

In the past few decades, various types of sensors have been utilised as patient monitoring and data acquisition tools. In this section, the use of three types of
Chapter 1. Introduction

1.4.1 Kinect

Recently, a number of non-invasive, portable and affordable optical 3D motion capture devices have emerged. These products include Leap Motion®, controller, ASUS® Xtion PRO LIVE, Intel® Creative Senz3D, Microsoft Kinect® and so on. Among them, Kinect is the most popular motion capture device for the whole body motion capturing. Its first version was released in 2010 with Xbox 360 for gaming purpose and the second version was with Xbox One in 2014. Since the second version was released for just one year, the majority of the applications in tele-rehabilitation field was with the first version.

The first version of Kinect utilised a depth sensor provided by a company named “PrimeSense” [69]. The appearance and components of this version of Kinect is shown in Fig. 1.3. The infrared projector and the corresponding camera can be modelled as indicated in Fig. 1.4. These sensors measure the depth information via structured light principle, which analyses a pattern (such as that in Fig. 1.5) of bright spots projected to the surface of an object [72]. In the case of Kinect, these “bright spots” are infrared light and unobservable by human eyes directly. Furthermore, Kinect utilised the other two techniques to further process the in-

Figure 1.3: Appearance and components of Kinect version 1 [70]
Chapter 1. Introduction

Figure 1.4: The pinhole camera model of Kinect version 1[71].

Figure 1.5: An example of projected pattern of bright spots on an object [72]

formation to generate depth maps. These two tools include depth from focus and depth from stereo [73]. The principle of the former is that the further the object is, more blurred it will be [74], while the latter utilised parallax to estimate the depth information.

Different from the first version of Kinect, the second version (refer to Fig. 1.6) measures the depth information with time-of-flight (ToF) technique[75], which is stated that the distance can be measured by knowing the speed of light and the duration the light used to travel from active emitter to the target. Guessed in [75], this version of Kinect utilised indirect time-of-flight, which measures the “phase shift between emitted and received signal”. The depth is computed as

\[ d = c \frac{\Delta \phi}{4\pi f}, \]

(1.4.1)

where \( f \) is the modulation frequency, \( c \) is the light speed and \( \Delta \phi \) is determined phase shift.

Studies have also been conducted to investigate the accuracy of joints positions
tracked by the first version of Kinect. Different studies have different conclusions on the accuracy of skeleton joint tracking. For instance, Webster et al. [77] reported that the accuracy was around 0.0275m after removing offset by resetting the alignment of the average points for each record set. Therefore, they concluded that Kinect version 1 was sufficient for clinical and in-home use. Obdrzalek et al. [78] evaluated the accuracy of the first version of Kinect, PhaseSpace Recap and Autodesk MotionBuilder in the environment of coaching elderly people. The result reported in their paper is that the error of the skeleton built by both Kinect and MotionBuilder is around 5 cm. However, in general, the accuracy is about 10 cm due to unavoidable factors, such as occlusions. Therefore, they suggested that current skeletonisation approach enabled Kinect to measure general trends of movements, while improved skeletonisation algorithms should be investigated if Kinect was used for quantitative estimation. Furthermore, Xu et al. [79] evaluated the accuracy of both the first and the second version of Kinect for static postures. From their experiment, they concluded that the accuracy of the first version Kinect varies depending on the posture. For instance, the error was only 26mm for shoulder centre in upright standing posture, while 452 mm for the right foot joint in sitting posture of the right leg on top of the left one. By comparison, for the second version of Kinect, when the right foot was raised, the error of left elbow was only 26mm, while 418 mm when the right leg was on the left one. As a result, a conclusion was made that though the resolution of the second version of Kinect was improved
significantly, its tracking accuracy of joint centre was not improved.

The comparison of the specifications between two versions of Kinects is shown in Table 1.3. From the comparison, it is obvious that the second version of Kinect provides a larger viewing angle, higher resolution in both depth images and color images, and more number of tracking joints.

Table 1.3: Comparison of basic technical specifications between two versions of Kinects

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewing Angle (vertical)</td>
<td>43°</td>
<td>70°</td>
</tr>
<tr>
<td>Viewing Angle (horizontal)</td>
<td>57°</td>
<td>60°</td>
</tr>
<tr>
<td>Vertical tilt range</td>
<td>±28°</td>
<td>no</td>
</tr>
<tr>
<td>Frame rate (frame per second)</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td>Depth Resolution (pixels)</td>
<td>320 × 240</td>
<td>512 × 424</td>
</tr>
<tr>
<td>Color Stream Resolution (pixels)</td>
<td>640 × 480</td>
<td>1920 × 1080</td>
</tr>
<tr>
<td>Skeleton</td>
<td><img src="image" alt="Skeleton" /></td>
<td><img src="image" alt="Skeleton" /></td>
</tr>
<tr>
<td>Number of Tracking Subjects</td>
<td>6 (can only display 2)</td>
<td>6 (can display 6)</td>
</tr>
<tr>
<td>Sensing Principle</td>
<td>[82]</td>
<td>[83]</td>
</tr>
</tbody>
</table>

Though Kinect is initially developed for gaming, it is considered in tele-rehabilitation as a non-invasive and affordable motion capture device. Antón [84] proposed a tele-rehabilitation system (KiReS) using Kinect as the motion capture device. On the patient side, two avatars were displayed to represent the motion recorded by the therapist (reference motion) and that performed by the patient. Therefore, the patient was able to see the differences between his/her motion and the reference. Eventually, the incorrect movements could be corrected in time. On the therapist side, new motions could be created to suit the patient’s conditions by composing various existing movements or record completely new ones. Luna-Oliva et al. [85]
utilised Kinect Sports™, Joy Ride™ and Disneyland Adventures™ to provide tele-rehabilitation services to children with cerebral palsy in their school. Their experiment results shown that it is feasible to use Kinect as a therapeutic tool in children with cerebral palsy and the improvements in global motor function could be the result of using this tool. Ortiz-Gutierrez et al. [86] applied Kinect in providing tele-rehabilitation services to patients with postural control disorders. The experiment results demonstrated an improvement over general balance in both groups. In experimental group, the significant differences resulted from visual preference and the contribution of vestibular information.

1.4.2 RGB camera and microphone

Apart from Kinect, conventional RGB cameras and microphone are also pervasively used, especially in the early stage of the history of tele-rehabilitation when virtual reality devices have not been well developed and pervasively utilised. One of the potential reasons is that they are easy to install, cost-effective and well-developed.

In the early stage of the development of tele-rehabilitation, plain old telephone system (POTS) was widely used as the infrastructure of videoconference, which was sufficient to provide tele-consultation. Fig. 1.7 shows an example of using POTS for tele-consultation in the patient’s home, which is given in [87]. Delaplain et al. [88] conducted a pioneering trial between two islands to conduct 59 medical tele-consultations in the form of video-conference. In this trial, diagnostic and therapeutic decisions were made in a number of specialities including physical therapy. This is deemed as one of the first examples of applying video-conference in tele-rehabilitation (although at that time, the word “tele-rehabilitation” had not been raised) with cameras and microphones. Success of these illustrated the feasibility of using video-conferencing in tele-rehabilitation. Later, in 2002, Clark et al. [87] successfully managed a teletherapy case for 17 months. In this case, a POTS was set up between the therapist’ site with a desktop videophone and patient’s home with
a traditional telephone and a television to provide post-stroke tele-rehabilitation services in the form of two-way interactive video-conference. However, the lesson learnt from the case is that the use of this novel approach has a number of requirements to the patients, as well as the caregiver, which may be a potential issue in developing tele-rehabilitation system in the future. As the conclusion, they mentioned that although tele-rehabilitation cannot totally replace the conventional way to deliver rehabilitation services, it indeed contributes to traditional therapy. Furthermore, Savard et al. [89] reported two cases of using video-conferencing to provide tele-consultation service for neurologic diagnosis. Different from the previous two studies where the videoconference systems were utilised by patients at home, the systems in [89] were installed in clinics. Therefore, patients had to visit the clinics for using these facilities. Although it was found that the time consumption of therapists in tele-consultation was similar to in-person consultation, the time was more efficient in the former since multiple parties were able to participate the consultation simultaneously. However, completing remote tests for clinicians would be an important factor that ensures true impact of tele-rehabilitation.

![Figure 1.7: The patient side of the POTS-based tele-rehabilitation system. [87]](image)

With the advancement of the Internet and computers, POTS is gradually replaced by them to deliver tele-rehabilitation services. Russell et al. set-up an Internet-based computer system in two separate rooms in a clinic to evaluate the feasibility of using video-conferencing to assess the kinematic gait [90] by evaluating
the performance of measuring knee angles via the Internet, named Internet-based goniometer (IBG), against the traditional face-to-face approach. The interface of the software is shown in Fig.1.8(a). As a conclusion of the experiment, IBG was comparable to universal goniometer (UG) in terms of intra- and inter-rater reliability. After that, the same set-up was used in the next year to provide tele-medicine to patients after total knee replacements[91]. This modern way of delivering rehabilitation services was welcome by both therapists and patients since it was safe, easy to use and integrate in daily clinic practice. More importantly, the outcome of using this approach was similar to traditional rehabilitation method. Another example is that video cameras were used by Lemaire [92] in a telehealth program to provide tele-consultation and educational services for various disorders. The majority of the patients accessing these services were satisfied with tele-rehabilitation. Though the study also found that the time consumption in tele-consultation was similar to traditional approach, the majority of the therapists agreed that the tele-health was easy to use and they were confident to the assessment results done with telehealth. However, since it was in 2000, not all hospitals could afford brand new computers. Therefore, it is critical to develop low cost tele-rehabilitation system so that the majority can afford it.

1.4.3 Inertial measurement unit (IMU)

IMU is a device that measures angular velocity, orientation, gravitational force and magnetic direction. In the early stage of the development of IMU, gyroscope and accelerometer were usually utilised to provide angular velocity and inertial acceleration. Later on, the integration of magnetometer enables an IMU to measure magnetic direction. As a result, the measurement from an IMU can be more accurate [93]. Since all these sensors are able to provide three dimensional measurements, IMU is widely utilised in applications, such as aircraft navigation [94].

Recently, thanks to the advancement in micro-electromechanical system (MEMS),
(a) Software interface developed and used in [90] to measure the angle of knee via the Internet

(b) A therapist monitoring the knee angle measurement remotely through the Internet.

Figure 1.8: The physiotherapist monitoring the exercise performance of his patient remotely through the Internet. [90]

IMUs can be produced with the size small enough to be worn by human beings. As a result, in recent years, more and more applications of IMUs have been seen in rehabilitation and tele-rehabilitation fields as human motion capture devices [93]. Currently, there have been a number of companies producing and selling IMU sensors, for example Xsens®, YEI Technology®, MotionNode® and so on. Products from them are considered as wearable devices for motion tracking.

(a) Xsens[95]  (b) YEI Technology[96]  (c) MotionNode[97]

Figure 1.9: Pictures of animals

In a small number of applications, single IMU is used to monitor specific conditions (usually relating the movement of one joint) in tele-rehabilitation. For instance, Giansanti [98] utilised an IMU with one 3 axes accelerometer and gyroscope
to detect the risk of falling in tele-rehabilitation. The other example is Han et al. [99] integrated an 6 degree of freedom (6DOF, including 3 axes accelerometer and gyroscope) IMU with a customised ankle foot orthosis (AFO) to provide a tele-rehabilitation diagnostic service for patients with two types of conditions including muscle weakness due to brain injuries and knee replacement surgery most likely due to conditions such as osteoarthritis of the knee joints. Experiments in the study confirmed the high sensitivity and specificity of AFO-IMU module in measuring the flexion and extension motions of the knee joint.

However, it is obvious that single IMU is insufficient to monitor the movement involving multiple joints, such as upper extremity movements and whole body movements. Therefore, body area network (BAN) or body sensor network (BSN) was developed to fill this gap. For instance, Nerino et al. [100] proposed a BSN to provide knee tele-rehabilitation services to patients with anterior cruciate ligament (ACL). In the propose system, multiple IMUs with 9 DOF (including 3 axes accelerometer, gyroscope and magnetometer) were attached to thigh, calf and foot. Therefore, the angle of knee and ankle could be measured. After comparing to the VICON system, the average angular errors measured by the BSN on knee and ankle were 2.4° and 3.1° with standard deviation of 1.8° and 2.4° respectively. Horak et al. [101] summarised the role played by body-worn movement monitor devices in rehabilitation services for balance and gait. Cancela et al. [102] evaluated the wearability of a BAN-based system, named PERFORM, to monitor the symptoms of patients with Parkinson’s disease (PD) remotely. This system is composed of four tri-axial accelerometer locating on two legs and two arms respectively and a central sensor with one tri-axial accelerometer and gyroscope positioned on the waist (refer to Fig. 1.10). Analysis was looking into the comfortableness, biomechanics and physiology of the system. According to the experiment, it was found that patients were generally satisfied to wear such systems. However, some patients concerned
their privacy, as well as how others think about them in public area, thereby showing a bit of anxious and unwillingness to use this system. Furthermore, the strap made patients uncomfortable and difficult to wear the system by themselves. Last but not the least, feedback is necessary during monitoring so that patients know that the system is working properly.

![Figure 1.10: Locations of five sensor worn by a subject [102]](image)

1.4.4 Summary and challenges

From the literature, it is clear that a number of devices have been considered in tele-rehabilitation. For the purpose of evaluation and demonstrating the underlying ideas, Kinects were utilised as an example of affordable OBMCDs to collect data in this thesis for the following two major reasons:

First of all, Kinect is a non-contact motion capture device. Patients do not need to wear any additional item on their body. As a result, they may be able to perform rehabilitation exercises or ADLs with more natural poses.

Secondly, compared to conventional RGB cameras, Kinect is able to tracking 3D positions of 20 to 25 joints throughout a human body. Therefore, without making much effort in processing images, data collected from a Kinect is much easier for further analysis. For assessing kinematic performance in different tele-rehabilitation exercises, information from various numbers of joints are required. For example, in the assessment of the elbow, only three joints, including shoulder, elbow and wrist, are required. However, for assessing the balance ability, information from
joints on two legs, as well as some joints on the trunk and even arms may be needed. Therefore, the optimum or minimum number of joints in tele-rehabilitation applications is determined by the type of the movement and the required features. This is critical for automatically assessing the kinematic performance of patients in a tele-rehabilitation system without the presence of a professional therapist.

However, the majority of affordable OBMCDs still suffers from a number of drawbacks. One is that they may not be accurate enough in tracking small movements, as well as the positions of joins occluded by other body parts. Secondly, their small range of view limits their applications in lower extremity rehabilitation, which in many cases involves a large moving area. A solution for these limitations is discussed in Chapter II.

1.5 Human Motion Encoding in Tele-rehabilitation

After capturing patients’ movements, it is vital to reduce information so that the key features representing the characteristics of the movements can be selected. Therefore, before conducting automated performance assessment, it is important to extract these features by encoding human motions.

1.5.1 Human motion encoders in action recognition

The underlying problem has been extensively studied in the field of human action recognition. For instance, Ren et al. [103] employed the silhouette of a dancer to represent his/her performance by extracting local features to control animated human characters. Wang et al. [104] obtained the contour of a walker from his/her silhouette to represent the walking motion. A spatio-temporal silhouette representation, the silhouette energy image (SEI), and a variability action models were used by Ahmad et al. [105] to represent and classify human actions. In both visual based and non-visual based human action recognition differential features such as velocity
and acceleration, where motion statistics, their spectra and a variety of clustering
and smoothing methods have been used to identify motion types. A two-stage dy-
namic model was established by Kristan et al. [106] to track the centre of gravity
of subjects in images. Velocity was employed as one of the features by Yoon et al.
[107] to represent the hand movement for the purpose of classification. Further,
Panahandeh et al. [108] collected acceleration and rotation data from an inertial
measurement unit (IMU) mounted on a pedestrian’s chest to classify the activities
with continuous hidden Markov model. Ito [109] estimated human walking motion
by monitoring the acceleration of the subject with 3D acceleration sensors. More-
over, angular features, especially joint angle and angular velocity, have been used
to monitor and reconstruct articulated rigid body models corresponding to action
states and types. Zhang et al. [110] fused various raw data into angular velocity
and orientation of upper arm to estimate its motion. Donno et al. [111] collected
angle and angular velocity data from a goniometer to monitor the motions of hu-
man joints. Angle was also utilised by Gu et al. [112] to recognise human motions
to instruct a robot. Amft et al. [113] detected the feeding phases by constructing
hidden Markov model with the angle feature from the lower arm rotation. Apart
from the above, only a few have considered a similar approach of trajectory shape
features such as curvature and torsion. For example, Zhou et al. [114] extracted
the trajectories of upper limb and classified its motion by computing the similarity
of these trajectories.

1.5.2 Human motion encoders in physical tele-rehabilitation

Although a number of encoding approaches have been investigated, not all of these
approaches are adopted in physical tele-rehabilitation field, where the angles and
trajectories of joints are usually considered.

In these two representations, angle-based approach is more widely used. This
is mainly because human limbs are normally modelled as articulated rigid bodies.
Additionally, some measurement devices, such as IMU, are able to measure the orientation of limbs easily. Therefore, angles of joints and orientation of limbs can be acquired without much difficulties. Limb segments are hinged together with various degrees of freedom (DOF), which can be seen in Table 1.1 and 1.2.

A number of examples utilising angle and its derivatives as encoders can be found. For instance, Tseng et al. [115] evaluated two platforms (Octopus II and ECO) to capture human motions for home rehabilitation. In two platforms, angles of joints were measured to represent the movement of limbs by the same type of compass (TDCM3 electrical compass) and different accelerometers, including the FreeScale MMA7260QT accelerometer and the Hitachi-Metal H34C accelerometer respectively. Two types of angles were taken into consideration. One was the joint angle between limbs, such as the elbow and knee angle. The second type was the angle between the orientation of a sensor and the gravity. Moreover, in the telerehabilitation system developed by Luo et al. [116], angles of joints were utilised to encode the movements of the upper extremity. In this system, the shoulder and wrist angles were measured by two IMUs since these two joints were modelled with three degree of freedom on each joint while that of elbow and joints fingers were measured by a optical linear encoder (OLE) and a glove made by multiple OLEs because these joints could be modelled with one degree of freedom. Additionally, Durfee et al. [117] introduced two bilateral electrogoniometers in a home telerehabilitation system for post stroke patients. These bilateral electrogoniometers were attached to the wrist and hand of the subject respectively. The angle of flexion and extension movements in the wrist and the first metacarpophalangeal (MCP) joints were measured to represent the movement of the wrist and hand. Two potentiometers (refer to Fig. 1.11(a)) were utilised to calculate the angles of the joints ($\theta$) as

$$
\theta = 180 - \cos^{-1} \left( \frac{l_3^2 + L^2 - l_1^2}{2l_3 L} \right) - \cos^{-1} \left( \frac{l_4^2 + L^2 - l_2^2}{2l_4 L} \right),
$$

(1.5.1)
where the distance from the anatomic joint to the linkage joint is

\[ L^2 = l_1^2 + l_3^2 - 2l_1 l_3 \cos \alpha. \]  

(1.5.2)

Apart from the above three examples, in some studies that motion trajectories of joints are captured, angle information is still derived for encoding human movements. For example, Adams et al. [118] developed a virtual reality system to assess the motor function of upper extremities in daily living. To encode the movement, they used swing angle of shoulder joint along Y and Z axis, twist angle of shoulder, angle of elbow, their first and second derivatives, bone length of collarbone, upper arm and forearm, as well as the pose of the vector along the collarbone to describe the movement of upper body. Here collarbone is a virtual bone connecting two shoulders. These parameters were utilised in an unscented Kalman filter as state, while the positions of shoulders, elbows and wrists reading from a Kinect formed the observation. Another example is that Wenbing et al. [119] evaluated the feasibility of using a single Kinect with a series of rules to assess the quality of movements in rehabilitation. Five movements, including hip abduction, bowling, sit to stand, turning and toe touching were studied in this paper. For the first four movements, angles were used as encoders. For instance, the change of angle between left and
right thighs (the vector from hip centre to left and right knee) was used to represent the angle of hip abduction, while the dot product of two vectors (from the hip centre to left and right shoulder) was utilised to compute the angle encoding the movement of bowling. Additionally, Olesh et al. [120] proposed an automated approach to assess the impairment of upper limb movements caused by stroke. To encode the movement of upper extremities, the angle of four joints, including shoulder flexion-extension, shoulder abduction-adduction, elbow flexion-extension and wrist flexion-extension, were calculated with the 3D positions of joints measured with Kinect.

Though angles of joints, as well as their derivatives are utilised widely in encoding human motions, trajectories of joints and their derivatives can also be observed in some rehabilitation and tele-rehabilitation applications.

First example is that Chang et al. [121] developed a program to use Kinect as the motion capture device for spinal cord injuries (SCI) rehabilitation. In this program, the trajectories of hand, elbow and shoulder were recorded to represent the external rotation of upper extremities. Similarly, Su [122] also developed a rehabilitation system, named KHRD, to provide home-based rehabilitation services. To represent human motions, two key features were used, including trajectories of joints, as well as their speed. Additionally, Cordella et al. [123] modified the Kinect into a marker-based device to measure the positions of joints on a hand (refer to Fig. 1.12). Markers with dimension of 1.2 cm were attached to the joints of fingers, as well as the wrist. After detecting the center of these markers, a robust tracking scheme was developed to track the position of each markers. Thus the movement of a hand was encoded by the trajectories of each joint on the hand, as well as the trajectory of the wrist.
1.5.3 Summary and challenges

From the literature, it is found that encoders used in human motion recognition are similar to those in physical tele-rehabilitation. For instance, features like trajectories, velocity, acceleration, angle, angular velocity and angular acceleration are most commonly used in both fields. Though patients with movement disorders usually have limited range of motions, they may be required to do certain tasks so as to evaluate their ability to performance ADLs, which usually are composed of a series of simple movements.

As a result, there remain challenges in developing formal descriptions and robust computational procedures for the automatic interpretation and representation of motions of patients. The majority of studies [124, 125] employed a variety of human motion encoders to recognise or decompose general movement, such as reaching, waving hands, jumping, walking and so on. Few of them investigate details in each general movement, for example, the even smaller atomic components included in these general movements which are of importance for syntactic and structural descriptions of human movements in detail, especially in clinical and rehabilitation environment, where the details of movements of body parts require a form of motion language or, at least, syntax. A novel approach to encode human motion trajectories will be discussed in Chapter III.
1.6 Patients’ Performance Evaluation

In recent decades, with the advancements in tele-rehabilitation and associated motion capture technologies, an increasing number of research and development activities are focusing on the development of automated quantitative measures of patient performance in ADLs [126, 127]. Due to the important role played by the upper extremity in ADLs [128], an automated approach for measuring and assessing the ability of upper extremities to perform certain tasks is vital for tele-rehabilitation systems to achieve their full potential.

1.6.1 Questionnaire based assessment scales

In the past few decades, a number of approaches have been proposed for assessing upper extremities, the majority of which are questionnaire-based. For musculoskeletal movement disorders of the extremities, most scales are generic. For instance, the self-reported Musculoskeletal Function Assessment (MFA) instrument [129], Short Musculoskeletal Function Assessment (SMFA) questionnaire [130] and self-administered measure of disabilities of the arm, shoulder and hand [131] were developed but few of them are condition specific. More examples can be found in [128]. However for neurological movement disorders, assessments are more disease-specific and rarely focus on upper extremities. For instance, Chedokee-McMaster (CM) assessment, the Fugl-Meyer (FM) assessment, and the Wolf Motor Function Test (WMFT) assessment are for stroke, The Fahn-Marsden rating scale (F-M) [132], Global Dystonia Rating Scale (GDRS) [133], Unified Dystonia Rating Scale (UDRS) [134] and so on [135] were developed for dystonia. The Parkinson’s Disease Questionnaire (PDQ-39) [136] and its shorter version (PDQ-8) [137], as well as Parkinson’s Disease Quality of Life questionnaire (PDQL) [138], Webster [139] and Unified Parkinsons Disease Rating Scale (UPDRS) [140] were developed for
Parkinson’s disease. More relevant to our work, Lane et al. [141] developed Abnormal Involuntary Movement Scale (AIMS) to assess patients with dyskinesia. In this scale, the amplitude of involuntary movements was taken into consideration. In addition, Goetz et al. [142] proposed to use the Objective Dyskinesia Rating Scale (also known as the Rush Dyskinesia scale) to assess the severity of dyskinesia.

1.6.2 Automated kinematic performance assessment

Recently, with the development of sensing technologies, a number of approaches have been proposed to either automate conventional testing scales or develop new methods. Some examples are shown as follow.

Knorr et al. [143] automated two tasks, including reaching to the front and to the side, in WMFT for people after stroke. Three accelerometers were attached to the hand, the corresponding forearm and upper arm respectively to capture the characteristics of motion patterns. Eventually, two linear features (root mean square errors of acceleration and jerk) and a non-linear feature (approximate entropy of acceleration) were evaluated as the measurements of functional limitation and motor impairment. Similarly, Wade et al. [144] tried to automate widely used WMFT for post-stroke patients. In their proposal, one IMU was attached to the wrist of the subject to measure the time used by the subject to finish each task in different WMFT tasks. Hester et al. [145] utilised 14 features to predict the clinical measurement scores of patients with stroke. These features were extracted from data collected by four accelerometers, three of which were attached on the affected hand, forearm and upper arm, while the fourth sensor was on the trunk. As for the tasks, different from previous two literatures, some tasks were not included in WMFT. The process of predicting clinical scores from these 14 features is shown in Fig. 1.13. Furthermore, Leibowitz et al. [146] introduced a protocol to quantitatively measure the proprioception deficit. In the protocol, the affected hand was located under a square board so that it could not be seen by the subject.
Figure 1.13: The process of predicting clinical scores from 14 features.

After passively moving the impaired hand to one of four locations, the health hand was required to be moved to the same location actively. Eventually, the positions of both hands were measured by a MiniBIRD500 magnetic tracking system and the positional differences between the impaired and non-impaired hands were computed as the indicator.

1.6.3 Summary and challenges

In rehabilitation field, assessing the performance of limb functions is of importance in evaluating other treatment procedures[147]. This task can be relatively easy achieved in clinical environments. However, when it comes to tele-rehabilitation, it becomes difficult due to the absence of well-trained clinicians in the majority of cases. From the literature, it is found that a number of features have been used to perform automated assessments. However, the majority of them are calculated from dynamic aspect of movements, such as velocity, acceleration and jerk. Therefore, a challenge here is to derive features for kinematic movement assessment based on shape information from motion trajectories. A novel approach to evaluate patients’ kinematic performance by investigating both shape and dynamics of the motion trajectories will be discussed in Chapter IV.
1.7 Contributions

To tackle the challenges in developing an automated human kinematic performance assessment tool for tele-rehabilitation listed in Sec. 1.2, three main contributions are made in this thesis as follows.

First of all, robust linear filtering is proposed to fuse 3D skeletons collected from a number of Microsoft Kinects®. By doing so, tracking accuracy of human motion can be increased, the range of view can be extended and the problems caused by occlusion can also be reduced to a certain degree. In paper [148], the robust linear filtering is introduced to fuse data from multiple Kinects.

Secondly, a novel two-component encoding model is proposed to represent human motion trajectories in sensor (joint) level. This model encodes a human motion trajectory into two types of independent sub-models, namely a shape model and a dynamic model. The former is further used to decompose complex human motion trajectories into pre-defined atomic motions for further analysis. This encoding model is introduced in [149, 150, 151].

Thirdly, after finishing a tele-rehabilitation session, one or multiple motion trajectories from various joints of a patient with dyskinesia is encoded with shape models and dynamic models. Then the smoothness of these models are analysed to assess the performance of the movement during the session. Eventually, the ability of the patient in performing certain upper extremity tasks can be evaluated from a kinematic point of view. The detail of this contribution is introduced in [152].

Lastly, some contributions are made to develop a mobile cloud based tele-rehabilitation system [153]. The characteristics of motion trajectories is taken into consideration to build a multi-level motion encoding scheme. Thus motion trajectories of patients can be encoded with various methods depending on the mobile devices they use, as well as the speed of Internet connection. Eventually, the duration for a user to use mobile devices to access tele-rehabilitation services can be
extended as much as possible.

The process of conducting automated assessment of human kinematic performance is summarised in algorithm 1.

\textbf{Algorithm 1} Automated assessment of human kinematic performance

1: Define the motion protocol to be tested;
2: Define the optimal number of Kinects $M$ and empirically put Kinects to the suitably places;
3: Calibrate the Kinects (refer to Chapter 2) and estimate $R_m$ and $t_m$ for each Kinect;
4: Fuse the joint positions from the build-in skeletons (or reflective markers) in these Kinects (refer to Chapter 2) to estimate $\hat{s}$
5: Extract two-component encoding model $\kappa$, $\tau$ and $v$ (refer to Chapter 3)
6: Calculate the approximate entropy of the shape model $H_{\kappa,\tau}$ and instantaneous acceleration $H_{\bar{a}}$ to represent the smoothness (refer to Chapter 4)
7: Classify the exercise into one of pre-defined ability levels (refer to Chapter 4)

\section{1.8 Thesis Outline}

The thesis is organised as follow.

Chapter 2 outlines the theoretical background of filtering techniques for information fusion, including Kalman filter, particle filter and robust linear filtering. In the experiments, fusion results from the Kalman filter and the particle filter were compared with that from the robust linear filter. The results illustrate the superiority of the proposed approach for cases with and without occlusion.

Chapter 3 presents the two-component encoding model, the spatial indexing scheme, as well as results of applying them on human motion trajectory representation and decomposition. Additionally, various commonly used filters are compared to reduce the noise in captured data. Simulation experiments show the characteristics of this encoder and the feasibility of using this encoder to decompose complex motion trajectories. A real-data experiment follows to further illustrate the performance of the proposed method in real situations with noisy data collected from a Kinect. Eventually, the most suitable approach to eliminate noise in raw data is
selected from three candidates.

Chapter 4 is written for presenting a novel method for estimating the smoothness of motion trajectories using the approximate entropy of shape and dynamic models. This approach is utilised to assess the kinematic performance of patients with dyskinesia. The results show that the smoothness of motion trajectories can potentially be used as a criterion to evaluate the ability of patients with neurological movement disorders, such as dyskinesia, to perform functional upper extremity tasks in their daily life. The automated approach is able to reach a high agreement with human observers, which illustrates the feasibility of applying the proposed approach.

Chapter 5 introduces a framework to build a mobile cloud-based tele-rehabilitation system. This system take the characteristics of motion trajectories into consideration so as to provide various encoding approaches. These methods are used as a scheme to alleviate one of the fundamental issues, namely the limited battery resources in mobile devices. Furthermore, the concept of analysis oriented decision support system (AODSS) and security service layer (SSL) are also introduced into the system to protect data in the tele-rehabilitation system and enable it to provide supportive suggestions for therapists to make better treatment decisions.

Chapter 6 concludes the thesis by reviewing the main contributions raising possible future work in tele-rehabilitation area from an engineering perspective.
Chapter 2
Multi-Kinect Skeleton Fusion

2.1 Introduction

Human motion tracking and recognition has received renewed interest particularly
due to the progress made in sensing and data integration[154, 155, 156]. Numerous
approaches have been pursued, employing readily available commercial products
that impact many areas ranging from computer gaming to remote rehabilitation.
Inertial Measurement Units (IMUs) are used[157] in wearable[158] wireless systems
to capture the movements in real time (MT9 in Xsens Motion Tech, G-Link). In
contrast, systems with integrated vision normally operate with markers for the real
time tracking of body parts. Robot assisted bio-kinematic motion detector systems
are typically used to extract particular actions via pattern recognition or data min-
ing means[159]. Although a number of approaches have been discussed to address
various shortcomings in these systems designed for human movement detection
applications, limited attempts[111] have been made to enhance the measurement
accuracy for these commercial devices that are primarily designed for the com-
puter gaming industry. For more effective use of these systems in clinical settings,
reliability and robustness are two key aspects that need closer attention. Multi-
ple cameras[160] are commonly used for improved accuracy with marker (active
or passive) based tracking[155] with distinct advantages despite the requirement of
perspective imagery, occlusions and the need for relatively significant infrastructure
costs. Furthermore, the need for designated clothing, multiple cameras and addressing the data association[161] problem limit the scope of the application domain.

Recently, the Microsoft Kinect© that provides 3D locations of up to twenty joints [162] (blue dots on joints in Fig. 2.1) at an affordable cost has drawn the attention of rehabilitation researchers as a potential device for exercise monitoring[163]. Compared to the commercially available and lab-based VICON system (costs hundreds of thousands of dollars), a Kinect system is more suitable for home-based tele-rehabilitation as people with disabilities are more likely to possess financial constraints due to generally lower socioeconomic situations[164]. Therefore, affordable and user-friendly sensors that can be easily setup is a core requirement; particularly in home environments or confined clinical assessment areas. In terms of joint position measurement, the human body is segmented with randomised decision forests in a point cloud environment produced by projecting an infrared mesh on to the human body from the infrared camera on the Kinect©. The positions of the joints are subsequently deduced with a local model-finding approach[165]. Primarily, considered as a motion and voice sensing input device for the Xbox video game console, Kinect© is currently considered as the fastest selling consumer electronics device. The introduction of this device has somewhat transformed academic research in this arena markedly as it provides the developers with data that enable them to concentrate on the higher level architectures in integrated systems design. For an example, with this device, the multi-camera vision problem traditionally considered as an inherently a non-linear estimation problem[166] involving perspective projection can now be considered as a linear problem with readily available position data.

However, there remain some open issues which exist pertaining to the robustness, accuracy and reliability in using these devices in health care applications despite some attempts being made [154] to improve their localization accuracy. While
a majority of past work[167] has primarily focused on the reconstruction of 3D object surfaces with the cloud points generated by Kinects®, some work has been reported on combining the Kinects® with other devices, such as cameras [168, 154] and inertial sensors [169, 170] to improve measurement and tracking accuracy. In this work, human movements are precisely tracked with multiple Kinects® using generic 3D point position information.

As optical tracking systems typically form a non-linear estimation problem, employing model based filtering in such settings can cause the estimator to diverge. It is well known that linearisation of such models when implemented with Extended Kalman Filtering may potentially be unstable without appropriate initialisations in addition to the computational cost compared to their linear counterparts[161]. In contrast, the entire formulation was developed in a linear framework therefore removing the need to use non-linear estimation techniques. A global coordinate system was established to map the positions of human body joints by these sensor arrays. A setup phase employing maximum likelihood estimation was executed initially to ascertain the position and the orientation of Kinect® receivers that were
positioned utilizing the empirical knowledge of the exercise routines as this information is required for the subsequent filter implementations. Using both simulated data and real data, a Kalman filter, a particle filter and a robust linear filter were used to fuse the sets of skeleton data in the local coordinate frame of each Kinect into a common set of skeletal data in the global coordinate frame thereby improving the accuracy of the joint positions resulting in real time iso-skeletal positioning. Assuming a more generic uncertainty description, a robust linear filter [171, 161] was proven to perform better than the Kalman filter as well as the particle filter in a number of experimental scenarios.

This chapter essentially looks into the problem of increasing the accuracy and reliability of 3D vision sensor based human motion capture aimed at tele-rehabilitation. In this context, real time application and occlusions are intrinsic features that require closer investigation while affordability and ease of use are vital for the uptake by the health care providers and patients undergoing therapy [164].

The main contributions of this study are highlighted as follows:

- Linear formulations of human body kinematic tracking;
- Robust version of an extended Kalman filter improving the accuracy and the robustness in comparison to existing approaches. Positioning and occlusion effects vital in a practical context as a natural extension to the filter implementation;
- Numerical justification of the largest gain when increasing the number of receivers;

The remainder of the chapter is structured as follows. In Sec. 2.2, the dynamic and measurement models that postulate our approach is described. The maximum likelihood based approach is also introduced for empirically placed receiver position estimation. The relevant theory of Kalman filter, particle filter and robust linear
filtering are all described in Sec. 2.3 leading to our proposed robust version of the Kalman filter. Sec. 2.4 further discusses the improvements due to fusion of information and provides an insight into the advantages of increasing the number of sensors. In Sec. 2.6 simulations and experimental validations of our proposed approach are presented followed by concluding remarks in Sec. 2.7.

2.2 Linear Model of Human Motion Multi-Kinect System

In this study, model-based filtering is employed for real-time motion tracking where \( N \in \mathbb{N}^+ \) number of points are tracked by \( M \in \mathbb{N}^+ \) number of receivers. The position of \( m^{th} \) Kinect© with respect to the global coordinate system is denoted as

\[
t^m = [t^m_1, t^m_2, t^m_3]^\top \in \mathbb{R}^3
\]

with \( \top \) denoting transposition. Here \( m = 1, 2, \ldots, M \) and \( n = 1, 2, \ldots, N \).

Assume that the \( n^{th} \) object trajectory with respect to the global co-ordinate system is denoted by

\[
x^n = [x^n_1, x^n_2, x^n_3, x^n_4, x^n_5, x^n_6, x^n_7, x^n_8, x^n_9]^\top \in \mathbb{R}^9.
\]

Here, the three tuples correspond to position, velocity and acceleration along the cartesian \( X, Y \) and \( Z \) directions respectively.

Moreover, the observation data of the \( n^{th} \) joint from the \( m^{th} \) Kinect is denoted as

\[
y^n_m = [y^n_{m,1}, y^n_{m,2}, y^n_{m,3}]^\top \in \mathbb{R}^3
\]

with reference to the built-in coordinate system of the \( m^{th} \) Kinect© of the multi-Kinect© system. Since it can capture more than one object at the same time, the observation data structure of the \( m^{th} \) Kinect© is

\[
y_m = [y^1_m \ldots y^n_m \ldots y^N_m]^\top \in \mathbb{R}^{3N}.
\]

The state transaction matrix \( A \), the input matrix \( B \) can be defined as:

\[
A = \begin{bmatrix}
A & 0_9 & \cdots & 0_9 \\
0_9 & A & \cdots & 0_9 \\
\vdots & \vdots & \ddots & \vdots \\
0_9 & \cdots & 0_9 & A
\end{bmatrix} \in \mathbb{R}^{9N \times 9N}, \quad (2.2.1)
\]
\[ B = \begin{bmatrix} B & 0_{9 \times 3} & \cdots & 0_{9 \times 3} \\ \vdots & \vdots & & \vdots \\ 0_{9 \times 3} & \cdots & B \end{bmatrix} \in \mathbb{R}^{9N \times 3N}, \quad (2.2.2) \]

where

\[ A = \begin{bmatrix} I_3 & I_3\Delta t & I_3\Delta t^2/2 \\ 0_3 & I_3 & I_3\Delta t \\ 0_3 & 0_3 & I_3 \end{bmatrix} \in \mathbb{R}^{9 \times 9}. \quad (2.2.3) \]

and

\[ B = \begin{bmatrix} \Delta t^2/2 & 0 & 0 \\ 0 & \Delta t^2/2 & 0 \\ 0 & 0 & \Delta t^2/2 \\ \Delta t & 0 & 0 \\ 0 & \Delta t & 0 \\ 0 & 0 & \Delta t \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \in \mathbb{R}^{9 \times 3}. \quad (2.2.4) \]

Here \( \Delta t \) is the time interval between two consecutive sets of data collected from the multi-Kinect\textsuperscript{©} system and the underlying mathematical model for the human joint movement captured by a multi-Kinect\textsuperscript{©} is then given by

\[ x(k) = Ax(k-1) + B\omega(k), \quad (2.2.5) \]

where \( x(k) = [x^1(k) \cdots x^N(k)]^\top \in \mathbb{R}^{9N} \) and \( \omega(k) \in \mathbb{R}^{3N} \) is the disturbance or uncertain bio-kinematic manoeuvres. The measurement matrix \( C \) is

\[ C = \begin{bmatrix} C_1 \\ C_2 \\ \vdots \\ C_M \end{bmatrix} \in \mathbb{R}^{3NM \times 9N}, \quad (2.2.6) \]
where
\[ C_m = \begin{bmatrix}
\bar{C}_m & 0_{3\times9} & \cdots & 0_{3\times9} \\
0_{3\times9} & \bar{C}_m & \cdots & 0_{3\times9} \\
\vdots & \vdots & \ddots & \vdots \\
0_{3\times9} & 0_{3\times9} & \cdots & \bar{C}_m
\end{bmatrix} \in \mathbb{R}^{3N \times 9N} \tag{2.2.7}\]
and \( \bar{C}_m = \begin{bmatrix} R_m & 0_{3\times6} \end{bmatrix} \in \mathbb{R}^{3\times9} \). \( R_m, m = 1, 2, \ldots, M \), is the rotation matrix of \( m^{th} \) Kinect© with reference to the global coordinate system. Let \( \theta_m, \phi_m \) and \( \varphi_m \) be the yaw, pitch and roll angles respectively for the \( m^{th} \) Kinect© for counterclockwise rotation, then the rotation matrix can be computed as,
\[ R_m = R_z(\varphi_m)R_y(\phi_m)R_x(\theta_m), m = 1, 2, \ldots, M \tag{2.2.8}\]
where,
\[ R_x(\theta_m) = \begin{bmatrix} 1 & 0 & 0 \\
0 & \cos\theta_m & -\sin\theta_m \\
0 & \sin\theta_m & \cos\theta_m \end{bmatrix}, \tag{2.2.9}\]
\[ R_y(\phi_m) = \begin{bmatrix} \cos\phi_m & 0 & \sin\phi_m \\
0 & 1 & 0 \\
-\sin\phi_m & 0 & \cos\phi_m \end{bmatrix}, \tag{2.2.10}\]
and
\[ R_z(\varphi_m) = \begin{bmatrix} \cos\varphi_m & -\sin\varphi_m & 0 \\
\sin\varphi_m & \cos\varphi_m & 0 \\
0 & 0 & 1 \end{bmatrix}. \tag{2.2.11}\]
Therefore, the underlying measurement model for the multi-joint tracking can be stated as,
\[ y(k) = Cs(k) + v(k) \tag{2.2.12}\]
where \( y(k) = [y_1(k) \cdots y_M(k)]^T \in \mathbb{R}^{3MN} \) and \( v(k) \in \mathbb{R}^{3MN} \) is the measurement noise. Note that the translation of each Kinect© is corrected, i.e. \( y(k) = \tilde{y}(k) - t_m \), where \( \tilde{y}(k) \) is the actual measurement and \( t_m \) is the displacement vector of each Kinect© estimated (together with the rotational matrices pertaining to orientation) as described in Sec. 2.2.1.
2.2.1 Rotation and translation of the Kinect\textsuperscript{©}

In human motion capture applications, it is vital for the user to have some flexibility in positioning the receivers (i.e due to space restrictions, mitigating occlusions with the nature of the exercise routines). Arbitrary Kinect\textsuperscript{©} positioning instigates the problem of obtaining the rotation matrix of each Kinect\textsuperscript{©} which is crucial for the physical implementation in the 3D case. This can easily be achieved by a minimum of four distinct measurements. Let the measurement be $y^n_m(k)$ for each $s_n(k)$ measurement, $n \in [1, \cdots, N]$, $N \geq 4$ and $\bar{x}^n = [x_{1}^n, x_{2}^n, x_{3}^n]^\top \in \mathbb{R}^3$. Then

$$y^n_m(k) = R_m \bar{x}^n(k) + t_m + e_n(k),$$

where $e_n(k) \in \mathbb{R}^3$ is the residual error. This can be re-arranged as,

$$y_m(k) = \bar{K}(k)r_m + e(k)$$

where, $\bar{K}(k) = [\bar{K}_1(k)^\top, \cdots, \bar{K}_N(k)^\top]^\top \in \mathbb{R}^{3N \times 12}$ with $\bar{K}_n(k) = [I_3 \text{ Diag}\{\bar{x}_n(k)^\top}\} \in \mathbb{R}^{3 \times 12}$, $e(k) = [e_1(k)^\top \cdots e_N(k)^\top]^\top \in \mathbb{R}^{3N}$ and $r_m = [t_m^\top r_m^1 r_m^2 r_m^3]^\top$. Here, $r_m^i$ for $i \in [1, 2, 3]$ are the rows of the $R_m$ matrix, i.e $R_m = [(r_m^1)^\top (r_m^2)^\top (r_m^3)^\top]^\top$. Therefore, the least squares solution $r_m = (\bar{K}(k)^\top \bar{K}(k))^{-1} \bar{K}(k)^\top y_m$ will provide the optimal position vector $(t_m)$ and rotation matrix($R_m$) for the $m^{th}$ Kinect\textsuperscript{©}.

2.3 Model-based State Estimation

The real-time position of the physical joints of the human body is estimated using the aforementioned kinematic description of the dynamics (equation 2.2.5) and the measurements (equation 2.2.12). The natural choice is to use the standard Kalman filter under Gaussian assumptions of measurement and state noise. Particularly for the case of human joint movements during exercise or regular day-to-day movement routines, it is unrealistic to assume a Gaussian distribution for the unknown acceleration components such as $\omega$ in equation 2.2.5 or $\upsilon$ in equation
2.2.12. Therefore, similar to the implementation in [171, 161], robust version of the Kalman filter is used with a more generic assumption on the deterministic uncertainty descriptions[172, 173, 174]. Two commonly used and relevant filters for real-time applications of this nature - a standard Kalman filter and a particle filter - are also implemented. These are compared with our proposed robust version of the Kalman filter which accounts for uncertain human movements modelled in a more generic, deterministic and bounded context.

2.3.1 Kalman filter (KF)

The Kalman filter is used extensively as an optimal state estimator under Bayesian assumptions and is extended to cover non-linear dynamic and measurement models with stochastic, normal or Gaussian distributed noise [175, 176, 177]. This filter was introduced with the purpose of addressing the limitations of other filters in solving the Wiener problems [178]. Various versions of the Kalman filter, ranging from optimal (also called standard) to extended and various unscented versions pertaining to both linear and non-linear systems have covered a large number of application scenarios. The proposed system is linear with position, velocity and acceleration acting as state variables enabling the implementation of a Kalman filter for data fusion.

The dynamic model given in equation 2.2.5 and the measurement model given in equation 2.2.12 are used with the standard Kalman filter assumptions, i.e $\omega(k) \sim \mathcal{N}(0, S_\omega)$ the process noise and $\upsilon(k) \sim \mathcal{N}(0, S_\upsilon)$ the measurement noise. Here, $S_\omega = E[\omega^\top \omega]$ corresponds to the process noise covariance and $S_\upsilon = E[\upsilon^\top \upsilon]$ is the measurement noise covariance. The optimal estimator under these assumptions is given as follows.

$$K(k) = A P(k) C^\top (C P(k) C^\top + S_\upsilon)^{-1},$$

$$\hat{x}(k + 1) = (A \hat{x}(k) + B \omega(x)) + K(k + 1)(y(k + 1)$$
- C\hat{x}(k)),

\begin{equation}
P(k + 1) = AP(k)A^\top + S_\omega - AP(k)C^\top S_\nu^\top CP(k)A^\top
\end{equation}

where \( P \) is the state estimate covariance matrix, \( K \) is the Kalman gain matrix and \( \hat{x} \) is the estimated state of the system.

Although Kalman filtering is pervasive with the underlying assumptions of Gaussian noise or system uncertainty distributions, when the uncertainties deviate from these assumptions the performance degradation can be significant and a more generic uncertainty assumption is warranted.

### 2.3.2 Particle filter (PF)

Particle filters which are a form of sequential Monte Carlo sampling, are widely used for state estimation problems with both linear and non-linear dynamic systems such as the ones considered in [179, 180, 181]. They relax the single Gaussian assumption in the state and the measurement uncertainties allowing the handling of more complex noise situations via sampling from multiple probability densities. When this function is a single Gaussian, the particle filter essentially simplifies to the standard Kalman filter. The application of particle filter in our linear system is described as follows.

1. **System model:** The system and the measurement model are used as described in equation 2.2.5 and 2.2.12 respectively. Inputs consist of \( \omega, \nu \), covariances \( S_\omega = E[\omega^\top \omega] \) and \( S_\nu = E[\nu^\top \nu] \) and unlike in the case of the standard Kalman filtering, the Gaussian assumption has been relaxed.

2. **Establishing Probability Density Function (PDF) of noise:** The probability density function of the state vector, as well as for each particle of the state vector can be denoted as \( p(\cdot) \). Given the past observations \( y(1 : k - 1) \), the posterior probability distribution of the state vector and particles in our linear system with discrete time can be computed as:
\[ p(x(1:k) | y(1:k)) = \frac{p(y(k) | x(k))p(x(k) | y(1:k-1))}{p(y(k) | 1:k)}, \] (2.3.4)

where \( p(y(k) | x(k)) \) can also be written as \( p(y(k) - Cx(k)) \) and it will be used for weight update in the following steps. The probability density function in the particle filter can be defined according to the known noise distribution in the particular system. If it is defined as a Gaussian distribution function, then, as mentioned above, the optimal solution to this filter is the Kalman filter [182].

3. **Particle initialisation:** Generate particle-weight pairs, \( \{p_i(k), W_i(k)\} \) with \( i = 1, 2, \ldots \bar{N} \) for each state vector \( x(k) \) of the system, \( p_i(k) \in \mathbb{R}^{9N} \) and \( W_i(k) \in [0, 1] \). Here, \( \bar{N} \in \mathbb{N}^+ \) is the total number of particles (samples) for each state vector. Additionally, the weight for each particle is initialised to \( 1/\bar{N} \).

4. **Particles and weight updates:** Given the particles at time \( k \), the posterior particles can be updated as:

\[ p_i(k+1) = Ap_i(k) + \omega(k). \] (2.3.5)

Further, the associated weight for this particle will be updated as:

\[ W_i(k+1) = W_i(k)p(y(k+1) | p_i(k+1)) = W_i(k)p(y(k+1) - C p_i(k+1)). \] (2.3.6)

After updating all the particles from time \( k \) to \( k + 1 \), the weights at \( k + 1 \) will be normalised to
\[ W_i(k+1) := \frac{W_i(k+1)}{\sum_{i=1}^N W_i(k+1)}. \]

5. **Computing the estimated state:** With the updated particles and the normalised weights, the estimated state at time \( k+1 \) can be computed as:

\[ \hat{x}(k+1) = \sum_{i=1}^N p_i(k+1)W_i(k+1). \quad (2.3.7) \]

6. **Sampling Importance Re-sampling (SIR) [183]:** To avoid the degeneracy of the filter, particles which are too far away (with lighter weights) should be dropped. For this purpose, a threshold \( \tilde{N} \in \mathbb{N}^+ \) is typically set and when the effective number of particles \( \frac{1}{\sum_{i=1}^N (W_i(k+1))^2} < \tilde{N} \), the re-sampling process is activated and when the particles are being re-sampled the associated weights are redistributed.

However, the computational cost relevant to sampling is a drawback. As evident from step 4, all the particles are updated at every iteration and as the dimension of the state increases, the number of particles grow [184]. Therefore, this significant resource consumption challenges the particle filter use in real-time tracking applications.

**2.3.3 Robust linear filtering (RLF)**

For \( N \) (joint) points, the measurement model is given by the equation 2.2.12, with the kinematic model defined by equation 2.2.5, which describes the motion of the joints with the matrix \( A \) non-singular. Let \( 0 < \tilde{p}_0 \leq 1 \) be a given constant and suppose that the system initial condition \( x(0) \), noise \( \omega(k) \) and the measurement noise \( \nu(k) \) satisfying the following assumption.

**Assumption 1.** The following inequalities with probability \( \tilde{p}_0 \) simultaneously hold:

\[ (x(0) - x_0)^\top \tilde{N} (x(0) - x_0) + \sum_{0}^{T-1} \omega(k)^\top Q(k)\omega(k) \leq \tau. \quad (2.3.8) \]
Here $x_0$ is the given initial state estimate vector, $\tilde{N} = \tilde{N}^\top$ and $Q = Q^\top$ are the positive definite weight matrices, $\tau > 0$ is a given constant associated with the system and $T > 0$ is a given time.

The underlying solution to the state estimation problem involves the following Riccati difference equation\cite{173,174},

$$
F(k+1) = B \left[ \hat{B}^\top S(k) \hat{B} + Q \right]^{-1} \hat{B}^\top S(k) \hat{A},
$$

$$
S(k+1) = \hat{A}^\top S(k) \left[ \hat{A} - F(k+1) \right] + C^\top \hat{U}(k+1) C - \hat{K}^\top \hat{K},
$$

$$
S(0) = \tilde{N}. \quad (2.3.9)
$$

where $\hat{A} \triangleq A^{-1}$ and $\hat{B} \triangleq A^{-1} B$. We also define

$$
\hat{K} \triangleq \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}, \quad (2.3.10)
$$

We now consider a set of state equations as follows,

$$
\eta(k+1) = \left[ \hat{A} - F(k+1) \right]^\top \eta(k) + C^\top \hat{V}(k+1) y(k+1),
$$

$$
\eta(0) = \tilde{N} x_0,
$$

$$
g(k+1) = g(k) + y(k+1)^\top \hat{W}(k+1) y(k+1) - \eta(k)^\top \hat{B} \left[ \hat{B}^\top S(k) \hat{B} + Q(k) \right]^{-1} \hat{B}^\top \eta(k),
$$

$$
g(0) = x_0^\top \tilde{N} x_0.
$$

The above state equation (2.3.11) and Riccati equation (2.3.9) can simply be thought of as a robust implementation of the standard linear Kalman Filter\cite{172} for uncertainties obeying Assumption 1, e.g. see\cite{172,173,174}. Now the main result of this section can be presented.
Remark 1. Notice that the matrices $\hat{U}, \hat{V}, \hat{W}$ are appropriately defined to account for the incomplete information due to occlusions considered in Sec. 2.5.1. For the complete information case, the matrices are evaluated to identity matrices.

Theorem 1. Let $0 < \bar{p}_0 \leq 1$ be given, and suppose that Assumption 1 holds. Then the state $x(T)$ of the system (2.2.5) with probability $\bar{p} \geq \bar{p}_0$ belongs to the ellipsoid

$$E_T \triangleq \left\{ x_T \in \mathbb{R}^{9N} : \| (S(T)^\frac{1}{2} x_T - S(T)^{-\frac{1}{2}} \eta(T)) \| \leq \rho + \tau \right\}$$

where

$$\rho \triangleq \eta(T)^\top S(T)^{-1} \eta(T) - g(T)$$

and $\eta(T)$ and $g(T)$ are defined by the equations (2.3.11). Also, $\rho + \tau \geq 0$ is required.

Proof. Assume (2.3.8) holds to a probability $\bar{p} \geq \bar{p}_0$. Therefore,

$$y(k) = Cx(k) + \nu(k), \quad (2.3.12)$$

where $\nu(k) \triangleq [\bar{n}_1(k) \cdots \bar{n}_M(k)]^\top \in \mathbb{R}^{3MN}$ and the condition

$$\| \nu(k) \|^2 \leq \| \hat{K} x(k) \|^2,$$

holds together with (2.3.8) to a probability $\bar{p} \geq \bar{p}_0$, where $\| \cdot \|$ indicates the vector norm and $\bar{n}_k$ is the measurement noise associated with the Kinect $m \in [1 \cdots M]$. From (2.3.8) and (2.3.13) we obtain that the following sum quadratic constraint that needs to be satisfied,

$$(x(0) - x_0)^\top \tilde{N}(x(0) - x_0) + \sum_{k=0}^{T-1} \omega(k)^\top Q(k) \omega(k) + \| \bar{n}(k + 1) \|^2 \leq \tau + \sum_{k=0}^{T-1} \| \hat{K} x(k) \|^2,$$

with probability $\bar{p} \geq \bar{p}_0$. Now it follows from Theorem 5.3.1 of [174], p. 75 (see also [173]) that the state $x(T)$ of the system (2.2.5), (2.3.12) belongs to the ellipsoid (2.3.11) with probability $\bar{p} \geq \bar{p}_0$.

A point value state estimate can be obtained from the centroid of the bounded ellipsoidal set which is given by

$$\hat{x}(k) = S(k)^{-1} \eta(k). \quad (2.3.15)$$

It can be deduced that the worst error in the estimates can be obtained by using diagonalizing matrices $\Phi, \Xi$ as follows:

$$\hat{x}(k) = \Phi^{-1} \gamma^{-1} \left( \Xi^{-1} \delta + \frac{1}{\sqrt{\rho + \tau}} \Phi S(k)^{-\frac{1}{2}} \eta(k) \right),$$

$$S(k)^{\frac{1}{2}} = \Phi^{-1} \gamma \Phi, \quad \Phi^{-1} \gamma \Phi^{-1} = \Xi^\top \Theta \Xi.$$

(2.3.16)
Here, \( \delta = [0 \cdots \pm \sqrt{a_j} \cdots 0]^T \in \mathbb{R}^N \) with \( a_j = \max_{k=1}^{N} a_k \), where \( a_j \) is the spectral radius of \( \Theta^{-1} \). Note that \( \Upsilon \) and \( \Theta \) are diagonal matrices of appropriate dimensions. The end points of the major axis \((\hat{x}(k)_{\pm})\) of the ellipsoid and the centroid \((\hat{x}(k), \text{the state estimate})\) can be depicted as in Fig. 2.2 for each iteration together with the corresponding uncertainty ellipse that defines the actual bounds of the uncertainty.

![Figure 2.2: Uncertainty ellipse in RLF estimation](image)

It is therefore proved that the estimation errors are bounded in a probabilistic sense when the relevant uncertainties obey Assumption 1. The quadratic constraint given in Assumption 1 accommodates a large class of non-linear and dynamic process noise characteristics. As the Gaussian noise is bounded within the first standard deviation to a probability of \( \bar{p}_0 \approx 0.68 \) and within two standard deviations to a probability of \( \bar{p}_0 \approx 0.95 \) etc, we loose no generality by considering uncertainties satisfying Assumption 1. That is, the Gaussian measurement process and initial condition errors form a special case of Assumption 1 which defines a larger class of uncertainties. The real-time human movement tracking is solved in the linear domain and the proposed algorithm permits very large potential initial errors. No similar proofs exist for the extended Kalman filter (EKF) or the majority of other approaches that employ some form of Taylor-series based approximations. Indeed,
the fact that bounded tracking performance for human movement tracking with arbitrarily large initial condition errors can be proved as a novel contribution.

2.4 Fusion of Information

Information from multiple Kinects provides spatial diversity to measurements and plays a pivotal role in improving the estimation accuracy. Here we look at the advantages of employing multiple Kinects. Denoting state estimate ($\hat{x}$ in equation 2.3.15) with $M$ specially distributed Kinects as $\hat{x}_M$, let the Root Mean Squared Error (RMSE) function between the $M$ Kinect fused trajectory and the ideal one be $\hat{\phi}_M$, i.e $\hat{\phi}_1$ indicates the RMSE between the raw trajectory from the first (reference) Kinect and the ideal trajectory. Let $\hat{\phi}_{M-1:M}$ denote the consecutive improvement with the increase (from $M - 1$ to $M$) in the number of Kinects. Then,

$$\hat{\phi}_{M-1:M} = \frac{\hat{\phi}_{M-1} - \hat{\phi}_M}{\hat{\phi}_{M-1}} \times 100\% \quad \forall \ M \in [2 \cdots \hat{M}]$$

(2.4.1)

where,

$$\hat{\phi}_M = \sqrt{\frac{\sum_{k=1}^{T}(x_M(k) - \hat{x}_M(k))^2}{T}}.$$  

(2.4.2)

Here, $\hat{M}$ is the maximum number of Kinects considered.

2.5 Mitigation of Occlusions and Optimised Positioning

Positioning of the Kinects can affect occlusions as well as the accuracy of the measurements. As the mitigation of occlusions in this form is heavily dependent on the actual human motion, the linear filtering is extended to account for missing information for the generic case. The introduction of spatial diversity in positioning can also be considered to improve the overall measurement accuracy for generic human movements.
2.5.1 Occlusions and incomplete information

One of the key advantages in using the skeletal information from the Kinect is the absence of the data association problem as the joints are already identified. Therefore, the problem of occlusions can be directly addressed under the missing information scenario in model based filtering. This means that $y(k)$ is incomplete or not available for a certain time $k$. Let $\mathbf{m}(k) = [\mathcal{M}^1(k) \mathcal{M}^2(k) \cdots \mathcal{M}^{3MN}(k)]^\top$ be a given vector for $k = 1, 2, \cdots, T$ such that $\mathcal{M}^i \in \{0, 1\}$, for $i = 1, \cdots, 3MN$. Then the matrix $\mathbf{M} \triangleq [\mathbf{m}(1) \cdots \mathbf{m}(T)]^\top$, is referred as the incomplete matrix. With $\mathcal{M}^i$, let us define two sequences of matrices:

\[
\mathbf{E}(k) = \text{Diag}[\mathcal{M}^1(k) \mathcal{M}^2(k) \cdots \mathcal{M}^{3MN}(k)],
\]
\[
\mathbf{e}(k) = [\tilde{\mathcal{M}}^1(k) \tilde{\mathcal{M}}^2(k) \cdots \tilde{\mathcal{M}}^{3MN}(k)]^\top,
\]
(2.5.1)

where $\mathcal{M}^i(k) + \tilde{\mathcal{M}}^i(k) = 1$.

In figure 2.3(a), four cases that lead to missing data are presented. For the first and second cases, if the joint is in these areas, it can only be tracked by one Kinect (Kinect 1 if the joint is in area 1 and Kinect 2 if the joint is in area 2) since it is out of the view of the other one. For case 4 and 5, the joint is in the field of view of both Kinects, but occluded by an object. Therefore, the Kinect located in the same side of the object cannot track the position of the joint. The parameters of using missing information for data fusion are shown in Fig. 2.3(a) as well.

In the Riccati equation 2.3.9 and 2.3.11, $\hat{U}, \hat{V}$ and $\hat{W}$ are defined to account for the incomplete information [174].

\[
\hat{U} = \hat{\mathbf{E}}\hat{W}\hat{\mathbf{E}},
\]
\[
\hat{V} = \hat{\mathbf{E}}\hat{W},
\]
\[
\hat{W} = \begin{cases} 
I - \hat{\mathbf{e}}(\hat{\mathbf{e}}^\top \hat{\mathbf{e}})^{-1}\hat{\mathbf{e}}^\top, & \hat{\mathbf{e}} \neq 0 \\
I, & \hat{\mathbf{e}} = 0
\end{cases}
\]
(2.5.2)

Remark 2. For the case of complete information, $\mathbf{e}$ is the zero vector and $\hat{\mathbf{E}}$ is the
identity matrix. This ensures that $\hat{U}, \hat{V}$ and $\hat{W}$ are evaluated as identity matrices as stated in Sec. 2.3.3.

Target dynamics with missing information in each Kinect\textregistered will be triggered by the absence of measurements directly via the Kinect\textregistered Software Development Kit.

### 2.5.2 Kinect\textregistered optimal positioning

Using the similar approach presented in [185], the general Fisher’s information matrix $I(x) = \nabla_x y^T R_y \nabla_x y = C^T C$. $\det I(x) = 0$, implies that the position measurement scheme has no orientation dependency in an information theoretic sense under Gaussian assumptions. This is unlike the case for the angle, distance and time delay of arrival measurement schemes. Considering the 2D orientation scenario under a equidistance exercise monitoring setting as depicted in Fig. 2.3(b), the position measurement can be considered in an angle measuring context for the purpose of information theoretic evaluation ($x_1^n = -r \sin \alpha$, $x_2^n = r \cos \alpha$, $r$ – a constant indicating the range specified by the Kinect\textregistered to maximise the field of view).

The determinant of the Fisher’s information matrix (in an angle measurement context) is maximum (see Fig. 2.3(b)) when the $\theta = \pi/2$. Indeed the $XY$ plane of each Kinect\textregistered is orthogonal to each other for the two Kinect\textregistered case causing a $\pi/2$ angle subtended at the joint being tracked. For the cases of an any arbitrary number greater that 2 Kinects\textregistered, an extensive positioning analysis is given in [185]. Note that this reasoning is based purely on information theoretic assumptions and as the potential occlusions depend on the exercise routine, the receiver positions may need to be adjusted accordingly. The improvements due to the suppression of occlusions are much more significant than the gain due to Kinect\textregistered’s optimal positioning (Notice the improvement due to two Kinects\textregistered as opposed to one in Fig. 2.10 and the improvement due to missing data accounting by the RLF). Therefore, after positioning the Kinects based on the empirical knowledge to minimise the occlusion, the position estimation approach described in Sec. 2.2.1 can be used to estimate
the rotation matrices and translation vectors required for fusion of information.

Figure 2.3: Fig. 2.3(a) illustrates four instances where the two Kinect system may have missing data and another instance without any occlusion. Here, the incomplete matrix parameters of the RLF based two Kinect fusion are explicitly stated. Fig. 2.3(b) shows the optimal positioning of two Kinects.
A hypothetical scenario is used in a computer simulated environment as well as in a practical scenario with a hardware setup to test and validate our theoretical assertions. In evaluating the RMSE for $M$ Kinects® ($\hat{\phi}_M$) from equation 2.4.2, in computer simulations the known ideal trajectory $\hat{s}(k)$ is used. In the hardware experimentation, the VICON captured trajectory is used as the ground truth. In order to convey these ideas on multi-Kinect® fusion, one Kinect® ($\hat{K}_1$) is considered as the reference. $\hat{\phi}_M$ is averaged over number of runs and the average RMSE is denoted as $\hat{\phi}_M^a$ ($M = 1, 2, ..., 20$ for computer simulations and $M = 1, 2, 3$ for the hardware experimentation) which in turn is used to calculate the average relative improvement percentage ($\hat{\phi}_{M-1:M}^a$). This is used to demonstrate the contribution of adding a Kinect® to the multi-Kinect® system in terms of reducing the RMSE. Note that the fused trajectories captured by the Kinect® system should be rotated and translated to the coordinate system of the VICON system when calculating the RMSEs.

In addition to the overarching assertion of superior performance in terms of accuracy and robustness in the proposed linear filter, improvement due to multiple
Kinect\textsuperscript{©} sensor fusion, mitigating the effects of occlusions via extending the idea of missing information were evaluated in simulated experiments as well as in a hardware based practical context.

When selecting tuning parameters for various approaches considered, in a generic sense, parameters that optimised the mean squared error between the estimated states and the actual state was used. The actual state was captured from the VICON system in the real data experiment as the ground truth. This was obtained using a common collection of training datasets. Indeed the interval for the search was based on empirical knowledge of the underlying experiment. For real-data experiments, a smooth curve was fitted to the sample training trajectories and the standard deviation of the measurement noise with spectral densities ($S_\omega$ and $S_\upsilon$) were obtained under Gaussian assumptions. Obviously for the simulated examples the specific noise distributions were assumed and considered as additive uncertainties to the perfect measurements. With this reasoning, an empirical understanding of the magnitude ($\hat{s}$) of $S_\omega$, $S_\upsilon$ and $P(0)$ is acquired. A sufficiently large interval $\hat{s}I_9[10^{-5}, 10^{-4}, \cdots, 10^4, 10^5]$ was chosen to test each combination of $S_\omega$, $S_\upsilon$ and $P(0)$ in computing the error (difference between the fused and the ideal trajectory) and the combination that provided the smallest RMSE was selected. This was further optimised by employing the Expectation Minimisation (EM) algorithm using all the training datasets. Indeed the underlying assumption here is similar noise and uncertainty characteristics across the training and testing exercises. The same noise distribution is assumed for all the filters with the standard Kalman filter being observed as a special case of the RLF with the interpretation of probabilistically bounded Gaussian noise. For the particle filter, when $\omega(k)$ in (2.3.5) and $p(y(k+1) - Cp_i(k+1))$ in (2.3.6) was computed, the standard deviation of $\omega$ and $\upsilon$ ($S_\omega$ and $S_\upsilon$) could be obtained from Kalman filter. As the number of particles increase, a better approximation is reached. However,
due to significant computational cost, the number of particles only raises from 1000 to 10000 with a step size of 1000. In robust linear filtering, the combinations of $\tilde{N}$ and $Q$ which were in the range of $[10^{-10}I_{9N}, 10^{-9}I_{9N}, \cdots, 10^0I_{9N}, 10^{10}I_{9N}]$ and $[10^{-10}I_{3N}, 10^{-9}I_{3N}, \cdots, 10^0I_{3N}, 10^{10}I_{3N}]$ were considered in order to obtain the combination that gave the smallest RMSE. Further, if $\tilde{N} = 10^nI_{9N}$ and $Q = 10^9I_{3N}$ gave the smallest RMSE, the range $([10^{n-1}I_{9N}, 2 \times 10^{n-1}I_{9N}, \cdots, 10^nI_{9N}, 2 \times 10^nI_{9N}, \cdots, 10^{n+1}I_{9N}])$ was refined for $\tilde{N}$ (and similarly for $Q$) and repeated the process until the improvement in RMSE was not sufficiently significant. In the simulation based example, since the standard deviation of the noise was known, that information was directly used to determine parameters for the Kalman filter and particle filter, while the same approach described above was used to determine the parameters for the robust linear filter. Indeed all the data in the real experiment was collected from the same Kinects® positioned in the same environment and hence our assumption of similar distribution over different filters and training and testing scenarios was justified.

### 2.6.1 Computer simulations

#### Improvement due to fusion

A helical trajectory with 1000 samples and a total length of 6 meters was generated to represent the motion of a joint in the human body as the ground truth or the ideal trajectory of our experiment. Using this trajectory, another 20 trajectories with bounded noise were generated to represent data captured from 20 different Kinects® with known positions and orientations. To illustrate the robustness of these algorithms, three levels of Signal-Noise Ratio(SNR)s, 20, 25 and 30, were considered.

The noisy trajectories collected from the Kinects® in each round of the experiment would be fused through each of the filters. For example, the trajectories gained from Kinects® were fused into one trajectory as per our basic postulation.
leading to the model in equation 2.2.5 with all the filter implementations and this would be conducted for up to 20 Kinects®. The RMSEs (\(\hat{\phi}_M, M = 1, 2, \ldots, 20\)) were computed between these 20 trajectories and the ground truth to compare the performance of the multi-Kinect® systems. Furthermore, the relative improvement percentages between \(M - 1\) Kinect® fused trajectory with \(M\) Kinect® fused trajectory were computed as described in Sec. 2.4 (see equation 2.4.1).

The RMSE and relative improvement percentage diagrams are shown in Fig. 2.5. It can be seen that RMSEs between fused and model trajectories gradually decrease with the increasing number of the Kinects® irrespective of the noise level considered. For instance, in Fig. 2.5(a), the RMSEs of fused trajectories corresponding to KF, PF and RLF drop from 0.168m, 0.153m and 0.122m to 0.039m, 0.037m and 0.034m with the increase in the number of Kinects® from 1 to 20. Alternatively, the increased number of Kinects® enabled countering the noise effects in the fused trajectories. Moreover, it is obvious that robust linear filter outperformed the others by generating fused trajectories with the smallest RMSE demonstrating its superiority over the other filters considered.

From Fig. 2.5(b), 2.5(d) and 2.5(f), while RMSEs of fused trajectories decrease with the increase in the number of Kinects®, relative improvement percentages (\(\hat{\phi}_{M-1,M}\)) decayed almost in an exponential fashion. For an example, in the three cases with different SNRs, the relative improvement percentages of RLF with two Kinects® are always more than 25%, while this number drops to around 15% when the number of Kinects® raises to three and a similar behaviour can be expected from the other two state estimators.

**System cost and complexity in applied tele-rehabilitation**

Additional Kinects® indeed improve the accuracy of human kinematic estimations and also incurs added cost. Assuming the cost of a Kinect® is \(\tilde{C}\), the combined
Figure 2.5: Filter performance improvement against multiple Kinects\textsuperscript{©} subjected to different uncertainty levels
cost function optimisation is,

\[
\min_{M \in \mathbb{N}^+} \Psi(M) \quad (2.6.1)
\]

where,

\[
\Psi(M) = \alpha \Phi_{M-1:M}^2 + (1 - \alpha)\overline{CM}^2.
\]

Additionally, $\Phi_{M-1:M} \in [0, 1]$ and $\overline{CM} \in [0, 1]$ are the normalised values of $\frac{1}{\Phi_{M-1:M}}$ and $\overline{CM}$ over $M$. Here $\alpha \in [0, 1]$ is the relative weighting parameter. To illustrate the relationship between $\Psi(M)$, $\alpha$ and $M$, a simulation experiment was conducted for the case of robust linear filtering under the two different scenarios of interest; occluded and non-occluded cases. For the former, it is assumed that there is no missing data, while for the latter, the second half of some datasets were intentionally set with missing data.

As illustrated in Fig. 2.6, for both cases (with and without occlusion), the smaller $M$ is associated with smaller $\Psi(M)$ for a range of $\alpha$. Therefore, in order to ensure a compromise in the cost and system complexity, it is reasonable to select a small $M$ for applications in tele-rehabilitation. It is noteworthy that the change of $\alpha$ exerts little influence on $\Psi(M)$ because $\Phi_{M-1:M} \approx \overline{CM}$ for $M = 19$.

![Figure 2.6: The relationship between $\Psi(M)$, $\alpha$ and $M$ in the cases with and without occlusion. Robust linear filtering was utilised for data fusion.](image)
2.6.2 Hardware experiments

Improvement due to multiple Kinects®

The hardware experiment was conducted with three non-linearly placed Kinects® (30 Hz sampling rate and labelled as $K_1$, $K_2$ and $K_3$) and a VICON system (250 Hz sampling rate) set up as depicted in Fig. 2.7. The distance between the subject and each Kinect® was approximately three meters to ensure that each Kinect® could capture and generate the complete skeleton of the subject. The orientation of each Kinect® is placed with empirical knowledge to minimise occlusion. At the same time, a marker was placed on the right wrist of the subject so that the positions of the right wrist could be captured by the VICON system simultaneously.

![Experimental setup: VICON and Multi-Kinect® system](image)

Figure 2.7: Experimental setup: VICON and Multi-Kinect® system

During the experiment, three healthy male subjects conducted three types of repetitive movements to simulate a relatively longer period of exercises: swinging (S) right arm for approximately one minute, reaching (R) forward with the right arm and drawing circles (C) in front of the body with the right arm for approximately thirty seconds. The detail of these gestures such as the total trajectory length for each type of movement performed by the three subjects, and corresponding RMSE of the trajectories from individual Kinects® are shown in Table 2.1.

Since the multi-Kinect® system and the VICON system are two independent standalone systems, certain implementation related clarifications are needed. Firstly,
although it is possible to trigger the two systems simultaneously, in this experiment, the data captured from the two systems was synchronised manually using the kinematic event itself. More specifically, when two systems began capturing data, the subjects remained steady for a few seconds prior to commencing the exercises for synchronisation purposes. This was purely for comparison (with VICON) purposes and not required in the actual application. The starting point of the movements can be clearly observed from the captured data, which was used to synchronise the multi-Kinect® system and the VICON system. Secondly, data from the multi-Kinect® system was resampled at 250 Hz, again for comparison purposes with VICON. Thirdly, during the evaluation phase, trajectories captured from the multi-Kinect® system was compared to that from the VICON system which was used for calibrating the two systems. In our experiment, the algorithm introduced in Sec. 2.2.1 was utilised to compute the rotation matrix and translation vector between each Kinect® and also with the VICON system.

The time used to compute the fusion of each frame with different approaches in the computer with Intel® Core(TM) i5 (3.0Hz) and 8 GB memory was also recorded for evaluation as a real-time system.

The results of the hardware experiment are shown in Fig. 2.8 and 2.9. The legends in both figures are indicated in the form of \( (\text{number of Kinects}^{\circledR}) - (\text{filter name}) \). For example, \( 3K - KF \) denotes the three Kinects® fused by the Kalman filter.

Fig. 2.8 illustrates the root mean square errors due to multi-Kinect® fusion. Generally, the RMSEs of three Kinect® fusion (with lighter colours) with all the
Figure 2.8: Averaged RMSEs ($\hat{\phi}_M$, $M = 2, 3$) over the same type of exercises conducted by the three subjects.

Figure 2.9: Averaged relative improvement percentages ($\hat{\phi}_{M-1:M}$, $M = 2, 3$) of multi-Kinect fusion.
combinations of motion types and fusion approaches are smaller than that of two Kinect® fusion (with darker colours). Here the smaller RMSE indicates the proximity between the fused trajectory and the captured trajectory from the VICON system. Alternatively, the fused trajectory with three Kinects® have less noise than in the case for two. In other words, fusion of more Kinects® counter noise, resulting in smaller RMSEs with respect to trajectories captured from the VICON system.

This result corresponds to Fig. 2.5(a), 2.5(c) and 2.5(e) in computer simulations. In addition, robust linear filtering generates the smallest root mean square errors in all the cases and varying number of Kinects®, while particle filter gives the largest ones except for fusion of swing motion with three Kinects® (0.06cm smaller in this case compared to Kalman filter), which confirms the superior performance of robust linear filtering.

Furthermore, Fig. 2.9 depicts the relative improvement percentages of two (darker colours) and three (lighter colours) Kinect® fusion. It is evident that the two Kinect® fusion generates much higher relative improvement percentages (ranging from 10.28 % to 18.92 %) than the latter (between 0.20% and 7.24%) at least for the movement types and fusion methods considered.

Considering the average computational time of 4.2123e-5 second per frame (SPF) for RLF, compared to 3.016e-4 SPF and 0.348 SPF for KF and PF respectively, it is clear that the RLF is more suitable for real-time application. The Particle Filter needs rigorous optimisation prior to any consideration of real-time use.

**Robustness due to occlusions**

A real-data experiment had been conducted to illustrate the capabilities of the proposed approach. This deals with missing information due to occlusions of body parts in monitoring human movements which is considered as one of the major
limitations in using a single Kinect® as a motion tracking device. In this experi-
ment, we not only recorded the positions of a joint during a time frame, but the
tracking status of this joint was recorded from the Kinect® SDK. Three subjects
were required to perform a swing motion repeatedly around a minute. During the
first half of the experiment, the monitored joint (right wrist) was within the field of
view of both Kinects®, ensuring that there was no missing data during this period.
In contrast, during the second half of the exercise routine, all the subjects were
required to conceal their right arm behind their body to ensure only the Kinect®
on their right side could capture the wrist position information with the Kinect® on
their left side experiencing missing data. Data collected from both Kinects® were
fused by the robust linear filtering with and without missing information accounted
and then compared to the trajectory captured from the VICON system. Due to
very low computational efficiency, the data fusion with the particle filter has not
been performed.

![Error Graph](image)

Figure 2.10: Errors of two Kinect® fusion with missing data. The temporal interval
between consecutive samples are 1/240 second.

The results of this experiment is shown in Fig. 2.10 and the comparison be-
tween our proposed approach and those introduced in [186, 187, 188] is shown in
Table 2.2. Fig. 2.10 indicates the error when using one Kinect® and also when
using two Kinects® fused by robust linear filtering with (Sec. 2.5.1) and without
missing information accounted. The error was computed as the Euclidean distance
Table 2.2: Comparison (in the presence of missing information) of Accuracy and execution times between the proposed RLF and the ones introduced in the references [186] [187] and [188].

<table>
<thead>
<tr>
<th>Subject 1 (cm)</th>
<th>Subject 2 (cm)</th>
<th>Subject 3 (cm)</th>
<th>Speed (SPF)</th>
</tr>
</thead>
<tbody>
<tr>
<td>RLF</td>
<td>7.77</td>
<td>8</td>
<td>10.16</td>
</tr>
<tr>
<td>[186]</td>
<td>15.7</td>
<td>13.49</td>
<td>17.48</td>
</tr>
<tr>
<td>[187]</td>
<td>9.14</td>
<td>10.12</td>
<td>11.82</td>
</tr>
<tr>
<td>[188]</td>
<td>12.46</td>
<td>15.94</td>
<td>16.14</td>
</tr>
</tbody>
</table>

between the trajectories deduced from the Kinect® system and the VICON system. It is evident that in the first half of the trial (with no missing data), the position estimation error for the cases of one Kinect®, RLF with and without missing information accounted, are relatively similar as the Kinects® could track the monitored joint. However, in the second half, the position estimation error for the case of one Kinect® is increased significantly since it was no longer able to track the joint due to occlusions. As RLF without accounting missing information can still utilise the information from the second Kinect®, it is able to track the joint with the errors still under approximately 0.2 meters. Indeed the RLF with missing information accounted further improves the estimation accuracy by restricting the error to less than 0.1 meters. Table 3.5 depicts the average RMSE and processing time per frame for the three approaches. As evident, RLF with missing information accounted, always has the smallest RMSE compared to other relevant approaches using skeleton information. When comparing the execution times, all these approaches were implemented in Matlab 2014b® and executed under the same conditions. The approach in [186] required 1.03e-7 seconds, which was faster compared to RLF. The approaches in [187] and [188] required 0.519 and 0.0686 seconds per frame respectively. As Kinect® can only provide one skeletal frame in 0.33 seconds, our approach and the one in [186, 188] can be considered for real time implementations.
2.6.3 Discussion

The computational cost in using particle filtering or a point cloud based approach is significant in comparison to the model based filtering as [189] uses 0.5 of CPU time and also our simulations indicate an inferior performance with particle filtering. Skeleton based measurement schemes such as ours require a more sophisticated model based filtering unlike the simple averaging based approach used in [186] which reports a lesser accuracy of 0.13m compared to the 0.08m accuracy in our proposed method. Although in computer simulations, the particle filter out performs the Kalman filter, the converse in the hardware experiment is observed. This is due to the fact that 10000 particles used in the hardware experiment are not sufficient for this multi-Kinect© fusion. Indeed the computational cost and the resource consumption have already made particle filter not suitable for real-time applications of this nature.

Particularly in an exercise monitoring scenario, the occlusions can predominantly be caused by other parts of the patient’s body. Occlusions have been considered as a missing information scenario and the positioning of the Kinects© need to ensure these effects are minimal. Nevertheless, some form of occlusions are likely to exist and the severity of these are dependant on the actual exercise routine. The optimal positioning of the Kinects© can be achieved based on the information theoretic reasoning utilising the Kramer Rao bound concepts as in [185]. Optimal positioning of receivers subjected to missing information is considered as an optimal angular positioning exercise with a fixed range to maximise the field of view. Even so, the occlusions are significantly dependant on the actual nature of the exercise movements. The resulting positioning outcomes are based on the exercise dynamics and hence the positioning needs some flexibility to cater for the nature of the exercises. For practical use, arbitrary positioning of receivers provides clinicians
greater flexibility while minimizing occlusions. Robust linear filtering with missing information is more suitable for multi-Kinect© fusion for its lower RMSEs and shorter computational time. The particle filter exhibits shortcomings despite being widely used in the multi-sensor fusion arena. From both the simulation and the real-data experiment, it is obvious that increasing the number of Kinects© improves the measurement accuracy of the overall multi-Kinect© system albeit increasing the cost as well as the complexity in setting up and managing the system. Indeed, the number of Kinects© used is a compromise between counteracting factors, such as the cost, system complexity, overall accuracy and the complexity in the exercise routine. The significant improvement in increasing to two Kinects© is deemed as a notable aspect to consider for certain applications.

2.6.4 Application in tele-rehabilitation

An ideal good application of multi-Kinect fusion is improving the measurement and assessment accuracy in tele-rehabilitation. Based on the aforementioned theoretical assertions, a novel cloud based remote rehabilitation exercise monitoring system with bio-feedback has been implemented and is currently being deployed for relevant formal clinical trials. The conceptual architecture is depicted in the Fig. 2.11, where the crucial aspects are enhanced accuracy and bio-feedback with performance measurements; intrinsic requirements in providing cost effective remote rehabilitation without jeopardising the quality of care. Indeed we endeavour to capture human motions more effectively using multiple Kinect© fusion to develop a more clinically effective and commercially viable system.

Let the normalised trajectory of the prescribed exercise routine captured from the clinician denoted by, $\mathcal{J}(k) \in \mathbb{R}^{9N}$. Then the patient exercise performance measure (Ψ), can be denoted in the mean square sense as,

$$\Psi = \frac{\Psi_0 - \sum_{k=0}^{T-1} \mathcal{J}(k) - \hat{s}_i(k)\mathcal{J}(k) - \hat{s}_i(k))}{\Psi_0}$$

(2.6.2)
where $\Psi_0 = \sum_{k=0}^{T-1} J(k)^\top J(k)$ and $\hat{x}_\perp$ denotes the normalised state estimate from equation 2.3.15. The ideal exercise trajectory $J(k)$ captured at the therapist workplace is used as the benchmark in the patient system at home as well as to guide the patient’s exercise motion. Performance index $\Psi$ deduced from the patient trajectory $\hat{s}_\perp$, is the quantitative measure the therapists use to monitor the improvements due to the exercises performed remotely. It is noteworthy that the performance index used here is an example. In Chapter 4, some other encouraging features, such as approximate entropy of shape model and instantaneous acceleration, are introduced. Additionally, more features utilised for automating patients’ performance assessment in rehabilitation exercises and activities of daily living can be found in literature listed in Sec. 1.6.2.

The advantages of using multi-Kinect© fusion for tele-rehabilitation was illustrated. We assumed that the motion trajectory from a professional clinician captured by the VICON system is the model motion and the patient is provided with this trajectory to follow remotely. In our experiment, since the motions of three subjects were captured by the VICON and multi-Kinect system simultaneously, all these subjects can attempt to gain 100% in the performance measure. From Fig. 2.12, RLF with missing information accounted gave performances closest to 100
% with 99.83%, 99.82% and 99.49%, while RLF without accounting for missing information provided 99.74%, 99.66% and 99.18%.

Figure 2.12: Average RMSE and performance measure of two Kinect fusion with missing data.

2.7 Summary

Being an indispensable component of the human kinematic performance assessment systems, motion capture devices should robustly and accurately capture the human motions. In this study, joint positions collected from multiple OBMCDs, such as Kinects, are fused with model based linear filtering (robust linear filtering) to overcome two major issues in affordable OBMCDs, including the limited accuracy and field of view. The comparison between the proposed approach and the other two commonly utilised method (Kalman filter and particle filter) in this study has confirmed the superiority of the RLF. Simultaneously, the evaluation of the multi-Kinect system against the commercially available VICON system illustrates the significant improvements from single Kinect under with and without missing data scenarios resulted from occlusions.

However, there remain two limitations in this work. One is that the parameters utilised in achieving acceptable results are selected manually, which leads to a problem that, in real-world applications, poorly tuned parameters may lead to inaccurate estimation, while a large amount of time should be spent to find the best parameters that give the best results. The other limitation is that there is no
human kinematic model considered. In this study, all the joints in a human body are considered independent with each other in the dynamic model used in all three filters. It is expected that the fusion result is highly likely to be improved if the kinematic model of human body can be taken into consideration in the dynamic system.

As a result, two potential approaches can be considered in the near future. Firstly, similar to Kalman filter and particle filter, algorithms can be investigated to estimate the parameters used by robust linear filtering. During this process, the prior knowledge about the human motion, as well as the tracking device can be taken into account to estimate the parameters more accurately. Secondly, parameters, such as bone length, angular range of movement between limbs etc., in human body kinematic models can be integrated into the dynamic model. However, one potential problem that may emerge is that the whole system may no longer be a linear system. Therefore, suitable filters should be selected for information fusion.
Chapter 3

Human Motion Trajectory Encoder

3.1 Introduction

Actions as much as language can express ideas and communication not only in humans but also by various other species to the extent that recent work suggests there are strong links between motor and language areas of the human brain[190]. While there are a few studies that have shown the application of the language of action in various areas [191, 192, 193, 194], it is also a key to apply pervasive motion sensing technologies to clinical kinematics, which is rarely studied.

However, there still remain challenges in developing formal descriptions and robust computational procedures for the automatic interpretation and representation of human actions. The majority of studies employed a variety of human motion encoders to recognise or decompose general movement, such as reaching, waving hands, jumping, walking and so on. Few of them investigate details in each general movement, for example, the even smaller atomic components included in these general movements which are of importance for syntactic and structural descriptions of human movements in detail, especially in clinic and rehabilitation environment, where the details of movements of body parts require a form of motion language or, at least, syntax.

A detailed example of current work is the POETICA system of Aloimonos et al.
which integrates formal generative action language with sensor (IMU) velocity and acceleration measures. Specifically, Guerra-Filho et al. [196] used velocity and acceleration to represent and divide human actions into atomic motions as the phonemes for a human action language (HAL). They also extracted atomic components using velocity, acceleration and joint angle - motion "phonemes". In Sant’Anna et al. [197] analysed of gait motion in the elderly was performed using a motion language based on accelerometers. Moreover, joint angle have also been used pervasively to find primitives of human actions [198, 199].

It is interesting to note that in some cases motion primitives belonging to the same type of action may share the same motion trajectory with various dynamics and orientations, such as shaking hands, while sometimes, although the trajectories of two motions are similar, they are two different motions due to various dynamics, like walking and running. Therefore it is most important to separate the shape from the dynamics of human actions. Further it is important to have motion encoders that are capable of uniquely encoding the shape and dynamics invariant to the limbs absolute position and pose.

In light of the above, this chapter develops three contributions to these issue.

1. A two-component action encoding model is proposed to separate a complex motion trajectory into its shape and dynamics, which extending earlier work of Caelli et al. [200] where only preliminary modelling and signal processing was performed.

2. A spatial indexing scheme is introduced to index the trajectory of a motion since the shape component of the proposed model is based on the arc-length of a trajectory. Compared with the temporal indexing scheme, a shape model can be computed more precisely by using spatial indexing scheme.
3. Three commonly used filters, including least-square Gaussian filter, Savitski-Golay filter and optimal Kalman filter, are compared to choose the best one for our proposed encoding model.

4. The bridge between our work and canonical actions is built to show the potential application of the proposed encoding model, which is shown in the last experiment by decomposing some complex motions into pre-defined atomic motions.

More generally, this work is embedded in a more general three-level description framework (see Fig. 3.1) involving $L_1$ sensor, $L_2$ limb and, $L_3$, complete action descriptions - and their relations - that fit within specific contexts. In our context, this is clinical kinematics such as rehabilitation and patient monitoring. Albeit the main focus here is the development of robust computational methods for computing unique, invariant and useful sensor “atomic states”, or, motion primitives, for the sensor ($L_1$) level and how they apply to other levels. A simple example. Consider the right elbow flexion exercise involving the planar movement of the right arm as described in PhysioAdvisor[201].

Figure 3.1: Three-level syntactic description framework for building language for human action

*Bend and straighten your elbow as far as possible without experiencing pain and provided you feel no more than a mild to moderate stretch. Repeat 10 - 20 times provided the exercise is pain free.*
This implies that the arm limbs have required states as well as the sensors measuring the displacement in ways that would enable automated evaluation if the exercise is performed correctly. For example, it is more or less independent of speed, smoothness, etc.

The rest of the chapter is arranged as follows: The two-component encoding model is introduced in Sec. 3.2, followed by two trajectory indexing schemes, namely temporal and spatial schemes, to compute the proposed encoding model. In Sec. 3.4, three filters are explored and compared to deal with noise in the raw data, followed by the introduction of switching continuous hidden Markov model in Sec. 3.5 to demonstrate the benefits of the proposed encoder in action recognition. In the next section, a potential human action alphabet for human actions is introduced based on this encoder and experimentally explored in Sec. 3.7. The Conclusion follows.

3.2 The Two-component Encoder Theory

In Differential Geometry the shape of a curve is described by the Frenet-Serret equations [202, 203], a system of linear differential equations. Let \( \mathbf{\bar{p}} \) be a point in \( \mathbb{R}^3 \), with position vector \( \mathbf{r} = [\bar{x}, \bar{y}, \bar{z}]^\top \), moving along a trajectory \( \gamma \); the simplest form of the Frenet-Serret equations arises when \( \gamma \) is parametrised by arc-length \( s \in \mathbb{R}^+ \), whose infinitesimal element \( ds \) is defined as

\[
ds = \sqrt{d\bar{x}^2 + d\bar{y}^2 + d\bar{z}^2}.
\] (3.2.1)

In this parametrisation the tangent vector

\[
\mathbf{u}(s) = \frac{d\mathbf{r}(s)}{ds} = \left( \frac{d\bar{x}(s)}{ds}, \frac{d\bar{y}(s)}{ds}, \frac{d\bar{z}(s)}{ds} \right)
\] (3.2.2)

is the unit vector and its derivative is a vector normal to \( \mathbf{u}(s) \) whose norm is the curvature \( \kappa(s) \); this result is the first Frenet-Serret equation

\[
\frac{d\mathbf{u}(s)}{ds} = \kappa(s) \cdot \mathbf{n}(s),
\] (3.2.3)
where $\mathbf{n}(s)$, the unit vector, corresponds to the principal normal vector. Curvature thus measures the amount of arc-rate of change of $\mathbf{u}(s)$ and can be computed via the formula

$$\kappa(s) = \left\| \frac{d\mathbf{u}(s)}{ds} \right\|. \quad (3.2.4)$$

Vectors $\mathbf{u}(s)$ and $\mathbf{n}(s)$ thus define a plane, the osculating plane, and the orientation of $\gamma$ in space can be now completely specified once a third unit vector is given, the binormal vector, defined as

$$\mathbf{b}(s) = \mathbf{u}(s) \times \mathbf{n}(s). \quad (3.2.5)$$

The system of Frenet-Serret equations can now be completed:

$$\frac{d\mathbf{n}(s)}{ds} = -\kappa(s)\mathbf{u}(s) + \tau(s) \cdot \mathbf{b}(s), \quad (3.2.6)$$

$$\frac{d\mathbf{b}(s)}{ds} = -\tau(s)\mathbf{n}(s). \quad (3.2.7)$$

where the torsion $\tau(s)$ measures how the curve winds out of the plane,

$$\tau(s) = -\frac{d\mathbf{b}(s)}{ds} \cdot \mathbf{n}(s). \quad (3.2.8)$$

These equations are the basis for the fundamental theorem of curves which proves that a curve parametrised by arc-length $s$ is uniquely defined up to rigid motion and that $\kappa(s), \tau(s)$ encode the curve in a way invariant (or blind) to rotations and translations. This is because the position vector $\mathbf{r}$ can be computed at each $s$ by first integrating the Frenet-Serret equations to obtain the tangent, normal and binormal vectors given an initial orientation (thus losing the invariance to rotations) and next by integration of (3.2.2); the resulting vector, $\mathbf{r}$, depends on the initial position i.e. it is not invariant to translations.

The role of $\kappa(s)$ and $\tau(s)$ can be better understood if one considers the Taylor expansion around, say, $s = 0$, retaining just the first four terms

$$\mathbf{r}(s) = \mathbf{r}(0) + s\mathbf{u}(0) + \kappa(0)\frac{s^2}{2}\mathbf{n}(0) + \kappa(0)\tau(0)\frac{s^3}{6}\mathbf{b}(0) + O(4). \quad (3.2.9)$$
The first two terms provide the best linear approximation of the curve near $r(0)$ and the curvature at $s = 0$ modulates the departure from linearity in the osculating plane. Finally the torsion appears in the last and smallest term of the expansion controls the deviation of $r$ from the osculating plane. Note that, in a neighbourhood of $s = 0$ the largest variation is along $u$ followed by lesser variations along $n$ and $b$ respectively. Suppose the curve from a set of experimental data is sampled. If the neighbourhood is small enough then $u, n, b$ are, respectively, the first, second and third principal direction of the sample distribution.

Curvature and torsion encode solely the shape of the curve and provide nothing about the speed of the point $\vec{p}$, since the arc-length parametrisation uses velocity $v$ as a unit vector. In order to have a complete, invariant description of the motion encoding both the shape and the kinematic dynamics of point $\vec{p}$, the speed $v$ at each position should be known - noting that $v$ is a scalar and, as such, is invariant under rigid motion. In other words, the relation between time $k$ and $s$ is needed, which is provided by the equations

$$v = \left| \frac{ds}{dk} \right|. \quad (3.2.10)$$

This becomes more explicit when considering temporal parametrisation of $\gamma$.

### 3.3 Encoding Methods

It is possible to compute the trajectory shape and dynamics (kinematics) using two different indexing schemes for $\gamma$: one with respect to discrete time sampling, $(dk)$, or with respect to discrete spatial/distance $(ds)$ - as follows.

#### 3.3.1 Temporal index

This method assumes that each position $[\bar{x}, \bar{y}, \bar{z}]$ is indexed by time resulting in a position vector $[200]$:

$$r(k) = [\bar{x}(k), \bar{y}(k), \bar{z}(k)]^T, \quad (3.3.1)$$
where \( k = cdk, c = 0, \ldots, T \) for a constant \( dk \) while the corresponding \( ds \) is variable.

From \( r(k) \) quantities such as \( \{ \kappa(k), \tau(k) \} \) and \( v \) can be computed, via finite differences schemes, as follows: compute first

\[
v(k) = [v_x(k), v_y(k), v_z(k)] \approx [\frac{d\bar{x}(k)}{dk}, \frac{d\bar{y}(k)}{dk}, \frac{d\bar{z}(k)}{dk}],
\]

then

\[
a(k) = [a_x(k), a_y(k), a_z(k)] \approx \left(\frac{dv_x(k)}{dk}, \frac{dv_y(k)}{dk}, \frac{dv_z(k)}{dk}\right),
\]

and

\[
\frac{da(k)}{dk} \approx \left(\frac{da_x(k)}{dk}, \frac{da_y(k)}{dk}, \frac{da_z(k)}{dk}\right).
\]

Thus \( v \) can be obtained from

\[
v(k) = \left[\left(\frac{d\bar{x}(k)}{dk}\right)^2 + \left(\frac{d\bar{y}(k)}{dk}\right)^2 + \left(\frac{d\bar{z}(k)}{dk}\right)^2\right]^{1/2},
\]

and \( \kappa(k), \tau(k) \) from

\[
\kappa(k) = \frac{\|v(k) \times a(k)\|}{v(k)^3}, \quad \tau(k) = \frac{(v(k) \times a(k)) \cdot da(k)/dk}{\|v(k) \times a(k)\|^2}.
\]

respectively. However, computations of (3.3.5) and (3.3.6) requires performing up to third order numerical differentiation and so filtering must be performed to deal with noise.

### 3.3.2 Spatial index

Here the trajectory shape is separated from its dynamics right from the initial encoding by considering \( r \) as a function of \( s \). The coordinates

\[
r(s) = [\bar{x}(s), \bar{y}(s), \bar{z}(s)]^T
\]

are sampled at values of \( s = \bar{cd}s, \bar{c} = 0, \ldots, S \) with \( ds \) constant, and so \( dk \) becomes a function of \( ds \) and not vice-versa as in the temporal indexing scheme.

In this case computations are as follows.
First

\[
\mathbf{u}(s) \approx \frac{\begin{bmatrix} \frac{d\bar{x}(s)}{ds}, \frac{d\bar{y}(s)}{ds}, \frac{d\bar{z}(s)}{ds} \end{bmatrix}^\top}{\left(\left(\frac{d\bar{x}(s)}{ds}\right)^2 + \left(\frac{d\bar{y}(s)}{ds}\right)^2 + \left(\frac{d\bar{z}(s)}{ds}\right)^2\right)^{1/2}} = \frac{[d\bar{x}(s), d\bar{y}(s), d\bar{z}(s)]^\top}{[(d\bar{x}(s))^2 + (d\bar{y}(s))^2 + (d\bar{z}(s))^2]^{1/2}},
\]

where the division ensures \(\|\mathbf{u}(s)\| = 1\).

Then

\[
\frac{d\mathbf{u}(s)}{ds} \approx \frac{\begin{bmatrix} \frac{d\mathbf{u}_x(s)}{ds}, \frac{d\mathbf{u}_y(s)}{ds}, \frac{d\mathbf{u}_z(s)}{ds} \end{bmatrix}^\top}{\left\|\frac{d\mathbf{u}(s)}{ds}\right\|}
\]

and

\[
\mathbf{n}(s) \approx \frac{\begin{bmatrix} \frac{d\mathbf{u}_x(s)}{ds}, \frac{d\mathbf{u}_y(s)}{ds}, \frac{d\mathbf{u}_z(s)}{ds} \end{bmatrix}}{\left\|\frac{d\mathbf{u}(s)}{ds}\right\|}
\]

(giving

\[
\mathbf{b}(s) = \frac{\mathbf{u}(s) \times \mathbf{n}(s)}{\|\mathbf{u}(s) \times \mathbf{n}(s)\|}.
\]

Note that derivatives of \(\mathbf{n}(s)\) and \(\mathbf{b}(s)\) can be computed for \(\mathbf{u}(s)\), and application of (3.2.4) and (3.2.8) provide values of \(\kappa(s)\) and \(\tau(s)\), respectively.

The speed \(v\) can be computed by making use of (3.2.10) and it can then be indexed as \(v(s)\); alternatively, since the interval \(ds\) is fixed for any \(s\), the speed can be encoded by using the time intervals \(dk(s)\) variable with \(s\).

We have shown that an action (trajectory shape and dynamics) can be uniquely encoded in a way invariant to rigid motion in the space \((\kappa, \tau, v)\) and simple and complex single sensor actions can be defined as contours in this domain. It is sometimes convenient to encode the trajectory shape separate from dynamics in a 2D \(\kappa-\tau\) space and plot the dynamics, \(v\), separately as shown in Fig. 3.2 and 3.3. However there are a few canonical trajectory shapes worth noting including linear \((\kappa = 0, \tau = 0)\), planar \((\kappa = \text{const}, \tau = 0)\) screw \((\kappa = \text{const}, \tau = \text{const})\) and spiral \((\kappa \neq 0, \tau \neq 0)\) actions. Accordingly, a uniform rectilinear motion is encoded by the point \((0, 0, v_0)\), a circular uniform motion by \((\kappa_0, 0, v_0)\), and a motion with constant
Chapter 3. Human Motion Trajectory Encoder

Figure 3.2: Shape model. Here the locus of points in the 2D $\kappa$-$\tau$ space corresponds to a 3D trajectory positions

Figure 3.3: Dynamic model. Here the speed along the trajectory, $v$, is indexed over time, $t$.

speed along a helix by a point $(\kappa_0, \tau_0, v_0)$. Obviously, in general, curves are mapped to curves in the $\kappa, \tau, v$ space.

3.4 Dealing with Noise

One of the major problems with collecting data from kinematic sensors is noise, particularly when the relevant information involves differential operators of different orders. Precisely because the trajectories correspond to actions the ideal filter is one that retains important amplitudes while removing small variations due to noise. In the following, least-square Gaussian (LS), Savitski-Golay (SG) and optimal Kalman filters (KF) are explored and compared with this in mind.

3.4.1 Least-square Gaussian filter

The least-square Gaussian filter is a non-linear least squares fitting regime minimising the noise in a raw dataset and smoothing the curve by using one or a mixture of Gaussians. The fitting process is updated by minimizing the sum of the squared errors between the raw dataset and the fitting equation. Here, the raw dataset is a set of paired data defined by $\{\bar{Y}_k, \bar{X}_k\}, k = 1, 2, \ldots, T$ as the indexing number[204]. The estimated values of $\bar{Y}_k$ is notated as

$$\bar{Y}_k = \sum_{g=1}^{G} \tilde{a}_g e^{-\left(\frac{x_k - b_g}{s_g}\right)^2},$$ (3.4.1)
where \( k = 1, 2, \ldots, T \) is the indexing number and \( G \) is the order of the filter (the number of Gaussians). \( \tilde{a}_g, \tilde{b}_g \) and \( \tilde{c}_g, g = 1, 2, \ldots, G \) are the parameters chosen to minimise \( \epsilon \) [204]:

\[
\varepsilon = \sum_{k=1}^{T} (\hat{Y}_k - \tilde{Y}_k)^2, \quad k = 1, 2, \ldots, T, \tag{3.4.2}
\]

and

\[
\min_{\tilde{a}_1, \tilde{b}_1, \tilde{c}_1, \ldots, \tilde{a}(G), \tilde{b}(G), \tilde{c}(G)} \varepsilon = \min_{\tilde{a}_1, \tilde{b}_1, \tilde{c}_1, \ldots, \tilde{a}(G), \tilde{b}(G), \tilde{c}(G)} \sum_{k=1}^{T} (\hat{Y}_k - \tilde{Y}_k)^2 \tag{3.4.3}
\]

Here, \( k \) is the indexing number and \( G \) is order of fitting equation.

A solution introduced in [205, 204] is to compute the partial derivatives of the error and set the result to 0:

\[ \nabla \varepsilon = 0, \quad (3.4.4) \]

where, \( \nabla = \left[ \frac{\partial}{\partial \tilde{a}_1}, \frac{\partial}{\partial \tilde{b}_1}, \frac{\partial}{\partial \tilde{c}_1}, \ldots, \frac{\partial}{\partial \tilde{a}(G)}, \frac{\partial}{\partial \tilde{b}(G)}, \frac{\partial}{\partial \tilde{c}(G)} \right]^\top \). This solution can be used in both linear and non-linear least squares methods (see [204] for more details).

However, least squares filters have known disadvantages. Firstly, it is particularly sensitive to outliers, especially the extreme values [204] and, secondly, it is essentially a smoothing filter which does not attempt to preserve steep gradients or important discontinuities so critical in kinematics. To this end higher-order moments need to be included via filters such as the Savitzky-Golay (SG) least squares polynomial filters.

### 3.4.2 The Savitzky-Golay filter

The SG filter is designed to preserve such higher order moments by approximating the underlying function using a best-fitting polynomial moving window. Since the process uses least squares fitting it involves matrix inversion to derive the coefficients of a fitted polynomial that form a convolution kernel defined by their sampling in (window size) and polynomial order [206]. The result is a set of least squares
polynomial filters which can be applied to each of the \(\ddot{x}(t), \ddot{y}(t), \ddot{z}(t)\) recordings of the form:

\[
\begin{align*}
\ddot{X}(k) &= SG(w, \tilde{g}) * \ddot{x}(k) \\
\ddot{Y}(k) &= SG(w, \tilde{g}) * \ddot{y}(k) \\
\ddot{Z}(k) &= SG(w, \tilde{g}) * \ddot{z}(k)
\end{align*}
\] (3.4.5)

where \(*\) denotes convolution, \(w, g\) refer to the window size and order of the polynomial, respectively.

The most important benefit of such polynomial approximations is that higher-order derivatives can be determined algebraically from the derived polynomial coefficients. The norms of velocity, \(v(k)\), acceleration, \(a(k)\) can then be computed using these coefficients for each position parameter, \((\ddot{X}(k), \ddot{Y}(k), \ddot{Z}(k))\), as

\[
\begin{align*}
v(k) &= \sqrt{\left(\frac{d\ddot{X}(k)}{dk}\right)^2 + \left(\frac{d\ddot{Y}(k)}{dk}\right)^2 + \left(\frac{d\ddot{Z}(k)}{dk}\right)^2}, \\a(k) &= \sqrt{\left(\frac{d^2\ddot{X}(k)}{dk^2}\right)^2 + \left(\frac{d^2\ddot{Y}(k)}{dk^2}\right)^2 + \left(\frac{d^2\ddot{Z}(k)}{dk^2}\right)^2},
\end{align*}
\] (3.4.8)

\(v(k)\) and \(a(k)\) provide an invariant (to absolute pose and position) description of the dynamics of a curve, but not its shape. Again, the reason for separating dynamics from shape features is that a known action can have the same shape but different dynamics - even when repeated by the same person.

An experiment was done (refer to Sec. 3.7.1) to compare the performance of SG and LS filter to show why and how SG filter outperforms the other one.

### 3.4.3 Kalman filter

The third type of filter examined was the class of Kalman Filters (KF) varying from the normal linear to extended and unscented versions to accommodate for non-linearities in the data. Although the extended KF linearises inherent non-linearities and the unscented allows for more robust sampling and estimation[207]
it is found that the standard (optimal) KF performed as well as the others so this has been compared with the previous two filters. The normal KF smooths the data by adapting the Kalman gain online in order to minimize the error in fitting a single Gaussian error model. The equations for Kalman filter are listed in Sec. 2.3.1 with

\[
A = \begin{bmatrix}
I_3 & I_3dk & I_3dk^2/2 \\
0_3 & I_3 & I_3dk \\
0_3 & 0_3 & I_3
\end{bmatrix} \in \mathbb{R}^{9 \times 9},
\]

(3.4.10)

\[
B = \begin{bmatrix}
I_3dk^2/2 \\
I_3dk \\
I_3
\end{bmatrix} \in \mathbb{R}^{9 \times 3}
\]

(3.4.11)

and

\[
C = [I_3 \ 0_3 \ 0_3] \in \mathbb{R}^{3 \times 9}.
\]

(3.4.12)

A series of experiments (stated in Sec. 3.7.2 and 3.7.3) was then performed to determine the best combination of indexing and noise filtering schemes for computing shape and kinematics for a given sensor.

### 3.5 Complex Motion Decomposition using Switching Continuous Hidden Markov Models

To syntactically describe complex motions, it is of importance to segment and classify them into smaller atomic units. In this work, switching continuous hidden Markov models [208] will be investigated to perform motion segmentation and classification simultaneously that use atomic motion components to identify complex actions.

#### 3.5.1 Building model for atomic motions

Since a hidden Markov model (HMM) involves two major sets of variables: hidden states and observations, it is essential to determine these variables. For states,
because they are hidden, it is more critical to know the number of states rather than what they are. While for observations, a Gaussian mixture model (GMM) is used, in the continuous HMM case, to model the observations where each observation symbol corresponds to a Gaussian component of the mixture model. The GMM is a weighted sum of $M$ component Gaussian densities defined by

$$
p(\tilde{x}|\lambda) = \sum_{i=1}^{M} \omega_i \tilde{g}(\tilde{x}|\mu_i, \Sigma_i), \tag{3.5.1}
$$

where $\omega_i$, $\mu_i$ and $\Sigma_i$ correspond to the weight, mean and covariance of the $i^{th}$ Gaussian model and $\tilde{x}$ is multidimensional vector. The Expectation-Maximization (EM) algorithm is then used to estimation these Gaussian mixture model GMM parameters $\omega$, $\mu$ and $\Sigma$.

As is typical of HMM estimation, each HMM atomic model uses, again, the EM algorithm with randomly set initial guesses and updated using training data till an optimal (MAP) solutions obtained\[209\], thereby establishing a HMM notated as $\lambda = \{\tilde{A}, \tilde{B}, \tilde{\pi}\}$. Here, $\tilde{A}$ is a state transition probability distribution with $\tilde{A} = \{\tilde{a}_{si,sj}\}, i, j \in [1, \tilde{G}]$ ($s_i$ is the $i^{th}$ state at time $k$, $s_j$ is the $j^{th}$ state at time $k + 1$ and $\tilde{G}$ is the total number of states). $\tilde{B}$ is observation symbol probability distribution with $\tilde{B} = \{\tilde{b}_{oi}\}_{i \in [1, \tilde{G}], l \in [1, \tilde{L}]}$ ($s_i$ is the $i^{th}$ state at time $k$ and $o_l$ is $l^{th}$ observation symbol). $\tilde{\pi}$ is initial state distribution with $\tilde{\pi} = \{\pi_i\}_{i \in [1, \tilde{G}]}$. This version of the EM model is implemented efficiently using the Forward-Backward algorithm\[210\].

### 3.5.2 Complex motion decomposition

In previous section (Sec. 3.5.1), the approach for building a HMM for one atomic motion is introduced. Since a complex motion may contain $\tilde{Q}$ types atomic motions, it is essential to build HMM for each type of atomic motions, the collection of these HMMs is notated as $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_{\tilde{Q}}\}$.

To decompose this complex motion based on its shape, it should be represented
by \{\kappa_k, \tau_k\}, k = 1, 2, \ldots, T. Decomposing the motion is actually solving the following problem using the forward algorithm defined by

$$\arg \max_q p(\kappa_k, \tau_k | \lambda q),$$

(3.5.2)

where \(k = 1, 2, \ldots, T\) and \(q = 1, 2, \ldots, \hat{Q}\). The change of \(q\) means the change of atomic motions, thereby segmenting and classifying the complex motion into pre-defined atomic motions.

### 3.6 Canonical Actions and the Action Alphabet

Though the above derivations describe two components (trajectory shape and dynamics) of complex human motions, approaches for symbolic representation or the classification of these components are not provided, which will be more practical for those who need to monitor, analyse or describe these motions [200]. For example, in situations where it is important to describe how a patient should perform specific rehabilitation exercises or even how a specific limb is not performing normally it is necessary to have a description of the action that fits with current understanding of motor programs and processes underlying normal action productions. In fact, the challenge to construct generative languages for actions has become a focus in recent years in both human kinematics and robotics via the POETICA project[211].

To this end, a “point screw decomposition” has been developed to generate basic “atomic” action descriptions at least for the sensor movements. This method is related to, but not identical with, screw kinematics for rigid bodies as developed over the past two centuries for mechanical motions encoding in terms of the rotations and translations of rigid bodies about a given screw axis (see, for example, McCarthy[212] for a treatment in spherical kinematics terms). It is not aimed at the descriptions of actions at the next level of specific limb movement types and their relations in kinesiology but it is intended to underpin the inference of such actions.
It must first be noted that the curvature and torsion of a helical (screw) motion is constant. For this reason, then, any point, at time $t$, in $\kappa\tau$ space ($\kappa(t), \tau(t)$) can be interpreted as corresponding to a screw action such that if the motion remains at a “fixed point” (does not move) in the $\kappa\tau$ plane then, during this time period, its 3D trajectory corresponds to a constant point screw whose shape is defined by the $\kappa$ and $\tau$ values at that point. Specifically, for a helix defined by:

$$\mathbf{r}(k) = (\bar{m}\cos(k), \bar{m}\sin(k), \bar{n}k), \quad (3.6.1)$$

it is easy to shown that

$$\kappa = \frac{\bar{m}}{m^2 + \bar{n}^2}, \quad \tau = \frac{\bar{n}}{m^2 + \bar{n}^2}. \quad (3.6.2)$$

Similarly the radii of first (curvature-type) and second (torsion-type) curvatures, $\bar{m}, \bar{n}$ are defined by:

$$\bar{m} = \frac{\kappa}{\kappa^2 + \tau^2}, \quad \bar{n} = \frac{\tau}{\kappa^2 + \tau^2}. \quad (3.6.3)$$

These latter relations in (3.6.3) illustrate the relationship between $\kappa, \tau$ values and the types of point screw actions (such as “left-handed” and “right-handed”)[200] which can be used to estimate an atomic motion. Equally, one single point screw action or helix can be utilised to approximate temporally contiguous points that are close in a $\kappa\tau$ space or curve. Contiguity in local shape and time is the basis of our method to encoding such motions. Consequently, in the following section, our aim was[200]:

- to describe atomic motions with a sequence of point screws (specific $\kappa, \tau$ values) “states”,
- to determine how to relate the types of point screws and their dynamics, given the variabilities that occur in the execution of such actions by humans,
• to explore the approaches to compile efficient criteria for the encoding, prediction and recognition of complex single sensor actions via their point screw decompositions.

Again, it must be emphasized that two sets of invariant descriptors: $\kappa \tau$ and speed ($v$) signatures are chosen to encode sensor motions.

Nevertheless, as a result of this formulation a few basic (canonical) motion types are proposed, as shown in Table 3.1. As for direction each such motion can be, for planar: forward/backward; planar: clockwise/anticlockwise, helical: left-handed/right-handed screw actions.

<table>
<thead>
<tr>
<th>Type</th>
<th>Curvature</th>
<th>Torsion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Curved Planar</td>
<td>$&gt; 0$</td>
<td>0</td>
</tr>
<tr>
<td>Non-Planar Screw/Helix</td>
<td>$&gt; 0$</td>
<td>$\neq 0$</td>
</tr>
</tbody>
</table>

### 3.7 Experiments and Results

In this section, four experiments were done to compare filters and indexing schemes, as well as illustrate the application of two-component encoding model. To quantify the performance, standard deviation (STD), correlation coefficient (Corr) and mean square error (MSE) were used in the following subsections. First of all, as shown in Sec. 3.6, curvatures and torsions for linear, planar and helical motion should be constants, standard deviations of the noise (outliers) in curvatures and torsions were employed to show how computed curvatures and torsions related to expected values. Moreover, correlation coefficient and mean square error were used in the first experiment (refer to Sec. 3.7.1) to show the distortion of the filtered trajectories and the raw one. Moreover, in these experiments, all the parameters, such as the
spatial intervals, orders of least-square Gaussian filter, orders and window sizes of Savitzky-Golay filter and so on, were selected by minimising the standard deviations of curvatures and torsions of related trajectories.

In terms of the abbreviations, NF means that no filter was applied (the results were computed based on raw data), while SG, LS and KF are for Savitzky-Golay filter, least-square Gaussian filter and Kalman filter.

### 3.7.1 Comparison between Savitzky-Golay filter and least-squares Gaussian filter

The first experiment was done to illustrate how the SG and LS smooth the trajectory of a helical motion collected from a Microsoft Kinect with sampling rate of 30 frames per second (FPS), which was composed of the 3D positions of right wrist with total 118 samples (indexed by temporal scheme) performed by a healthy subject.

<table>
<thead>
<tr>
<th>Filter</th>
<th>SG</th>
<th>LS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>Window Size</td>
<td>27</td>
<td>-</td>
</tr>
<tr>
<td>STD ((\kappa)) (m(^2))</td>
<td>9.3953</td>
<td>33.9046</td>
</tr>
<tr>
<td>STD ((\tau)) (m(^2))</td>
<td>62.4270</td>
<td>333.0902</td>
</tr>
<tr>
<td>Corr/MSE (m(^2))</td>
<td>0.9995/0.0015</td>
<td>0.8979/0.0153</td>
</tr>
<tr>
<td>TIME (s)</td>
<td>0.0163</td>
<td>1.1324</td>
</tr>
</tbody>
</table>

From the table, it can be seen that SG filter results in the smaller standard deviation of curvature and torsion with less distortion (higher correlation coefficient and lower MSE with respect to the raw trajectory) and shorter computing time. By comparison, the LS filter yields curvature and torsion with standard deviations much higher than SG with lower value of the correlation coefficient and higher MSE than those of SG; moreover, LS has a longer computational time. All in all, when suitable parameters are selected, the SG filter appears to outperform LS in generating curvature and torsion with minimised errors (standard deviations).
Chapter 3. Human Motion Trajectory Encoder

Figure 3.4: Three motions used in experiments of this work. Fig.3.4(a) and 3.4(b) are for linear motion. Fig. 3.4(c) and 3.4(d) are for planar motion. Fig. 3.4(e) and 3.4(f) are for helical motion.

3.7.2 The computability of the two-component model

In this experiment, these two components were computed for a variety of actions including two linear and two planar motions (with 181, 121, 277 and 277 samples indexed by time) of the right wrist by using the Kinect™, sampling 3D positions of joints at 30Hz. Fig. 3.4 shows how these motions were captured - as it was with all the experiments in this chapter.

In Table 3.3 the first column shows the 3D action trajectories while the second Shape column is divided in two corresponding to the $\kappa$, $\tau$ values as a function of time. The third Dynamics column shows the speed $v$ over time. The rows refer to different motions and only trajectories with zero torsion have been considered. Here spatial indexing and SG filtering have been used in the calculations. Here, the third linear and planar motion are simulated perfect motion trajectories without noise.
Table 3.3: Shape and dynamics for four motions. In each left figure, the colour changes from black to grey with time.

<table>
<thead>
<tr>
<th>Trajectory</th>
<th>Shape</th>
<th>Torsion</th>
<th>Dynamics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Linear (1)</td>
<td><img src="image1.png" alt="Diagram" /></td>
<td><img src="image2.png" alt="Diagram" /></td>
<td><img src="image3.png" alt="Diagram" /></td>
</tr>
<tr>
<td>Linear (2)</td>
<td><img src="image4.png" alt="Diagram" /></td>
<td><img src="image5.png" alt="Diagram" /></td>
<td><img src="image6.png" alt="Diagram" /></td>
</tr>
<tr>
<td>Linear (3)</td>
<td><img src="image7.png" alt="Diagram" /></td>
<td><img src="image8.png" alt="Diagram" /></td>
<td><img src="image9.png" alt="Diagram" /></td>
</tr>
<tr>
<td>Planar (1)</td>
<td><img src="image10.png" alt="Diagram" /></td>
<td><img src="image11.png" alt="Diagram" /></td>
<td><img src="image12.png" alt="Diagram" /></td>
</tr>
<tr>
<td>Planar (2)</td>
<td><img src="image13.png" alt="Diagram" /></td>
<td><img src="image14.png" alt="Diagram" /></td>
<td><img src="image15.png" alt="Diagram" /></td>
</tr>
<tr>
<td>Planar (3)</td>
<td><img src="image16.png" alt="Diagram" /></td>
<td><img src="image17.png" alt="Diagram" /></td>
<td><img src="image18.png" alt="Diagram" /></td>
</tr>
</tbody>
</table>

for comparison purpose. It obvious that the values of $\kappa$ and $\tau$ for both simulated linear and planar motions meet that introduced in Table 3.1.

For linear motions, the Shape column reflects the shape of the motion trajectories independent of their dynamics. In these cases curvature and torsion for linear
motion should be 0 since there was no rotation or twist. Graphs of curvature and
torsion as functions of time show this to be the case for most of the time, with
points located away from 0 occurring mainly at the beginning and at the end of the
motion, that is, at points of temporal discontinuity. The shape computation results
were consistent with this with points mostly clustered around the origin of the \( \kappa, \tau \)
plane. In terms of the dynamic computation results, the speed was used, rather
than velocity vectors because, as noted in Sec. 3.3.1, they can be recovered from
\( \kappa, \tau \), i.e. the shape component of our model. Graphs of speed versus time show two
different kind of motions. In the first upward motion an initial increase of speed
was followed by an interval in which \( v \) varied a little around a constant value and
decreased when the motion stopped. In the second downward motion \( v \) increased
linearly, uniformly accelerating up to the end and then rapidly decreasing to zero.
Thus, although the motions have different locations and direction they had very
similar shapes (in the sense of \( \kappa, \tau \) values) while exhibiting different dynamics. The
fourth and fifth rows in Table 3.3 show results for two planar motions forming closed
curves with different directions (tangent vectors) and locations. In both cases the
curvature is relatively small for large portions of the paths while the torsion should
be zero since there was no twist in these motions.

The results were effected by the fact that curvature and torsion are very sensitive
to noise, so that values \( \kappa \) show a spread in the range from 0 to 200 \( m^{-1} \). Although
values of \( \tau \) are more noise than that of \( \kappa \), they fluctuate around 0 \( m^{-1} \), which
generally follows the expectation shown in Table 3.1.

However most of the significant outliers occurred at the beginning and end of the
motion or in points where sharp cusps occurred in the trajectory as, for instance,
where there were two peaks in the bottom part of the left planar motion 1 or in the
left hand side of motion 2.

Finally, computed \( v \) values showed very similar dynamics (independently of their
Chapter 3. Human Motion Trajectory Encoder

shape); here, not surprisingly, $v$ values were more scattered than in the linear case due to the inaccuracy of the Kinect and the difficulty of the subject accurately outlining the specified action. In spite of this, this first experiment demonstrates that the system can encode the shape and dynamics of an action reliably albeit somewhat noisy at the beginning or end of an action as expected from the nature of the filtering, computations involved.

In addition, a simulation was implemented to illustrate the independence between shape model and dynamic model. In this part, two helical trajectories with same shape but different dynamic were simulated with Matlab®. For the first trajectory, the speed was constant, while in the second trajectory, the speed varied. In the first, second, third and fourth quarter of the second trajectory, the speed was 10, 30, 5 and 40 times as much as that in the first trajectory respectively. These two trajectories were shown as Fig. 3.5.

The two trajectories are the same and then they share the same curvature and torsion (shape model) as shown in Fig. 3.6. However, the dynamics of these two trajectories were different, as could be proved by investigating their dynamic

![Figure 3.5](image-url)

Figure 3.5: Two helical trajectories with the same orientation and shape, but two different dynamics. The colour of the trajectory in Fig. 3.5(a) with constant speed (roughly 0.1 m/s) changes from black to grey with time, and the different line styles and colours of the trajectory in Fig. 3.5(b) indicates the change of velocities. The legend on the right side of Fig. 3.5(b) shows the approximate speed of different parts of the trajectory in Fig. 3.5(b).
models, as shown in Fig. 3.7.

From Fig. 3.7(a), it can be seen that the average speed of trajectory is around 0.1 m/s. In Fig. 3.7(b), the speed for the first quarter (from 1 s to 50 s) is approximately 1 m/s, which is 10 times as much as that of the average of first trajectory. Similarly, it is clear that in second (50 s to 100 s), third (100 s to 150 s) and fourth (150 s to 200 s) quarters, the speeds are 3 m/s, 0.5 m/s, and 4 m/s separately. This result is corresponding to the expectation in the beginning of the simulation.
Table 3.4: Parameters used to generate the standard deviation for filters processing trajectories with temporal (first four rows) and spatial (last four rows) indexing scheme. Here, L and P are for the linear and planar motion respectively. (the same in the rest tables).

<table>
<thead>
<tr>
<th>Interval</th>
<th>Order</th>
<th>Window Size</th>
<th>Order</th>
<th>Iterate Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temporal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>-</td>
<td>6</td>
<td>121</td>
<td>101</td>
</tr>
<tr>
<td>P</td>
<td>-</td>
<td>5</td>
<td>181</td>
<td>576</td>
</tr>
<tr>
<td>Spatial</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>L</td>
<td>0.01m</td>
<td>5</td>
<td>45</td>
<td>118</td>
</tr>
<tr>
<td>P</td>
<td>0.05m</td>
<td>5</td>
<td>59</td>
<td>566</td>
</tr>
</tbody>
</table>

3.7.3 Comparisons between the combinations of various indexing schemes and filtering approaches

In this experiment, the performance of the indexing and filtering methods for these linear and planar actions mentioned in the previous experiment have been more carefully examined, again, with the purpose of determining the best way to minimise noise. The indexing schemes (temporal and spatial) were used in conjunction with three filtering methods (LS, SG, KF) where, in each case, optimal parameters were used for comparison purposes. At the same time, sets of data generated from trajectories without applying any filter (NF) was used for comparison. Here, one linear and one planar motion are selected for result demonstration.

Table 3.4 summaries the parameters used by filters to process trajectories with temporal and spatial indexing scheme respectively. As can be seen, the orders and window sizes for Savitzky-Golay smoothing filter changes from time for two motions indexed with temporal indexing scheme, while keeping almost the same when spatial indexing scheme was applied. Moreover, the reason order 8 was used for LS filter is that although the lower orders gave a smoother trajectory, it is distorted, like that in Table 3.2. In terms of the KF the optimal iteration number varied considerably.

Table 3.5 illustrates the standard deviations of curvatures and torsions of two
Table 3.5: The comparison of curvatures and torsions for four trajectories indexed by temporal and spatial indexing schemes and filtered by SG, LS and KF. Here C is for curvature and T is for torsion.

<table>
<thead>
<tr>
<th></th>
<th>NF</th>
<th>SG</th>
<th>LS</th>
<th>KF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temporal</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>L</td>
<td>71.56</td>
<td>25.12</td>
<td>101.79</td>
</tr>
<tr>
<td>P</td>
<td>32.49</td>
<td>4.98</td>
<td>6.12</td>
<td>7.43</td>
</tr>
<tr>
<td>T</td>
<td>L</td>
<td>1.16E+7</td>
<td>7.75</td>
<td>42.77</td>
</tr>
<tr>
<td>P</td>
<td>70621.72</td>
<td>2.16</td>
<td>40.76</td>
<td>2.22</td>
</tr>
<tr>
<td>Spatial</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>L</td>
<td>4.06</td>
<td>2.64</td>
<td>3.84</td>
</tr>
<tr>
<td>P</td>
<td>2.88</td>
<td>0.87</td>
<td>1.14</td>
<td>2.74</td>
</tr>
<tr>
<td>T</td>
<td>L</td>
<td>7.49</td>
<td>1.85</td>
<td>5.41</td>
</tr>
<tr>
<td>P</td>
<td>3.23</td>
<td>0.27</td>
<td>1.14</td>
<td>2.74</td>
</tr>
</tbody>
</table>

Trajectories, namely linear (L) and planar (P), indexed with temporal and spatial indexing scheme and denoised with three filters. By comparing the performance between temporal and spatial indexing schemes (top four rows and bottom four rows), it can be seen that the spatial indexing scheme outperforms the temporal one by generating curvatures and torsions with smaller standard deviations. Moreover, SG outperforms LS and KF since the trajectory in each row filtered by SG has the smallest standard deviations of curvatures and torsions. It is also noteworthy that some numbers, such as the torsion for linear motion without applying filters, are extremely large, which is caused by some outliers in these trajectories.

3.7.4 Motion decomposition using switching continuous hidden Markov model (SCHMM)

Based on the conclusion of Experiment 1, it is clear that our two-component model can be used for motion decomposition. With the help of the result of the second experiment, the noise in raw data collected by a typically noisy sensor like the Kinect® can be significantly reduced. So, in the third experiment how the two-component model can be used to segment action data into such components using a standard Hidden Markov Model (HMM) formulation [208] was explored. In this experiment, a continuous Gaussian hidden Markov model (CHMM)[208] was
implemented in Matlab® based on Bayes Net Toolbox [214] where three HMMs were constructed for linear, planar and helical motions respectively and the segmentation into these three motion classes was determined by MAP criteria, that is, at any given time, the HMM was selected which produced the highest Viterbi score [208]. For testing, four complex motions being various combinations of linear, planar and helical motions were analysed - all based purely on their curvature and torsion values and having different dynamics.

Three linear motions (89, 94, 87 samples indexed by spatial indexing scheme with spatial interval of 0.015 meter), three planar motions (113, 122, 116 samples indexed by spatial indexing scheme with spatial interval of 0.04 meter) and two helical motions (142, 147 samples indexed by spatial indexing scheme with spatial interval of 0.04 meter) were selected as training data sets to construct three HMMs for linear, planar and helical motions respectively. Already these can define quite complex and different actions depending on how the motion components are sequenced. Six Gaussian mixtures, clusters, were used as six observation symbols in each HMM along with six (latent) states. These parameters resulted in excellent recognition rates compared with other explored values.

After building HMMs for trajectories of various atomic motions, four complex motions combining these atomic motions were used to illustrate the performance of trajectory decomposition with two-component encoding model, spatial indexing scheme and SG filter. Before decomposing these complex motions automatically with the proposed approach, they were manually segmented into different atomic motions and counted the samples in each segment as reference for computing recognition rate. The trajectory of these four complex motions and the examples of decomposition graphs are shown as follow.

Table 3.6 illustrates the performance of the SCHMM with shape models as features. The first complex motion consisted of linear motion and a planar motion
Table 3.6: Trajectories and decomposition of complex motions. In each top figure, the colour changes from black to grey with time in the trajectories.

<table>
<thead>
<tr>
<th>Motion</th>
<th>Trajectory</th>
<th>Decomposition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Motion 1</td>
<td><img src="image" alt="Trajectory 1" /></td>
<td><img src="image" alt="Decomposition 1" /></td>
</tr>
<tr>
<td>Motion 2</td>
<td><img src="image" alt="Trajectory 2" /></td>
<td><img src="image" alt="Decomposition 2" /></td>
</tr>
<tr>
<td>Motion 3</td>
<td><img src="image" alt="Trajectory 3" /></td>
<td><img src="image" alt="Decomposition 3" /></td>
</tr>
<tr>
<td>Motion 4</td>
<td><img src="image" alt="Trajectory 4" /></td>
<td><img src="image" alt="Decomposition 4" /></td>
</tr>
</tbody>
</table>

and is indexed by spatial indexing scheme with spatial interval of 0.007 meter. From the classification/segmentation results, it is easy to see that the first 0.14 meter of this motion is linear while the remainder corresponds to a planar motion. For motion 2, the first 0.154 meter were classified as planar, followed by planar motion of around 0.161 meter, linear motion of about 0.35 meter and planar motion of approximately 0.14 meter. In this diagram, the change from linear to planar motion after about 0.8 meter is caused by the small planar motion included in the linear motion. After all, it is very hard for a human being to do an exact linear motion.
Moreover, the third complex motion is compounded with one helical motion and a linear motion. This sequence can be told from the classification diagram easily, while the unexpected change in the last part is caused by the same reason as the previous motion. The last motion was combined by a helical, a planar and a linear motion, which could be reflected from the classification graph.

At the end of the experiment, 70 different complex motions combining linear, planar and helical motions conducted by ten healthy people in lab environment were tested to gather the classification rate. Each person not only performed four motions listed in Table 3.6, but also did three more trajectory composing linear, planar and helical atomic motions while different from pre-defined four motions. The average classification rate for linear motions was around 89.2%, for planar motions, it was roughly 90.6%, while helical motions was approximately 92.2%. Two major factors may lead to this result. First of all, the noise coming from testing subjects and Kinect®, which is unlikely to be eliminated completely by filters, may negatively influence the classification rate and, next, it is very hard for a person to make an exact linear, planar or helical motion. Therefore, it is reasonable to expect that some parts of linear motion to be classified as a planar, and analogously, parts of planar trajectory to be classed as helical, especially when one motion changes to another, as well as at the beginning and end of complex motions, such as cases 2, 3 and 4 in Table 3.6.

3.8 Summary

Similar to learning a language, it is critical to understand the basic attributes of interest in motions in order to study the motions and evaluate the kinematic performance in a physical tele-rehabilitation system. In this study, a novel approach was investigated to represent the human motions in joint (sensor) level regardless
of the orientation of the motions by extracting the atomic motions with the two-component (shape and dynamics) encoding model. By using the spatial indexing scheme and SG filter, the model can be built in a robust fashion. A number of simulations and real-data experiments have been conducted to illustrate the advantages of using spatial indexing scheme and SG filter. More importantly, the proposed model can be implemented to quite accurately extract predefined atomic sensor motions.

However, using this two-component encoder has some limitations. From the experiments performed in this study, it can be seen that the proposed two-component model is sensitive to noise. More specifically, outliers from motion capture devices, tremors, or pathological motion changes in trajectories may result in extreme curvatures and torsions. That is the reason why filters are applied. However, since the scope of this study is to introduce a new encoding model for human complex motion trajectory decomposition, rather than evaluating motor performance, the preliminary experiments were conducted with motion trajectories captured from healthy people.

Therefore, the future works relating to this study are in three main aspects. For the first aspect, this two-component encoder should be applied on data collected from patients whose motion trajectories may involves involuntary movements with various extents. There may be two potential approaches to deal with tremors and pathological motion changes. One is that they can be treated as noise and filtered with various filters. The other method is to build HMM for atomic motions of patients and use of these HMMs, instead of using atomic motion HMMs trained with trajectories of healthy people, to decompose more complex motions of patients. Secondly, to overcome the sensitivity characteristics of this two-component model, either a more suitable filter or algorithm should be explored to eliminate the noise in captured raw data or more accurately estimate elements in differential geometry,
including tangent, normal and binormal, thereby estimating the shape model more accurately. Thirdly, as it is mentioned in this chapter, this two-component encoder merely represent sensor (joint) level trajectories. As a future work, the limb level and action level should be explored to build a three-level encoding model to build a linguistic framework for human actions for tele-rehabilitation.
Chapter 4

Kinematic Performance Evaluation

4.1 Introduction

In recent decades, with the advancements in tele-rehabilitation and associated motion capture technologies, an increasing number of research and development activities are focusing on the development of automated quantitative measures of patient performance in Activities of Daily Living (ADL)[126, 127, 215]. Due to the important role played by the upper extremity in ADL [216, 128], an automated approach for measuring and assessing the ability of upper extremity to perform certain tasks is vital for tele-rehabilitation systems to deliver their full potential.

Although questionnaire-based tools have been utilised pervasively by clinicians, they are not suitable for the tele-rehabilitation environment. One reason is that the use of the majority of these tools requires clinicians, who are usually absent in the tele-rehabilitation sessions. Furthermore, some self-report questionnaires may lead to biased results. Therefore, in tele-rehabilitation, it is critical to develop an automated approach to objectively assess the ability of patients in order to assist therapists to make further clinical decisions.

In this chapter, a preliminary investigation of the feasibility of utilising an automatic approach to assess the ability of patients suffering from dyskinesia to perform an upper extremity reaching task in their daily living is introduced. This is assessed
by measuring the smoothness of motion trajectories and the duration to finish the task. As is pointed out by Daneault et al. [217], dyskinesia is one of the factors that adversely influence voluntary movement since some involuntary movements would be performed. Therefore, they proposed that dyskinesia in Parkinson’s disease could be seen as a factor in the signal-to-noise ratio (SNR) equation with voluntary movements as the numerator (motion input) and dyskinesia as one element in the denominator. Furthermore, to assess the severity of dyskinesia, the amplitude of involuntary movements is one of the important elements that has been utilised in Abnormal Involuntary Movement Scale (AIMS)[141, 217]. In addition, dyskinesia may be associated with some degree of jerk in the extremities [218, 219]. Due to inaccurate motion trajectories, patients with dyskinesia are more likely to reduce their speed and take a longer time to finish a task in comparison to healthy subjects [220]. Therefore, it is reasonable to infer that by looking at the sub-movements and jerks (smoothness) in motion trajectories, as well as the motion duration, the ability of the subject to perform reaching tasks in daily life can be evaluated from an action kinematic standpoint. In this work, sub-movements and jerks are used to infer involuntary movements with large and small amplitudes, which are defined in Table 4.1 for the experiments.

In line with our work, a number of studies have been conducted to evaluate automated performance measurements or kinematics relating to upper extremities [221](some commonly utilised features are given in Fig. 4.1). One of the most obvious critical factors negatively impacting on the quality of upper extremity movements is the smoothness of motion trajectories. Zariffa et al. [222] considered directional changes, mean velocity, ratio of mean and maximum velocity and mean jerk to measure the smoothness of trajectories in patients’ upper limb movements collected from a robotic rehabilitation device. In addition, Rohrer et al. [223] compared five features, namely jerk, speed, mean arrest period ratio, number of peaks
in speed and “tent” metric, to evaluate the smoothness of an arm motion trajectory performed by stroke patients. Moreover, Lum et al. [224] counted the number of times the tangential acceleration of hand passed zero to measure the smoothness of upper extremity movements in stroke patients. Apart from smoothness in motion trajectories, the duration of a specific task is also important for upper extremity performance evaluation. Murphy et al. [225] took duration of drinking into account to analyse the kinematic aspect of drinking with a cohort of healthy subjects. Similarly, duration was also considered as a factor of upper extremity movement assessment in [226, 227]. In addition, Balasubramanian et al. [226] highlighted the disadvantages in some existing approaches and proposed utilising the spectral arc-length metric of the movement speed profile’s Fourier magnitude spectrum to evaluate the smoothness of the movements.

The contributions of this chapter are three-fold:

- Using the shape of the trajectory and instantaneous acceleration to extract kinematic smoothness via the concept of motion entropy;
- Utilising smoothness and duration as criteria to evaluate the performance of an upper extremity reaching task;
- Investigate the possibility of using an affordable, non-invasive consumer device for evaluation of an upper extremity reaching task performance evaluation on a regular basis.

In Sec. 4.2, the methodology used in this chapter is discussed, including the analysis and feature extraction approaches, followed by the experiment setup (Sec. 4.3). After that, the results of computer simulation and the real-data experiment are shown in Sec. 4.4. Concluding remarks are given in Sec. 4.5.
Figure 4.1: Examples of commonly used techniques with features considered for ADL performance measurement. Feature-based performance evaluation (FPE) is primarily based on kinematic or kinetic measurements. Dynamic measurements such as number of velocity peaks [228], number of zero-crossing tangential accelerations [229], dimensionless jerk metrics [223] and spectral arc-length [226] are used in techniques such as VP, ZCA, DJ and SAL respectively in the literature. Our proposed entropy of shape and instantaneous acceleration (ESA) introduced in this chapter uses both shape and kinematic based measurements. Kinetic-related features, such as moment [230], force [231] and torque [232] have also been investigated.

4.2 Methodology

4.2.1 Severity levels definition

In order to perform computer simulations as well as to obtain data from healthy subjects mimicking the underlying involuntary movements, it is important to precisely specify the severity levels of involuntary movements in a kinematic standpoint. Since the frequency, amplitude of involuntary movements in addition to the duration of the specific task, are important factors in assessments [141, 233, 219], three severity levels of involuntary movements were defined to assess the kinematic performances of the upper extremity in a reaching task, which have been listed in Table 4.1. As this work is a preliminary study to investigate the feasibility of using kinematic measurements to evaluate the severity of involuntary movements, indeed
a more focused exercise to describe each level and a larger set of levels can be used [141]). Nevertheless, our proposed approach can simply be used with an improved distinction of severity levels and hence consider this aspect if not the primary focus of this work.

Table 4.1: The definition of three kinematic severity levels of involuntary movements and jerks, as well as their corresponding abilities in performing reaching tasks in daily living.

<table>
<thead>
<tr>
<th>Level</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of sub-movements</td>
<td>0</td>
<td>&lt;= 3</td>
<td>&gt;3</td>
</tr>
<tr>
<td>Amplitude of sub-movements</td>
<td>0</td>
<td>&lt;= 0.3 m</td>
<td>&gt;0.3 m</td>
</tr>
<tr>
<td>Number of jerks</td>
<td>0</td>
<td>&lt;= 3</td>
<td>&gt;3</td>
</tr>
<tr>
<td>Amplitude of jerks</td>
<td>0</td>
<td>&lt;= 0.03 m</td>
<td>&gt;0.03 m</td>
</tr>
<tr>
<td>Duration</td>
<td>&lt;= 5 s</td>
<td>5s ~ 10 s</td>
<td>&gt;10 s</td>
</tr>
<tr>
<td>Ability in performing reaching task</td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
</tr>
</tbody>
</table>

### 4.2.2 Feature extraction

In order to quantitatively evaluate the ability to perform the experimental task, various features need to be extracted from a raw 3D trajectory $\gamma(k) = [\bar{x}(k), \bar{y}(k), \bar{z}(k)]^\top$, where $\bar{x}(k)$, $\bar{y}(k)$ and $\bar{z}(k)$ are the joint position in a Cartesian coordinate frame at time $k = 1, 2, \cdots, T$ with a temporal interval of $dk$ captured from the Kinect ($dk = 30ms$). In this work, the concept of approximate entropy of motion trajectory and the duration associated with the motion were considered. The approximate entropy of the trajectory is related to the shape and the dynamics (i.e. instantaneous acceleration) computed from the trajectory. The feature extraction process is depicted as follows:

- **Shape Model**

Apart from the dynamics in a trajectory, its shape is also taken into account since the involuntary movements are usually associated with randomly moving joint positions, which are represented as unknown uncertainties in the shape.
Therefore, it is important to extract the shape of the trajectory for evaluation [234].

Curvature ($\kappa$) and torsion ($\tau$) of a trajectory can be computed as

\[ \kappa(k) = \frac{||v(k) \times a(k)||}{||v(k)||^3}; \]  \hspace{1cm} (4.2.1)

\[ \tau(k) = \frac{(v(k) \cdot a(k)) \times j(k)}{||v(k) \times a(k)||^2}, \]  \hspace{1cm} (4.2.2)

where $v(k)$ is velocity, $a(k)$ is acceleration and $j(k)$ is jerk.

Since the normal approach to compute numerical differentiation is very sensitive to noise, the approach introduced in [235] is utilised to estimate $v(k)$, $a(k)$ and $j(k)$.

- Instantaneous Acceleration

The instantaneous acceleration is the magnitude of the acceleration throughout the trajectory computed as

\[ \dot{a}^i(k) = ||a(k)||, \]  \hspace{1cm} (4.2.3)

where $k = 1, 2, \cdots, T$ and $a(k)$ was estimated in the previous step.

- Approximate Entropy

Since the uncertainty in the features, including acceleration and shape model, is implicitly captured to determine how smoothly or not a person performs a task, approximate entropy [236] can be computed based on the previous features.

To compute the approximate entropy of a variable, i.e. instantaneous acceleration, it is denoted in discrete form as

\[ \bar{a} = [\dot{a}_1, \dot{a}_2, \cdots, \dot{a}_T] \]  \hspace{1cm} (4.2.4)
for a trajectory with the length $T$. By defining a constant $\hat{m} \in \mathbb{N}^+$ for the length of the captured sequence of data, the vector format is given as,

$$\bar{b}_i = [\hat{a}_1, \hat{a}_2, \cdots, \hat{a}_{i+\hat{m}-1}], \quad (4.2.5)$$

where $i = 1, 2, \cdots, T - \hat{m} + 1$.

A given constant $r > 0$ indicates the filtering level and for each $i$, $\bar{c}_i^\hat{m}(r) \in \mathbb{R}$ is calculated by finding the number of $\bar{b}_j$s that satisfies the condition $d[\bar{b}_i, \bar{b}_j] \leq r$ and dividing it by $T - \hat{m} + 1$, where $j = 1, 2, \cdots, T - \hat{m} + 1$ and $d[\bar{b}_i, \bar{b}_j] = \max_{p=1,2,\cdots,\hat{m}}(|\hat{a}_{i+p-1} - \hat{a}_{j+p-1}|)$. Then the approximate entropy of the variable in (4.2.4) can be computed as

$$H_{\bar{a}} = \lim_{r \to 0} \lim_{\hat{m} \to +\infty} \lim_{T \to +\infty} [\bar{\Phi}_{\hat{m}}(r) - \bar{\Phi}_{\hat{m}+1}(r)], \quad (4.2.6)$$

where

$$\bar{\Phi}_i^\hat{m}(r) = (T - \hat{m} + 1)^{-1} \sum_{i=1}^{T - \hat{m} + 1} (\log_2 \bar{c}_i^\hat{m}(r)). \quad (4.2.7)$$

However, for a 3D trajectory shape, there are two variables of significance; curvature ($\kappa$) and torsion ($\tau$). Here the approximate entropy of the trajectory is computed, considering joint approximate entropy of $\kappa$ and $\tau$ as $H_{\kappa,\tau}$.

However, the following remark is vital for computational simplicity.

**Remark:** Curvature and torsion are independent variables ($\kappa \perp \perp \tau$).

According to the Frenet-Serret formulas \[237, 238\]

$$\begin{bmatrix} du(k) \\ dn(k) \\ db(k) \end{bmatrix} = \begin{bmatrix} 0 & \kappa(k) & 0 \\ \kappa(k) & 0 & \tau(k) \\ 0 & -\tau(k) & 0 \end{bmatrix} \begin{bmatrix} u(k) \\ n(k) \\ b(k) \end{bmatrix} \quad (4.2.8)$$

$\kappa(k)$ and $\tau(k)$ describes the relationship between $u(k)$, $n(k)$ and $b(k)$, where $u(k) \perp n(k) \perp b(k)$. From

$$\frac{du(k)}{dk} = \kappa(k)n(k), \quad (4.2.9)$$
It can be seen that $\kappa(k)$ is the amplitude of the projection of the change of tangent vector on the normal vector. Similarly,

$$\frac{db(k)}{dk} = -\tau(k)n(k), \quad (4.2.10)$$

shows that $\tau$ is the amplitude of the projection of the change of binormal vector on the normal vector. Since $\kappa$ and $\tau$ indicate the change in two independent vectors, they are independent of each other. Therefore, $H_{\kappa,\tau} = H_{\kappa} + H_{\tau}$.

4.3 Experiment Setup

4.3.1 Simulation data collection

The simulations were conducted with Matlab 2013a® to ensure that the proposed approach for smoothness measurement met the consistency, sensitivity and robustness requirements given in [226].

To simulate the reaching movement, the following process was used.

1. Voluntary movement

A noiseless, free reaching involuntary movement is simulated as a smooth arc with a duration $T$

$$\gamma(k) = \{\cos(\pi k/(3 \times T) + \pi/3), \sin(\pi k/(3 \times T) + \pi/3), 0\}, \quad (4.3.1)$$

where $k = 1, 2, \cdots, T$.

2. Sub-movements

The sub-movements are generated by the sum of multiple Gaussian models for each axis (X, Y and Z) as,

$$\tilde{I}(k) = \sum_{q_i=1}^{Q_i} F_{q_i} \exp \left(-\frac{(k - \mu_{q_i})^2}{2\sigma_{q_i}^2}\right), \quad (4.3.2)$$
where $\bar{Q}_i$ is the number of involuntary movements, $F_{q_i}$, $\mu_{q_i}$ and $\sigma_{q_i}$ are the amplitude, mean time and standard deviation of the duration of the $q_i^{th}$ involuntary movement. These variables can be different for various axes. By adding the $\tilde{I}(k)^x$, $\tilde{I}(k)^y$ and $\tilde{I}(k)^z$ to $\gamma_k$, a trajectory with involuntary movements can be created.

3. Jerk

Jucks are simulated by adding Gaussian noise with normal distribution and various amplitudes into the motion trajectories. There are four parameters to determine the jerk, including the number of jerks ($\bar{Z}_j$), starting time ($S_{\bar{Z}_j}$), duration ($D_{\bar{Z}_j}$) and amplitude of jerks ($F_{\bar{Z}_j}$).

To ascertain the consistency and sensitivity of the proposed approach, 50 trajectories were generated to simulate a reaching movement with various numbers and amplitudes of involuntary movements and jerks. The specifications for these trajectories are shown in Table 4.2.

To simplify the simulation without losing the effects, only the trajectories in levels two and three were generated. Furthermore, it is assumed that the numbers and amplitudes of sub-movements in three axes of the trajectories are the same, which means $\tilde{I}(k)^x = \tilde{I}(k)^y = \tilde{I}(k)^z$. It is the same for jerks.

In addition, $\frac{1}{SNR}$ is utilised as another method to illustrate the smoothness of trajectories. $SNR$ is the signal-to-noise ratio and is non-linear with respect to the linear change of smoothness, while $\frac{1}{SNR}$ is almost linear (refer to Fig. 4.2). Therefore, $\frac{1}{SNR}$ is used so that the value is directly proportional to the noise level in trajectories, as well as the severity levels. Fig. 4.2 shows the $\frac{1}{SNR}$ values of the generated 50 trajectories. Since the first 25 trajectories belongs to the second severity level and the last 25 trajectories are in severity level 3, the first 25 $\frac{1}{SNR}$ values (blue line) are smaller than the last 25 $\frac{1}{SNR}$ values (red line). These trajectories are used to simulate patients’ motion trajectories with various degrees of smoothness when
Table 4.2: Parameters used to simulate two groups of trajectories. These two groups of trajectories correspond to two severity levels of involuntary movements. The first 25 trajectories belong to the second level with two involuntary movements and two jerks. The last 25 trajectories are in the third level with four involuntary movements and jerks. To simulate the severity in one level, the amplitudes of involuntary movements and jerks increase with the index.

<table>
<thead>
<tr>
<th></th>
<th>Level 2</th>
<th>Level 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index</td>
<td>1, 2, ⋯, 25</td>
<td>26, 27, ⋯, 50</td>
</tr>
<tr>
<td>Duration</td>
<td>4.5 seconds (135 frames)</td>
<td>7.5 seconds (210 frames)</td>
</tr>
<tr>
<td>( \bar{Q}_i )</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>( \mu_{q_i}/\sigma_{q_i} )</td>
<td>45/50</td>
<td>40/50</td>
</tr>
<tr>
<td></td>
<td>90/50</td>
<td>90/60</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>140/60</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>190/30</td>
</tr>
<tr>
<td>( F_{q_i}(m) )</td>
<td>index ( \times 0.009 )</td>
<td>index ( \times 0.009 )</td>
</tr>
<tr>
<td></td>
<td>index ( \times 0.01 )</td>
<td>index ( \times 0.01 )</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>index ( \times 0.011 )</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>index ( \times 0.012 )</td>
</tr>
<tr>
<td>( Z_j )</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>( F_{z_j}(m) )</td>
<td>index ( \times 0.001 )</td>
<td>index ( \times 0.001 )</td>
</tr>
<tr>
<td></td>
<td>index ( \times 0.0011 )</td>
<td>index ( \times 0.0011 )</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>index ( \times 0.0012 )</td>
</tr>
<tr>
<td></td>
<td>-</td>
<td>index ( \times 0.0013 )</td>
</tr>
</tbody>
</table>

they are doing tele-rehabilitation exercises and activities of daily living. Based on these trajectories, the advantages of the propose approach, in terms of consistency and sensitivity, can be very well evaluated.

In addition, to evaluate the robustness of different approaches, 100 noiseless trajectories in level two were generated and the numbers and amplitudes of involuntary movements were randomly generated in the given range (refer to Table 4.2). For each trajectory, 100 noisy trajectories were generated by adding Gaussian noise with zero mean and 0.33\( v_{peak} \) standard deviation following the experiment in [226].

### 4.3.2 Real-data experiment data collection

For real-data experiment, no film recordings of subjects were made in this study. The Kinect camera provided numerical data that directly related to arm movements.
Only re-identified numerical data, representing motion vectors, were stored in the database. Volunteers were researchers and students at Deakin University. Ethics for conducting the experiments in this study has been approved by Deakin University.

The real data experiment was conducted with four healthy subjects mimicking three severity levels of involuntary movements (refer to Table 4.1) while performing an upper limb task, i.e. moving a book from one location to another and bringing it back to the original location. Before recording the data, subjects were required to practice the tasks to make sure that their movements for different levels involved the required involuntary movements and durations.

In the experiment, a table, a chair, a book, a second version Kinect and a laptop were used (refer to Fig. 4.3 and Fig. 4.4). The chair had no arms and its height was adjustable to suit the subjects. During the experiment, the subject was about 20 cm away from the front of the table so that the book could easily be reached with pure arm movements (without moving his/her trunk). The dimensions of the book were $23.5 \times 15.5 \times 1$ cm and it weighed 0.25 kg. At the same time, the book was placed near the edge of the table so that the subject could hold the book easily.
To accurately track the involuntary movements and jerk, a small infra-red reflective marker was attached on the tracking joint (wrist) and made sure the marker always faced the Kinect so that the Kinect could capture the position of the wrist. The data collection program was written with Kinect SDK v2.0-1409 with C# under Windows® 8.1. Although there was no precision evaluation on the second version of Kinect, according to [239], the precision of the new version of Kinect was close to its predecessor (less than 5 mm for the distance between the Kinect and the object less than 1.5 meter [240]).

Four healthy subjects were recruited for the experiment. Their demographic data can be seen in Table 4.3. They were required to perform the task of taking a book from one pre-defined location to another location and then returning it to the original location. The original and target locations were marked on the table in advance and the subjects were required to place the book at the designated position accurately. Further, the subjects were required to mimic three different severities of involuntary movements with various durations (10 iterations for each subject). These involuntary movements were mimicking patients experiencing dyskinesia so that the motion trajectories involved jerkiness and uncontrollable sub-movements. The numbers and amplitudes of involuntary movements generally followed the specification in Table 4.1. In addition, music was played with three different durations.
Figure 4.4: Real data experiment setup image. The top image shows the setup of the Kinect and the subject. The bottom left and right are the RGB and depth images taken from the Kinect. Note that the marker was on the right wrist of the subject (the depth and RGB camera in the Kinect reversed the image.)

(4, 7 and 15 seconds for levels one to three) so that the motion duration of the subjects could be generally controlled in three levels. Eventually, a deterioration was expected in smoothness from the first to the third level and increased duration. Therefore, the ability to perform reaching tasks decreased. These criteria would be used by a human observer to classify the ability to perform the reaching task into
three levels manually for the purpose of validation.

Table 4.3: Demographic data of subjects

<table>
<thead>
<tr>
<th>Subject</th>
<th>Age</th>
<th>Weight (kg)</th>
<th>Height (cm)</th>
<th>Gender</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>28</td>
<td>55</td>
<td>172</td>
<td>Male</td>
</tr>
<tr>
<td>2</td>
<td>29</td>
<td>70</td>
<td>175</td>
<td>Male</td>
</tr>
<tr>
<td>3</td>
<td>27</td>
<td>60</td>
<td>173</td>
<td>Male</td>
</tr>
<tr>
<td>4</td>
<td>22</td>
<td>58</td>
<td>160</td>
<td>Female</td>
</tr>
</tbody>
</table>

During the experiment, the subject initially held the book with his/her dominant hand (right hand for all subjects) and kept it steady. At the same time, the system operator checked whether the Kinect could capture the marker. If the Kinect could capture the marker and the subject was ready, the system operator gave the subject an instruction to start moving the book and played the music. In the meantime, the Kinect system started recording the position information of the marker into a database for offline analysis. As soon as the subject finished the task (replacing the book in the original position), the system operator stopped the system. Apart from the system operator and the subject, another researcher classified the task (one of the three levels). The manual classification criteria include the duration of finishing the task and numbers and amplitudes of involuntary movements listed in Table 4.1.

Each subject was required to perform the task at least 30 times in total to ensure that there were at least 10 trials at each level. All the 30 trials were conducted over three days with 10 to 15 trials per day depending on the subject availability. Between each trial, the subject could rest for thirty seconds.

4.4 Data Analysis and Results

4.4.1 Computer simulation data analysis

Three simulations were conducted to assess the performance of the proposed approach in terms of motion smoothness, which was evaluated with five approaches,
namely, the number of tangential velocity peaks (VP) \([228]\), the number of zero-crossing tangential acceleration (ZCA) \([229]\), dimensionless jerk metrics (DJ) \([223]\), spectral arc-length (SAL) \([226]\) and entropy of shape model and instantaneous acceleration (ESA). Except for SAL, the increase of metrics computed by the other approaches illustrates the deterioration of smoothness. Simulation results were analysed from three aspects, namely consistency, sensitivity and robustness. For consistency, the raw metrics computed with these five approaches were compared to see if the metrics could keep the same trend with the change of smoothness of trajectories. Secondly, the sensitivity of these approaches with respect to the change of smoothness was analysed. The improvements are computed as

\[
R_i = \frac{(\tilde{M}_i - \bar{M}_i)}{\bar{M}_i},
\]  

(4.4.1)

where \(\bar{M}_i\) is the metric value of trajectories with index of \(i = 1, 2, \ldots, 50\). The approach which had the most significant improvement with respect to \(\frac{1}{\text{SNR}}\) was the most sensitive to the change of smoothness since the smoothness deteriorates with the increase of severity level. Eventually, the robustness of these approaches was analysed. Firstly, the noisy trajectories were processed by a low pass filter with cut-off frequency of 14 Hz since the simulated frequency of the trajectories was 30 Hz. The normalised difference metrics were computed between the noisy and noiseless trajectories with approach \(h\) (\(h\) can be VP, ZCA, DJ SAL and ESA) as follows,

\[
(\bar{d}_j^h) = \left( \frac{\tilde{M}_j^i - \bar{m}_i}{\max(\tilde{m}) - \min(\tilde{m})} \right)_a,
\]  

(4.4.2)

where \(\bar{m}\) is the collection of the metrics (they are called ground truths in the rest of this chapter) of the 100 noiseless trajectories and \(\bar{m}_i\) is the metric of \(i^{th}\), \(i = 1, 2, \ldots, 100\) noiseless trajectory. In addition, \(\tilde{M}_j^i\) is the metric of the \(j^{th}\) \((j = 1, 2, \ldots, 100)\) noisy trajectories generated based on \(i^{th}\) noiseless trajectory.
Moreover, \( \max(\cdot) \) and \( \min(\cdot) \) selected the maximum and minimum data from the collection \( \tilde{m} \). Lastly, the probability density functions of \( \bar{d} \) (the collection of the normalised differences) were estimated by using \textit{ksdensity} function in Matlab. The whole evaluation process followed that introduced in [226].

### 4.4.2 Computer simulation result

Examples of generated trajectories are depicted in Fig. 4.5.

![Figure 4.5: These three graphs show trajectories in three levels. The left one is in level one for natural movements without involuntary movements and jerks. The middle trajectory is in the second level with two involuntary movements (with amplitudes of 0.225 and 0.25 meter) and jerks (with amplitudes of 0.025 and 0.0275 meter). The last one is in the third level with four involuntary movements (with amplitudes of 0.45, 0.5, 0.55 and 0.6 meter) and jerks (with amplitude of 0.05, 0.055, 0.06 and 0.065 meter). The red circles are examples of jerks in the second and third level.](image)

Fig. 4.6 illustrates the consistency characteristics of different approaches. Since the simulated trajectories were classified into two levels of severity of involuntary movements, the trend in individual levels were analysed first. For the results in level one (with blue colour), VP, SAL and ESA showed a consistent trend with the \( \frac{1}{SNR} \) (trajectory smoothness) where VP and ESA kept increasing from around 5 and 0.013 to approximately 17 and 0.1, while SAL decreased from -4 to around -5. The fluctuations were caused by the randomly generated values in the simulation (the same as follows). However, the consistent trend was hardly observed in ZCA and DJ. The metrics of the former fluctuate between 10 and 15, while those of
Figure 4.6: The metric given by these approaches tends to illustrate the consistency characteristics in various approaches used to evaluate the smoothness of trajectories in two severity levels. With the increase of numbers and amplitudes of both involuntary movements and jerks, the smoothness of the trajectories deteriorates. The first half (with blue from 1 to 25) is in the second level and the last half (with red from 26 to 50) is in the third level. The VP was computed with a threshold value of 0.01 meter/s and the temporal gap between two consecutive peaks was 100ms.
the latter decreased from -25 to around -75 and then increased to approximately -10. For the metrics of trajectories belonging to the third level, DJ, SAL and ESA show consistent trend, where DJ and ESA increased from about -25 and 0.2 to approximately -5 and 0.3, while SAL decreased from around -5.5 all of these approaches to -6. As for the other two approaches, the consistency was not obvious. Lastly, according to Sec. 4.2.1, the trajectories in level three were less smooth than those in level two. Therefore, metrics from these two levels should show the differences, which can be observed in the result for all these approaches. For example, the average metrics from VP, ZCA, DJ and ESA in level three were higher than the average metrics in the second level, while SAL showed lower metrics in level three than level two. All in all, SAL and ESA outperformed VP, ZCA and DJ in terms of consistency.

The second aspect was sensitivity (refer to Fig. 4.7), which analysed the change rate of metrics from various methods with respect to the change of motion smoothness ($\frac{1}{SNR}$). From the result, it is not hard to observe that ESA was the most sensitive approach since the improvement percentages changed from 0 % to around 1000 % for the second level and from 2000 % to about 2300 %. By comparison, metrics of SAL increased from 0 % to around 50 %. As for VP, it was quite sensitive to the change of the severity in the first 15 trajectories (increased from 0 % to about 100 %), but less sensitive for the rest. For ZCA, the improvement percentages were very minimal. Although DJ was very sensitive in the second level, the sensitivity gradually reduced in the third level. In addition, since the smoothness decreased linearly with respect to $\frac{1}{SNR}$, a line (orange lines) is fitted to the metrics generated by each method and computed the gradient of each line indicating the general sensitivity. This confirmed our conclusion that ESA was the most sensitive with the highest gradient.

Lastly, the robustness of the proposed approach was investigated. From Fig.
Figure 4.7: Sensitivity comparison of the five approaches with respect to the change in the severity of involuntary movements ($\frac{1}{\text{SNR}}$). A better evaluation technique is preferred to be sensitive to the small changes in the severity of involuntary movements and the change rate metric should be proportional to the change rate of severity. The blue and red lines show the improvement of metrics of the second and third severity levels with respect to the metric of the first trajectory computed with various approaches (refer to (4.4.1)). The orange lines are the regression lines of the corresponding metrics, showing the approximate improvement rate of each method. The value at the right bottom of each graph is the gradient of the regression line.
4.8, the performance of ESA was very close to SAL since they had a similar value (around 0.05) with the highest density. However, the range of ESA (0 to 0.75) was a little bigger than SAL (-0.1 to 0.5). By comparison, VP and DJ were not able to maintain the metrics with the influence of measurement noise since they had a large metrics range and large difference from the ground truth. Although the metric for ZCA was in a reasonable range (from 0.1 to 0.5), the offset from the ground truth was large (around 0.3).

Eventually, ESA outperformed SAL as it was more sensitive to the change of smoothness and also met the requirement of dimensionless, consistency and robustness.
4.4.3 Healthy subjects simulation data analysis

In the real data experiment, firstly, the smoothness of all the trajectories was evaluated using the same approaches considered in the computer simulation section. Additionally, by taking the duration into consideration, all these trials were classified into three levels of ability to perform the task by using three commonly used clustering methods, namely k-means clustering, Gaussian mixture model (GMM) and fuzzy clustering [241] so as to determine which clustering method suits the purpose of classifying motion trajectories into different levels of ability to perform upper extremity reaching tasks (severity levels of involuntary movements). Since the trials had also been classified by a human observer, a Cohen’s kappa correlation coefficient was computed between the human observer and the computer to indicate the assessment agreement. Higher coefficient values indicate a greater level of agreement between the utilised approach and the human observer. Except for the proposed method, the other four approaches, including VP, ZCA, DJ and SAL, were used for comparison.

4.4.4 Healthy subjects simulation result

Here the results of our preliminary real-data experiment with healthy subjects mimicking different severity levels of involuntary movements with their upper extremities while seated is presented. These motion trajectories were simulated according to the classification criteria (severity levels) in Table 4.1.

First of all, some examples of simulated trajectories and features are shown in Fig. 4.9. The first three rows are the trajectories in three axes, thereby showing the sub-movements (with red circles) and jerks (with green rectangles) more clearly. The third and fourth rows were shape models of these trajectories with curvature and torsion, while the last row is the instantaneous acceleration. As for the columns, levels one to three (refer to Sec. 4.2.1) are in the first to the third columns. As can
be observed, from the first to the third level, the number of sub-movements and jerks increased from 0 to 4 (two sub-movements and two jerks) and 9 (four sub-movements and five jerks) respectively. Correspondingly, the curvature, torsion and instantaneous acceleration were increasingly noisy.

Figure 4.9: Examples of trajectories (first three rows), shape models, including curvatures (fourth row) and torsions (fifth row), and instantaneous accelerations (sixth row) are illustrated for three levels of the ability to perform an upper extremity reaching task (columns one to three corresponding to levels one to three of the severity of involuntary movements). The red circles show examples of sub-movements and green rectangles are examples of jerks.

Secondly, the distributions of various features, including the duration of the task, as well as metrics computed with various approaches, are shown in Fig. 4.10. The first box plot was the distribution of durations for three severity levels of involuntary movements. As can be seen, although there was some overlap between two consecutive levels, the inter-quartile ranges followed the definition of severity.
levels (refer to 4.1). The remaining five graphs show the metrics computed with five approaches. First of all, it is obvious that, in VP, ZCA, DJ and ESA, the more severe involuntary movements were associated with higher values, while it was opposite in SAL (the more severe involuntary movements had lower metrics). However, this trend between the first and the second levels in DJ was very small. Secondly, although every approach showed a certain degree of overlap between two consecutive levels, DJ showed the worst situation since the metrics of its first and second levels were very similar. As for the other four approaches, VP and ZCA showed a larger overlap between the first and the second levels than SAL and ESA and VP had the largest overlap between the second and the third levels. Generally speaking, the differences between these three levels of ESA were more obvious than in the other four approaches. Since these metrics were used to classify motions into various severity levels, the correct classification rate would be impacted by their distribution in different levels. If the distribution of a metric in two levels overlapped a lot, such as level 1 and level 2 in DJ, the recognition rate for this metric in these two levels, or at least in one of these two levels, would be low. Therefore, from Fig. 4.10, it can be seen that ESA outperforms other features.

Table 4.4: Cohen’s Kappa ( \( p < 0.05 \)) between various automated approaches and the human observer.

<table>
<thead>
<tr>
<th>Approach</th>
<th>K-means</th>
<th>GMM</th>
<th>Fuzzy Clustering</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>VP</td>
<td>0.6875</td>
<td>0.6926</td>
<td>0.7250</td>
<td>0.7017</td>
</tr>
<tr>
<td>ZCA</td>
<td>0.7625</td>
<td>0.8000</td>
<td>0.7500</td>
<td>0.7708</td>
</tr>
<tr>
<td>DJ</td>
<td>0.5500</td>
<td>0.5631</td>
<td>0.5500</td>
<td>0.5544</td>
</tr>
<tr>
<td>SAL</td>
<td>0.7875</td>
<td>0.7875</td>
<td>0.7500</td>
<td>0.7750</td>
</tr>
<tr>
<td>ESA</td>
<td>0.8250</td>
<td>0.8250</td>
<td>0.8500</td>
<td>0.8333</td>
</tr>
</tbody>
</table>

Table 4.4 shows the Cohen’s Kappa between five automated approaches and the human observer generated by three clustering approaches. Generally speaking, ESA always gave the highest agreement (0.8250 for K-means and GMM, with 0.85 for fuzzy clustering). In addition, SAL and ZCA had a similar performance with kappa
values slightly smaller than 0.8 in the majority of cases. By comparison, VP was a little better than DJ. But these two approaches were worse than the other three methods. Moreover, the comparison of these three clustering approaches suggested that the fuzzy clustering was more suitable for automated assessment of the severity of involuntary movements, thereby evaluating the ability to perform reaching tasks in daily living in terms of kinematics.
4.5 Summary

One issue in a tele-rehabilitation system is how to evaluate the kinematic performance of patients in ADLs remotely and automatically based on the collected data from OBMCDs without the presence of experienced therapists. This study investigated a novel method to evaluate the ability of individuals with involuntary movements to perform reaching tasks involving the upper extremity. To achieve the goal, the smoothness of motion trajectories and the duration used to finish the activity are assessed. For the former, the entropy of the two components of a trajectory, including its shape and instantaneous acceleration, were used to capture the appropriate performance indices. Simulations in this study confirmed the superiority of the propose method in terms of its consistency, sensitivity and robustness, while the real-data experiment illustrated the high agreement between the human observer and the proposed automated approach.

However, since this was a preliminary study, there are some areas that require further attention. Firstly, being an affordable device, the Microsoft Kinect is not as accurate as other more expensive commercially available products, such as VICON. Therefore, the lower resolution (especially in Z axis) hinders the Kinect in identifying small movements of human joints. In other words, jitters or tremors with minimum amplitude may not be captured by the Kinect. Therefore, at present, it can only be utilised to detect involuntary movements with relatively large amplitude. In real applications, the Kinect should be optimised so that it can detect involuntary movement of few millimetres. However, this study illustrates the ability of the proposed approach to evaluate a person performing reaching task particularly in a non-clinical environment. It is noteworthy that although the real-data experiment consisted of a seated activity involving the arms, the proposed approach is not limited to upper body movements. Other applications will be further investigated in future work. Secondly, healthy subjects, instead of patients with involuntary
movements, were evaluated. Therefore, the proposed approach should be further validated and analysed with people who have involuntary movements and other movement impairments.

As for future works, first of all, the proposed approach should be evaluated with data from patients with dyskinesia so that the effectiveness of this method can be well studied in real-world environment. Secondly, since smoothness is only one criterion to evaluate the ability of a patient with movement disorders in performing activities in daily living, more features should be investigated so that their ability can be evaluate more comprehensively from the therapeutic point of view.
Chapter 5

Mobile Cloud-based Physical Tele-rehabilitation System: A Prototype

5.1 Introduction

Resource deficiencies in the healthcare industry to cater the rapid changes in demography entail the development of new rehabilitative practices [242] enabling therapists to provide their services to remote communities. Since the first article using the term “tele-rehabilitation” published in 1998[12], this new rehabilitation practice paradigm has been widely studied for its capabilities in reducing patients’ commute to hospital and saving time and cost for both clinicians and patients[243]. Tele-rehabilitation can be defined as a set of instruments and protocols aimed at providing access to rehabilitation services for patients at geographically distant locations[244].

Carignan et al. [245] mentioned that one of the challenges in developing a tele-rehabilitation system is to reduce the delay in the data communication and processing, especially where real-time bio-feedback is required. As is observed, the major delay in the tele-rehabilitation is caused by the low bandwidth of the Internet, large amount of data and the low computational power of the computer used to process captured data. Therefore, the delay resulted from the increase
in users can be partially addressed by increasing the computational power of the computer hosting the server side of the tele-rehabilitation system. Furthermore, the increase in users is associated with a significant increase in captured data from various sensors, which requires the flexible in capacity of the database. Additionally, applying pervasive computing in healthcare, such as tele-rehabilitation, is becoming popular in the recent years [246, 247], which leads to a challenge on how to integrate existing technologies to a tele-rehabilitation system.

With recent developments in ICT, Mobile Cloud Computing (MCC) [248] has emerged to bring cloud computing [242] power to mobile devices. MCC[248] can be considered as the ideal candidate to accommodate majority of the above criteria.

Firstly MCC inherently provides an architecture to support various mobile devices, such as smartphones, tablets, laptops and so on, which means that users of tele-rehabilitation services can readily access the service with less restrictions on the time and location.

Secondly, similar as generic cloud computing, MCC also provides an elastic approach to support the increasing number of users. One example is Amazon Web Services (AWS) [249], which enables users to expand the computation power and data storage space easily. More importantly, users deploying their services in cloud only pay for the resources they use, which is also called as “pay-as-you-go” (PAYG), thereby reducing the cost significantly compared to traditional approach to deploy services.

Thirdly, the computation power in MCC enables to perform calculation fast enough to offer patients real-time bio-feedback, which is critical in tele-rehabilitation [250] to improve the effectiveness of tele-rehabilitation session[251, 252, 253]. Because of the limited computational power and battery resources, mobile devices are unlikely to handle this task for a long time. Therefore, offloading the computation tasks to the cloud reduces the load in mobile devices and enables tele-rehabilitation
users to access the service as long as possible.

Although MCC has provided a well-designed platform to establish tele-rehabilitation services, there exist some gaps to be solved for this particular application. One of the most critical issues is how to reduce the energy consumption in mobile devices so that patients are able to access tele-rehabilitation services as long as possible. In addition, another open question is how to design a tele-rehabilitation platform running on mobile devices to provide rehabilitation exercise performance evaluation.

This chapter primarily seeks the answers for the previously mentioned questions. The main contribution of the study is a multi-level data encoding scheme for optimisation of computational offloading and data transferring. Although a vast array of studies discussed about strategies to optimise the computational offloading and data transferring [254, 255], no study focusing and customising the algorithm for biomedical signals was found. In order to answer the second question, an architecture of MCC-based tele-rehabilitation platform was proposed. By taking the contributions made in previous chapters, this platform can be deployed on mobile devices to assess the kinematic performance of patients during rehabilitation exercises. Furthermore, the concept of analysis oriented decision support system (AODSS) and the security service layer (SSL) are also integrated in the platform. The former utilises the huge amount of data to analyse the performance of patients from different levels of granularities, while the latter protects the information of data spanning from acquisition, storing and transferring though out the platform. These two are research challenges in conventional clinical decision support system (CDSS) [256] and will be discussed in Sec. 5.3.4 and 5.3.5 respectively.

The rest of the chapter is organized as follow. Sec. 5.2 introduces our main contribution, the encoding scheme and approach to optimise the energy of computation and data handling. Furthermore, an overview of the system architecture,
as well as the concepts of AODSS and SSL are presented in Sec. 5.3. Additionally, the simulation and real data experiments regarding the multi-level encoding scheme are presented in Sec. 5.4 where the demonstration of the interfaces of the tele-rehabilitation platform is included. The final section presents brief concluding remarks.

5.2 Multi-level Data Encoding Scheme

This section is dedicated for solving one of the limitation, namely limited power for computing and data transferring, in applying mobile cloud computing in tele-rehabilitation field by utilising the characteristics of biomedical data collected from various types of sensors.

5.2.1 Protocol

Due to the limited battery resources in a mobile device, how to reduce the power consumption is an open question in MCC field. In this study, a multi-level data encoding scheme is proposed so that the data can be encoded differently to reduce the quantity and time during transfer, thereby reducing the power utilized to transfer data. However, different encoding schemes require various computational times. Generally speaking, if the amount of data after encoding is smaller, the time utilised to encode the original data is longer. Therefore, it is crucial to find an approach to determine which level of the encoding scheme should be utilised.

Fig. 5.1 shows the data encoding schemes motion rehabilitation. The encoding scheme in the higher position indicates that the data amount after encoding is smaller than approaches in lower levels.

For instance, in human motion capturing, the details of each encoding scheme are introduced as follows.

- 3D Motion Trajectory
In the majority of the optical-based portable motion capture devices such as Kinect and Creative Senz3D, human motions are captured in terms of the positions of the joints in the forms of \( \gamma_n(k) = [\bar{x}_n(k), \bar{y}_n(k), \bar{z}_n(k)]^\top \), where \( \bar{x}_n(k), \bar{y}_n(k) \) and \( \bar{z}_n(k) \) are 3D positions of \( n^{th} \) joint \( (n = 1, 2, \cdots, N) \) on the X, Y and Z axes at time \( k \) in the traditional Cartesian coordinate system.

Compared to VGA video, a frame of 3D trajectories for \( N \) joints is \( 24 \times N \) bytes, where, for example, \( N = 20 \) in Kinect version 1 and \( N = 16 \) in Creative Senz3D. Since both videos and trajectories are collected from motion devices, it is assumed that the same power or energy is required to retrieve the data.

- **Elbow point technique**

In a motion trajectory, each point has its own importance and its contribution to the shape of trajectory is different from others. An array of points lying on a straight line can be represented by only two points at the two end-point of the line. Therefore, points at the middle of the straight line can be removed to reduce the computational cost and data storage. Points lying on a curve are called “elbow points” which are essential points to form the shape of the
trajectory [257]. Discrimination of “straight points” and “elbow points” can be based on curvature. Curvature at a point $p_n(k)$ is defined as:

$$\kappa_n(k) = \frac{\|v_n(k) \times a_n(k)\|}{v_n^3(k)}, \quad (5.2.1)$$

where

$$v_n(k) = \left[ \left( \frac{d\bar{x}_n(k)}{dk} \right)^2 + \left( \frac{d\bar{y}_n(k)}{dk} \right)^2 + \left( \frac{d\bar{z}_n(k)}{dk} \right)^2 \right]^{\frac{1}{2}}. \quad (5.2.2)$$

A point is marked as an “elbow point” when its curvature is larger than a specific threshold $\varepsilon$. Conversely, a point is marked as “straight point” if its curvature is larger than or equal to zero and less than the specific threshold $\varepsilon$. In elbow method, points with curvature $\kappa < \varepsilon$ will be removed from the trajectory. The original trajectory can be approximately reconstructed from the new trajectory if the coordinates of the elbow points and their sequential orders are determined. The new trajectory which includes only elbow points obviously requires less computational cost than the original trajectory. This technique is illustrated in Fig. 5.2 for the 2D case. The technique can be applied in 3D case[257]. Here, the function $f(\bar{x}) = \bar{x} \sin \bar{x}$ is plotted where $\bar{x}_1 = 0, ..., \bar{x}_{100} = 10\pi$ and $\bar{x}_{n+1} = \bar{x}_n - \bar{x}_{n-1}$. As the figure shows, the new trajectory formed by elbow points is almost identical to the original trajectory with all the points. In this example, the value of 0.05 is used for the threshold $\varepsilon$, hence half of the points from the trajectory have been removed. Depending on the application, this threshold value can be chosen accordingly.

- Shape model

To tackle some situations that the bandwidth is insufficient to transfer elbow points, shape models can be further derived to encode motion trajectories. Apart from the curvature in (5.2.1), torsion is needed since the motion trajectories are in three dimensions. It is derived as,
Figure 5.2: An illustration of the elbow point concept. Red points are elbow points when $\kappa > 0.05$. Blue points with curvature less than 0.05 is removed from the trajectory.

\[
\tau_n(k) = \frac{(v_n(k) \times a_n(k)) \cdot j_n(k)}{\|v_n(k) \times a_n(k)\|^2}, \tag{5.2.3}
\]

where

\[
j_n(k) = \sqrt{\left(\frac{d^3 \bar{x}_n(k)}{dk^3}\right)^2 + \left(\frac{d^3 \bar{\bar{y}}_n(k)}{dk^3}\right)^2 + \left(\frac{d^3 \bar{z}_n(k)}{dk^3}\right)^2}, \tag{5.2.4}
\]

is the jerk of the motion trajectory.

From the derivation, it is observed that the motion trajectory is encoded from three dimensions ($X, Y$ and $Z$) to two dimensions ($\kappa$ and $\tau$).

- Performance Measurement

Two methods can be utilised to encode and measure the performance of the tele-rehabilitation exercises, including smoothness based and elbow point based measurements. As for the former[258], it is specially designed for tele-rehabilitation service users with neurological movement disorders, such as dyskinesia, that involve large amplitude involuntary movements, which leads to less smooth motion trajectories than healthy people. Due to the fact that
the shape model [259] is very sensitive to noise in motion trajectories, the sub-
movements and jerky movements are also shown in the shape model. As a
result, the entropy of the shape models of these trajectories are computed to
represent the severity of involuntary movements, thereby indicating the abil-
ity of patients to perform tele-rehabilitation exercises given by their thera-
pists. Another method can be used for this encoding scheme is the algorithm
in [257]. In this approach, the authors used longest common sub-sequence
(LCSS) to match the trajectory from the patient and the corresponding one
from the therapist (model motion trajectory) and gave a score for performance
measurement.

Table 5.1: Notations used in the process of estimating power consumption for data
encoding and transfer. Here $i = 1, 2, 3, 4$ for motion tele-rehabilitation and $i = 1, 2, 3$
for respiratory tele-rehabilitation.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{b}$</td>
<td>byte</td>
<td>the size of memory occupied by a double value</td>
</tr>
<tr>
<td>$\hat{D}_i$</td>
<td>-</td>
<td>dimension of encoded data of each frame with $i^{th}$ encoding scheme</td>
</tr>
<tr>
<td>$\hat{T}_i$</td>
<td>frame</td>
<td>the length of exercise data encoded by the $i^{th}$ encoding scheme</td>
</tr>
<tr>
<td>$\bar{U}$</td>
<td>-</td>
<td>the number of monitored points in the motion</td>
</tr>
<tr>
<td>$\hat{S}_i$</td>
<td>byte</td>
<td>size of encoded data at the scheme $i^{th}$</td>
</tr>
<tr>
<td>$\bar{u}_i$</td>
<td>watt</td>
<td>the energy consumed by the mobile device to encode 1 frame at the scheme $i^{th}$</td>
</tr>
<tr>
<td>$\bar{v}$</td>
<td>watt</td>
<td>the energy consumed by the mobile device to transfer encoded exercise data to the cloud</td>
</tr>
<tr>
<td>$\hat{P}_{L,i}$</td>
<td>watt</td>
<td>the energy consumed by the mobile device to encode entire trajectory at the scheme $i^{th}$</td>
</tr>
<tr>
<td>$\hat{P}_{T,i}$</td>
<td>watt</td>
<td>the energy consumed by the mobile device to transfer entire encoded data to the cloud at the scheme $i^{th}$</td>
</tr>
<tr>
<td>$\hat{B}$</td>
<td>kbps</td>
<td>the speed of uploading data to the server</td>
</tr>
</tbody>
</table>

5.2.2 Determine encoding level

The introduction of various encoding approaches naturally raises the question of
determining the encoding level that should be used depending on the computational
power and the speed used to upload data to the cloud from the mobile device. [255] introduces an approach for offloading decision by estimating how much power is used for computations and transfer of data. This method is adopted to determine which encoding level should be utilised with respect to the bandwidth. Notations utilised in the estimation process are shown in the Tab 5.1. Unlike the approach introduced in [255], the computation time in the server is not considered in our case. The reason is that the data communication between the server and the mobile device uses asynchronous channels which means the channels are not blocked when the server is engaged in the computation. Therefore, there is no idle time in the mobile device. Further in our case, it is hard to estimate the number of instructions required by the computation for data encoding. As a result, numbers of unit energy consumption ($\bar{u}_i$) utilized to perform encoding for each frame of data is recorded, which can be retrieved automatically by the system.

The formula to calculate data size:

$$\hat{S}_i = \hat{b} \times \hat{D}_i \times \hat{T}_i \times \hat{U} \quad (5.2.5)$$

The formula used to calculate local computational power with scheme $i$ is

$$\hat{P}_i^L = \hat{P}_{i-1}^L + \bar{u}_i \times \hat{T}_{i-1} \times \hat{U}, \quad (5.2.6)$$

where $\bar{u}_1 = 0$; $\bar{u}_2$ is the average power consumption to determine whether a point is an elbow point and it includes the calculation of curvature; $\bar{u}_3$ is the average power consumption to calculate torsion value of 1 point; $\bar{u}_4$ is the average power consumption to calculate the performance.

The formula used to calculate data transfer power is

$$\hat{P}_i^T = \frac{\hat{S}_i}{\hat{b}} \times \bar{v} \quad (5.2.7)$$

where $\bar{v}$ is the energy consumption for uploading data in 1 second and $\hat{b}$ is the network speed. Tab 5.2 summarises the power consumption of various encoding scheme and encoded data transfer for motion tele-rehabilitation.
### Table 5.2: Power consumption of various encoding scheme and encoded data transfer for motion tele-rehabilitation.

<table>
<thead>
<tr>
<th>i</th>
<th>D</th>
<th>T</th>
<th>Data size</th>
<th>Energy consumption of local computation</th>
<th>Energy consumption of data transfer</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>T</td>
<td>$24 \times \tilde{T} \times \tilde{U}$</td>
<td>$\tilde{u}_2 \times \tilde{T} \times \tilde{U}$</td>
<td>$P_3 + \tilde{u}_4 \times 1 \times \tilde{U}$</td>
</tr>
<tr>
<td>2</td>
<td>3</td>
<td>$\approx 0.5 \times \tilde{T}$</td>
<td>$12 \times \tilde{U}$</td>
<td>$\tilde{u}_4 \times 0.5 \times \tilde{T} \times \tilde{U}$</td>
<td>$8 \times \tilde{T} \times \tilde{U} \times \tilde{U}$</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>$\approx 0.5 \times \tilde{T}$</td>
<td>$8 \times \tilde{U}$</td>
<td>$\tilde{u}_4 \times 0.5 \times \tilde{T} \times \tilde{U}$</td>
<td>$8 \times \tilde{T} \times \tilde{U} \times \tilde{U}$</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td>$\approx 0.5 \times \tilde{T}$</td>
<td>$1 \times \tilde{U}$</td>
<td>$\tilde{u}_4 \times 0.5 \times \tilde{T} \times \tilde{U}$</td>
<td>$8 \times \tilde{T} \times \tilde{U} \times \tilde{U}$</td>
</tr>
</tbody>
</table>

Original trajectory: $i \tilde{T} \times \tilde{U}$
Elbow points: $i \tilde{T} \times \tilde{U}$
Shape model: $i \tilde{T} \times \tilde{U}$
Overall performance: $i \tilde{T} \times \tilde{U}$
The following cost function is used to determine which encoding scheme is to be used

\[
\hat{E}(i) = \arg \min_i \left( \hat{P}_L^i + \hat{P}_T^i \right)
\]  

(5.2.8)

In some cases, if \(|\hat{E}_{i_1} - \hat{E}_{i_2}| < \epsilon\), and \(i_1 < i_2\), \(i = i_1\) is selected so that more data can be transferred to the cloud by consuming similar power. Here \(\epsilon\) is a small constant value.

5.3 System Architecture

In this section, the detail regarding the implementation of the MCC-based tele-rehabilitation platform is introduced.

5.3.1 Development platform

As for the platform to deploy the server side of the proposed system, Microsoft Windows\(^\text{®} 2012\text{R2}\) is selected since it is stable, easy to configure and well supported by various cloud computing providers, such as Amazon EC2. In addition, the programming language utilised to develop the overall system was C# due to the following three reasons.

- Microsoft Kinect utilised in our tele-rehabilitation system support C#.
- C# is well supported by Windows operating systems;
- The Windows Communication Foundation (WCF) can be integrated as the basic communication framework for the overall system, which can be easily developed by using C#.

Therefore, as per the selection of deploying platform and programming language, the integrated development environment (IDE) utilised to implement the whole system is Microsoft Visual Studio 2013. In order to use the latest functions of C#, .Net 4.5 is adopted as our common language run-time (CLR).
5.3.2 System (hardware) overview

From Fig. 5.3, it can be seen that there are three main components in the tele-rehabilitation platform: therapist side, server side and patient side. Except for the server side, there is only one instance of the other two components in the figure, which can actually be multiple instances simultaneously, representing a number of therapists and patients accessing the tele-rehabilitation services at the same time.

At current stage, due to the limitation of Kinect, it is unable to connect to tablet and mobile phone. However, it is noteworthy that a trend has been seen, in recent years, to integrate depth cameras into mobile devices, such as Dell® Venue™ 8 7000 series tablets and HTC® M7 smartphone. Therefore, our proposed tele-rehabilitation platform not only caters to the existing sensors, but also can be utilised when the sensors are integrated into mobile devices.

5.3.3 System (software) architecture overview

In this subsection, a detailed software architecture of the system is provided as follows.

- Therapist side
Primary focus of this side is on the inclusion of patient profile management, building exercise models (example can be found in Fig. 5.5) and visual (on-line or off-line) review of the exercise data and analysis result collected from various sensors and ADOSS (refer to 5.3.4). Exercise models built in this side have two major aims. First of all, the models can be downloaded by patients and utilised as a guidance in performing tele-rehabilitation exercises. Secondly, models built by therapists can be used as references for tele-rehabilitation exercise performance evaluation. As for the data flow, except for life streaming, which is introduced in the patient side.

Figure 5.4: Data flow model in therapist side (except for on-line review)

Fig. 5.4 shows the data flow between the cloud (server side) and the therapist side. Symbols in the graph with different colours indicate the data and requests from various sources and their corresponding responses. Here squares and downward triangles are the encoded version of data represented by circles, while upward triangles and diamonds represent database operations without storing new data. Additionally, as for the responses, there are two methods to display information. One is showing a confirmation statement to notify the therapists whether the requested operation is successful or not. The second
approach is visually displaying information obtained from the cloud, such as histograms. In addition, the data follow of off-line review (with green symbols) in Fig. 5.4 is an abstract process and detailed information is discussed in Sec. 5.3.4.

The example of the user interface for therapist side is shown as follow.

![Therapist Interface Example](image)

Figure 5.5: Example of making model motions of the therapist side with Kinect.

In Fig. 5.5, it can be seen that a therapist is able to make a specific model for a particular patient according to the condition of this patient. One or multiple joints can be selected as the important joints in this exercise model. Lastly, the functions of replay and crop of the recorded therapist’s model motion were incorporated in the system to adapt the needs of the therapist.

- **Patient side**

  The patient side of our platform provides interactive tele-rehabilitation services and passive exercise monitoring capabilities. The data flow is shown in Fig. 5.6. For interactive tele-rehabilitation, first of all, the patient requires the exercise model created by his/her therapist from the cloud (shown with blue symbols), which is later utilised in the rehabilitation exercises as references and streamed to the patient’s mobile devices (with purple symbols). After that, the performed exercise motions of the patient are recorded and sent to
Figure 5.6: Data flow model in patient side (with on-line review for therapists)

the cloud (with red symbols). It is noteworthy that, instead of sending video or audio data like typical tele-rehabilitation systems, the encoded data is sent in our system.

When the cloud (server side) receives the recoded exercise information, signal processing techniques are applied to filter out the noise and also to extract relevant features. The data is stored in patient exercise database for off-line review, and also used to provide corresponding biofeedback particularly for performance measurement and assessment through the comparison between patients acts and therapist model. The bio-feedback is denoted with blue triangles and presented in various forms (refer to Sec. 5.3.6). Lastly, if a therapist has enough bandwidth and choose to review the patient’s exercise online, he/she just has to register a channel, through which the processed motion information is forwarded by the cloud to the therapist. As for the passive exercise monitoring, the data follow the path indicated by red symbols.

In addition, to solving the disconnection issue in mobile terminals, a small
local database is maintained in the patient side to store the most recently used model motion information and the patient’s exercise motion data. Therefore, even when the patient is unable to connect the Internet temporarily, he/she is still able to use the tele-rehabilitation system. When the connection is established again, the patient’s motion data can be synchronised to the main database in the server side for further analysis. Indeed when the Internet is not available, the patient is unable to receive bio-feedback and performance measurement updates if the computational power in their mobile device is insufficient.

The example of the patient side is shown in Fig. 5.7. There are five components in this figure. The left side of the figure shows the front view of the patient skeleton with the guidance trajectory. The top right of the figure shows the top view of the tracking joint (yellow circle) and the guidance point (light blue circle). The middle right chart shows the history of the performance measurement for a period of time and the meter in the bottom right shows the performance’s level of the current session. Additionally, the message where the Start word appears shows the incoming instructions as a form of visual feedback. The aim of the exercise is to both match the red circle on the skeleton onto the red circle on the guidance (blue trajectory).

If the Internet connection is available, the motion trajectories of the patient is encoded with one of the approaches introduced in Sec. 5.2 depending on the computational power of the mobile device as well as the upload speed of the Internet. After each session, the performance measurement, which is also the level 4 encoder, is computed either on the local device or in the cloud platform. Eventually, this measurement is shown in the patient side as an overall bio-feedback.
5.3.4 Analysis oriented decision support system

In this subsection, the concept of analysis oriented decision support system (AODSS) is discussed, which is a combination of the concept of the service-oriented decision support system (SODSS) [260] and the clinic decision support system (CDSS) [256].

As the AODSS integrated in the MCC-based tele-rehabilitation platform, it contributes to the establishment of mobile CDSS mentioned in [256], which was considered as a research challenge in CDSS. The contribution of this study to AODSS is that data is analysed with different granularities, which can be collected from various types of sensors and stored in various databases in the cloud. This method is embedded into the data flow (with green symbols) shown in Fig. 5.4.

The concept of AODSS is illustrated in Fig. 5.8. The first column shows the source of data, which is not included in the data flow of information query and response between therapist side and the sever side. Data from the sensor is stored in one or multiple databases in the cloud which is highly likely to be distributed into different servers to maintain the response speed with a large number of queries. After raising a query from the therapist side (user layer), the data extraction tool
retrieves related data from the data layer, which is further processed by using various integrated data mining tools (in data analysis layer) and mining algorithms, such as clustering, classification, regression, attribute importance, anomaly detection, association and feature extraction. Eventually, the data is visually presented to the therapist to assist him/her to make further decisions.

The main contribution of this AODSS is that it is able to analyse a huge amount of data collected from Kinects and provide supportive information in various granularities to therapists based on the encoding levels of data sources.

Use stroke tele-rehabilitation as an example. Since the kinematic performance of a post-stroke patient can be assessed through the movement of limbs, AODSS first retrieves related data from the database. Due to the fact that data used by AODSS is encoded with various encoding schemes, AODSS is able to provide results in various granularities. For instance, AODSS can provide a histogram showing the change of performance (computed by the performance measurement algorithms mentioned in Sec. 5.2.1) of a patient in tele-rehabilitation sessions during a day, a month or even longer period. After receiving the analysis results, the therapist will have a general idea about the effectiveness of the therapy and the progress of the patient. If the therapist wants to know the movement patterns of the patient in each session, AODSS is also able to analyse the encoded data at motion trajectory,
elbow points or shape model level and generate a report to show the details in each session. These information assists the therapist to understand the detailed reason why the patient is under-progressing or improving. Thus, the therapists can re-evaluate the exercise components in the session to suit the patient’s capabilities to achieve better rehabilitative outcomes or have the confidence to encourage the patient to stick with the therapy.

5.3.5 Security service layer

As mentioned previously, security and privacy are one of the challenges faced in CDSS. Therefore, AODSS is inevitably impacted by this challenge. Security service layer (SSL) introduced in this subsection is a concept that has the potential to solve this challenge in the proposed AODSS, as well as the MCC-based tele-rehabilitation platform.

Since many parties such as patients, clinicians, health-care administrators, insurers, and researchers are involved, an End-to-End security control will be applied to enhance the protection of data. The security associations is managed by a Security Association Manager to coordinate the communication groups [261]. The Security Association Manager includes a Security Policy Collections, Security Association Flows, and Distributed Logs. The Security Policies Collections specifies which security level and which rules need to be used in associations with the types of parties. The Security Association Flows contains routing algorithms, keys, encryption schemes, protocol modes, and flow-level lifetime. The Distributed Logs store the logs of both end points in order to avoid fragment of logs stored in different repository. The Security Association Manager can collaborate with external Certificate Authority [262] to enhance authorization processes. This cooperate allows a patient or the service provider to rely on their identity provider to provide other e-Healthcare providers with only the specific data necessary to complete the transaction.
5.3.6 Optimise biofeedback

Traditionally, the feedback with regards to the rehabilitation exercises is given directly from therapists, which is the most effective approach. Therefore, this method is implemented in our system as a bio-feedback option. However, it is only available when the therapist reviews the exercises of the patient on-line so that the auditory message is routed by the server side from the therapist to the selected patient and vice versa for communication. It means that this method relies heavily on the large bandwidth, which is not always available for MCC. Therefore, in our system, there are three other types of bio-feedback, which can be selected according to the availability and the bandwidth of the allocated network connection.

First of all, since the model motions recorded by therapists are utilised to guide patients to perform tele-rehabilitation exercises, patients are able to receive the visual bio-feedback directly by looking at the differences between their motion trajectories and those from the model motions. Therapists’ 3D positions of joints, rather than video, are recorded in therapist side with a Kinect. The data is transferred to the server (in cloud) so it can be further process and stored, which is then downloaded by patients and used as model motions. However, due to the fact that the motion trajectories on the screen are always in 2 dimensions and the third one is unable to be observed by the patient, the top down view of both the model and patient’s motion are presented in the other window to illustrate the differences in the third dimension. By looking at the screen, the patient is able to see the gap between his/her motion with the model, thereby correcting their motion in time. Since the model motion is always available, either from the cloud (when the speed of the Internet is fast enough) or from the local temporary database (when the bandwidth is small or the Internet is disconnected), this type of bio-feedback is always available.

Secondly, auditory bio-feedback is also an option in the introduced platform to
correct patient’s tele-rehabilitation exercises. For instance, fast rhythm indicates that the patient should move the limbs or other body parts involved in the tele-rehabilitation session faster and the slow rhythm tends to slow down the patient’s movement. To modify the rhythm of the music, the speed of the patient’s movement is derived from their motion trajectories in the server side and is compared with that of the model motion. Eventually, a ratio is generated and sent to the patient side to change the rhythm.

Lastly and most importantly, although performance measurement is used as an encoding scheme (refer to 5.2), this value can also be utilised as a feedback indicating the performance of doing specific rehabilitation exercises. As is mentioned in the previous subsection, two approaches (including elbow-point based and entropy based) could be utilised to represent the performance of a rehabilitation exercise session. This feedback not only gives both the patient and the therapist an overview about the ability of the patient to perform certain tasks, but also is able to stimulate the patient to perform exercises more frequently, thereby achieving higher performance measurement.

5.4 Experiment Results

5.4.1 Computer simulation for multi-level encoding scheme

In this subsection, a numerical example is presented to demonstrate the effectiveness of the multi-level encoding scheme for motion exercise monitoring in saving energy in mobile devices. First of all, 18 motion trajectories were generated with $L_1 = 1000$ for each exercise. It is further assumed $u_1 = 0 \mu J$, $u_2 = 8.8 \mu J$, $u_3 = 21 \mu J$ and $u_4 = 0.6 mJ$. These assumptions were based on execution time for each specific task which directly related to the complexity of the computations and transmission. We also assumed that a connection setting which cost $v = 50 mJ$ for transferring data in 1 second was used. The total energy consumption was also examined for
both local computations and data transfer for different uploading speeds from 0 kbps to 2000 kbps. The numerical experimental result is shown in Fig. 5.9. In the figure, the minimum energy for the uploading speed from 0 kbps to 50 kbps can be achieved if level 4 of encoding scheme is used. Similarly, level 3 is used for uploading speed from 50 kbps to 200 kbps, level 2 for uploading speed from 200 kbps to 1200 kbps and level 1 for above 1400 kbps. At the uploading speed of 50 kbps, the total of energy spending for levels are 21.6 J, 10.9 J, 7.5 J and 1.3 J. The difference of energy between levels can be up to 20.3 J and this means the use of encoding schemes can save up to 20 times of energy for low connection speeds. Indeed, for better connections, for example, uploading speed above 1400 kbps, sending all data to the cloud for processing is the best in terms of saving energy and preserving information.
5.4.2 Real-data experiment for multi-level encoding scheme

To further illustrate the performance of the proposed multi-level encoding scheme, a preliminary real-data experiment with simulated motion trajectory data and four various encoding methods was performed. In this experiment, software named BatteryMon (V2.1 build 1008) and NetBalancer were utilised to monitor the energy consumption and to control the speed of the Internet connection. Furthermore, the encoding algorithms and data transferring program were implemented in C#. The experiment was done on a laptop with CPU of Inter® Core™ i7-3740QM and Wifi card of Intel® Centrino® Ultimate-N 6300AGN. To eliminate the influences of other software, first of all, BatteryMon was initialised for half an hour without running any unnecessary program to estimate the energy consumption of the laptop in idle state. All the following measurements were subtracted by this energy to compute the power utilised in order to compute the proposed encoding methods or to transfer the data to the cloud.

After that, a 3D trajectory with length of 1000000 frames was collected and further encoded with the other three encoders. Each of the encoding was repeated for 10 times (reliability test) where the average energy consumption of computing for each encoder was recorded in terms of per frame.

![Figure 5.10: Setup of the real-data experiment](image)

Furthermore, the setup of the real-data experiment is shown in Fig. 5.10. The
data receiver component of the data transferring program was deployed in a desktop connected to a wireless router with a network cable to secure the speed of the data transfer. Additionally, the laptop is connected to the router with theWifi card like a normal mobile device. Moreover, NetBalancer was used to control the Internet upload speed of the laptop to simulate the environment with different conditions of the Internet. The upload speed was limited from 80 to 800 Kbps with a step of 80 Kbps for testing because the upload speed on 3G/4G is about 0.45Mbps to 1.93Mbps [263]. Eventually, the average energy utilized to transfer one frame of the encoded data with each respective speed was recorded.

Lastly, the total power consumption for each encoding approach and various uploading speed for a trajectory with 10000 frames was computed. The result is shown in Fig. 5.11.

![Figure 5.11: Result of real-data experiment for the multi-level encoding scheme.](image)

From the result, a similar trend as the simulation was observed. When the upload speed was smaller than about 160Kbps, level 4 encoder was the best option since it consumed the lowest energy to encode the motion data and transfer the
result to the cloud, while the level 3 encoding approach should be selected when the upload speed of the Internet was ranging from 160 to 480 Kbps and subsequently, level 2 encoder should be utilised. Using real experiment’s data, it shows that the level 1 encoding method always consumes more energy than level 2 and level 3 encoder in this scenario.

According to the measurement, the average energy expenditure to capture 10000 frames data requires around 1232J. When the upload speed is very slow (80Kbps), the energy can be saved for more than 8.4%. This value is measured with a laptop with total AH capacity of 51190 mW. In other word, by selecting level 4, the user can use the tele-rehabilitation service for 19 more minutes.

5.5 Summary

One of the key issues in building a tele-rehabilitation system based on mobile cloud computing (MCC) is how to minimise the energy expenditure of mobile devices to extend the duration of accessing tele-rehabilitation services. In this study, the characteristics of bio-kinematic signals were taken into consideration to achieve this goal. More specifically, a multi-level encoding scheme is introduced to encode human motion trajectories into various levels according to the computational power and the Internet speed of the mobile device. The simulation and real-data experiment have confirmed the effectiveness of the proposed scheme in saving power of mobile devices.

Apart from that, an architecture of MCC-based tele-rehabilitation system was introduced, together with analysis-oriented decision support system and security service layer to provide advanced and secured data communication and analysis.

As future work, focus should be put on the development of suitable algorithms for AODSS for various conditions.
Chapter 6

Conclusion

Tele-rehabilitation has been studied in a large number of research publications in the past few decades. In recent years, with the development of information and communication technologies, tele-rehabilitation is being developed rapidly for a few reasons, such as the aging population, limited number of medical resources and significant amount of time and financial cost spent on providing and accessing rehabilitation services.

However, there remain challenges in implementing effective tele-rehabilitation services. For example, one challenge is to provide affordable but robust and accurate devices or systems for patients to use remotely to collect information regarding their body movements. Additionally, how to represent human motions in an effective way that facilitate further analysis in tele-rehabilitation area is also an open question. Thirdly, in tele-rehabilitation, therapists are not always available. In other words, the performance of patients in tele-rehabilitation sessions cannot be assessed in time. Therefore, one question here is how to develop methods to automatically perform certain evaluations regarding to the performance of the patients. The result may be useful as the feedback to patients and supportive information for therapists for further analysis.

In this dissertation, one of the most critical elements in a tele-rehabilitation system for automated kinematic performance assessment, namely the OBMCDs,
was first investigated. Multiple affordable OBMCDs, such as Kinects, were used to build a system that balanced the cost and accuracy of the complete system. To achieve this goal, robust linear filtering was used to fuse information (joint positions) from a number of OBMCDs. By doing so, the tracking accuracy could be improved significantly and the field of view could be extended to a certain degree. More importantly, the occlusion problem in the majority of OBMCDs could be eliminated to a large extent. However, due to the fact that the increasing number of OBMCDs inevitably increases the cost and the management complexity, it is critical to select the number of devices in a system to balance all the factors. For some clinics with limited budget but trying to improve the tracking accuracy with OBMCDs for patients’ kinematic performance assessment, it is recommended to use two OBMCDs for maximising the improvement percentage with minimum extra investment and management difficulty. According to the experiment result shown in Fig. 2.9, it is obvious that when the number of OBMCDs increases from one to two, the relative improvement percentage can be around 15%. However, when the number increases from two to three, the relative improvement percentage drops significantly. As a result, using two OBMCDs in this scenario offers the best balance between tracking accuracy and financial expenditure.

With a robust and accurate optoelectronic bio-kinematic motion capture system (OBMCS), patients’ motions can be recorded automatically and remotely, which is the first step of building an automated assessment system of human kinematic performance. After that, it is critical to extract features that can meaningfully represent human motions in the field of rehabilitation for further analysis. Therefore, a two-component encoding model was proposed to encode human motions in the joint (sensor) level. Indeed, spatial indexing scheme was proposed to encode the trajectories and then noise in trajectories was eliminated by using SG filter. Their performance has been confirmed in various experiments. Furthermore, continuous
hidden Markov model was used to decompose a complex trajectory into the combination of a number of pre-defined atomic motion trajectories with high accuracy. As a result, a complex motion trajectory can be further analysed by looking into its smaller components.

By using the spatial characteristic of the shape model, which is that it is highly sensitive to noise and able to amplify noise, a novel automated approach was developed to evaluate the ability of a person to perform upper extremity reaching tasks in daily living. As a result, even though there is no therapist around, patients’ performance can still be assessed to a certain extent. Simultaneously, therapists are able to take the assessment result into consideration when they make further rehabilitation plan for the patients. Here, the proposed method was designed based on the fact that dyskinesia patients are highly like to have a large number of involuntary movements, and their motions are usually slower than healthy people. After encoding their motion trajectories with a shape and a dynamic model, the entropy of these models is computed to illustrate the smoothness of motion trajectories. Eventually, the entropy and the duration for the motion were used to classify the motion trajectories into one of the pre-defined levels, corresponding to the ability levels to perform certain tasks. The classification results of this automated approach highly agreed with that of a human observer.

Last but not the least, a mobile cloud computing (MCC) based architecture was proposed to combine the techniques introduced in this dissertation to build a prototype of a tele-rehabilitation system so that patients are able to access the tele-rehabilitation service with their mobile devices regardless of their location and the time. In this investigation, a multi-level encoding scheme was utilised to minimise the energy expenditure of mobile devices by encoding motion trajectories into various levels. As a result, by selecting a suitable encoding method, patients are able to maximise the duration to access tele-rehabilitation services.
However, the proposed OBMCD-based assessment system has some technical and practical limitations in a real-world clinical environment. Firstly, to use this system, parameters for the fusion algorithm should be well tuned to gain the best performance. However, due to various environment conditions, such as light, these parameters may not be the same in different clinics. As a result, it may take some time to tune the parameters, which will be difficult for clinicians to do it. Secondly, as mentioned in Chapter 2, the locations and orientations of the OBMCDs are select empirically to minimise the occlusions, experiences should be accumulated for a period of time to select the optimal poses of OBMCDs. Thirdly, since the proposed OBMCS measures the position of joints to represent the movements, it is not able to measure some exercises, such as wrist pronation and supination. In light of the above, the proposed OBMCD-based system needs to be improved in the future.

Although a number of contributions have been made in this dissertation to automate the assessment of human kinematic performance for tele-rehabilitation, it is not the end of the study. As future work, three main problems should be solved before implementing an effective tele-rehabilitation system. On one hand, since bio-feedback is vital to improve the effectiveness of performing tele-rehabilitation, how to intelligently control the delivery of various forms of bio-feedback is essential in a tele-rehabilitation system. To achieve this, technologies developed in artificial intelligent field, such as machine learning, could be candidates. On the other hand, robust and accurate automated assessments should be developed for various conditions to overcome the reducing ratio between the number of therapists and patients. Thirdly, a sophisticated data analysis system should be investigated to establish a decision support system so that therapists can be benefited from using the tele-rehabilitation system to receive supportive information for making further decisions.
This dissertation has shown the promise of establishing an automated assessment for human kinematic performance in tele-rehabilitation. More refinements should be conducted between engineers and therapists to achieve better outcomes for patients and in general for the health care system.
List of Figures

1.1 A depiction of the passive and active locomotor system .......................... 2
1.2 The functions associated to certain areas of the brain. ............................. 3
1.3 Appearance and components of Kinect version 1[70] ................................. 11
1.4 The pinhole camera model of Kinect version 1[71]. .................................. 12
1.5 An example of projected pattern of bright spots on an object [72] . ............... 12
1.6 Appearance of Kinect version 2[76] ....................................................... 13
1.7 The patient side of the POTS-based tele-rehabilitation system. [87] ............. 16
1.8 The physiotherapist monitoring the exercise performance of his patient remotely through the Internet. [90] ......................................................... 18
1.9 Pictures of animals ................................................................................. 18
1.10 Locations of five sensor worn by a subject [102] ....................................... 20
1.11 The illustration of two bilateral electrogoniometers in a home tele-rehabilitation system ................................................................. 24
1.12 Marker-based hand tacking system [123] ................................................. 26
1.13 The process of predicting clinical scores from 14 features. ......................... 29

2.1 Virtual human mimicking the movements of real human with data captured by two Kinects© ................................................................. 35
2.2 Uncertainty ellipse in RLF estimation .......................................................... 47
2.3 Fig. 2.3(a) illustrates four instances where the two Kinect system may have missing data and another instance without any occlusion. Here, the *incomplete* matrix parameters of the RLF based two Kinect fusion are explicitly stated. Fig. 2.3(b) shows the optimal positioning of two Kinects. ........................................... 51

2.4 Information theoretic assessment of Kinect\(^\circ\) orientation .......... 52

2.5 Filter performance improvement against multiple Kinects\(^\circ\) subjected to different uncertainty levels ............................................. 56

2.6 The relationship between \(\Psi(M), \alpha\) and \(M\) in the cases with and without occlusion. Robust linear filtering was utilised for data fusion. 57

2.7 Experimental setup: VICON and Multi-Kinect\(^\circ\) system .......... 58

2.8 Averaged RMSEs \((\hat{\phi}_M^a, M = 2, 3)\) over the same type of exercises conducted by the three subjects ................................. 60

2.9 Averaged relative improvement percentages \((\hat{\phi}_{M-1:M}^a, M = 2, 3)\) of multi-Kinect\(^\circ\) fusion ............................................. 60

2.10 Errors of two Kinect\(^\circ\) fusion with missing data. The temporal interval between consecutive samples are 1/240 second. ................. 62

2.11 Cloud-based exercise monitoring and performance assessment with bio-feedback ............................................. 66

2.12 Average RMSE and performance measure of two Kinect fusion with missing data. ............................................. 67

3.1 Three-level syntactic description framework for building language for human action ............................................. 71

3.2 Shape model. Here the locus of points in the 2D \(\kappa-\tau\) space corresponds to a 3D trajectory positions .................... 77

3.3 Dynamic model. Here the speed along the trajectory, \(v\), is indexed over time, \(t\). ............................................. 77
3.4 Three motions used in experiments of this work. Fig.3.4(a) and 3.4(b) are for linear motion. Fig. 3.4(c) and 3.4(d) are for planar motion. Fig. 3.4(e) and 3.4(f) are for helical motion. ...... 86

3.5 Two helical trajectories with the same orientation and shape, but two different dynamics. The colour of the trajectory in Fig. 3.5(a) with constant speed (roughly 0.1 m/s) changes from black to grey with time, and the different line styles and colours of the trajectory in Fig. 3.5(b) indicates the change of velocities. The legend on the right side of Fig. 3.5(b) shows the approximate speed of different parts of the trajectory in Fig. 3.5(b) .......... 89

3.6 Shape models: Fig. 3.6(a) for trajectory in Fig. 3.5(a); Fig. 3.6(b) for trajectory in Fig. 3.5(b) ...................... 90

3.7 Dynamic models: Fig. 3.7(a) for trajectory in Fig. 3.5(a); Fig. 3.7(b) for trajectory in Fig. 3.5(b) ................... 90

4.1 Examples of commonly used techniques with features considered for ADL performance measurement. Feature-based performance evaluation (FPE) is primarily based on kinematic or kinetic measurements. Dynamic measurements such as number of velocity peaks [228], number of zero-crossing tangential accelerations [229], dimensionless jerk metrics [223] and spectral arc-length [226] are used in techniques such as VP, ZCA, DJ and SAL respectively in the literature. Our proposed entropy of shape and instantaneous acceleration (ESA) introduced in this chapter uses both shape and kinematic based measurements. Kinetic-related features, such as moment [230], force [231] and torque [232] have also been investigated. .................. 101

4.2 The smoothness level of simulated trajectories .................. 108

4.3 A diagrammatic view of the experimental setup. .................. 109
4.4 Real data experiment setup image. The top image shows the setup of the Kinect and the subject. The bottom left and right are the RGB and depth images taken from the Kinect. Note that the marker was on the right wrist of the subject (the depth and RGB camera in the Kinect reversed the image.) ........................................ 110

4.5 These three graphs show trajectories in three levels. The left one is in level one for natural movements without involuntary movements and jerks. The middle trajectory is in the second level with two involuntary movements (with amplitudes of 0.225 and 0.25 meter) and jerks (with amplitudes of 0.025 and 0.0275 meter). The last one is in the third level with four involuntary movements (with amplitudes of 0.45, 0.5, 0.55 and 0.6 meter) and jerks (with amplitude of 0.05, 0.055, 0.06 and 0.065 meter). The red circles are examples of jerks in the second and third level. ........................................ 113

4.6 The metric given by these approaches tends to illustrate the consistency characteristics in various approaches used to evaluate the smoothness of trajectories in two severity levels. With the increase of numbers and amplitudes of both involuntary movements and jerks, the smoothness of the trajectories deteriorates. The first half (with blue from 1 to 25) is in the second level and the last half (with red from 26 to 50) is in the third level. The VP was computed with a threshold value of 0.01 meter/s and the temporal gap between two consecutive peaks was 100ms. ..................... 114
4.7 Sensitivity comparison of the five approaches with respect to the change in the severity of involuntary movements ($\frac{1}{SNR}$). A better evaluation technique is preferred to be sensitive to the small changes in the severity of involuntary movements and the change rate metric should be proportional to the change rate of severity. The blue and red lines show the improvement of metrics of the second and third severity levels with respect to the metric of the first trajectory computed with various approaches (refer to (4.4.1). The orange lines are the regression lines of the corresponding metrics, showing the approximate improvement rate of each method. The value at the right bottom of each graph is the gradient of the regression line.

4.8 Robustness comparison of the five approaches with respect to the change in the severity of involuntary movements. A better evaluation technique needs to be limited to a smaller range (on the horizontal axis) in terms of the differences in metrics between the noisy trajectories and their corresponding noiseless ones, and the value with the highest amplitude in vertical axis should be close to zero.

4.9 Examples of trajectories (first three rows), shape models, including curvatures (fourth row) and torsions (fifth row), and instantaneous accelerations (sixth row) are illustrated for three levels of the ability to perform an upper extremity reaching task (columns one to three corresponding to levels one to three of the severity of involuntary movements). The red circles show examples of sub-movements and green rectangles are examples of jerks.
4.10 The distributions of durations utilised to finish the reaching task, as well as metrics computed by five approaches, for three severity levels. The threshold utilised to compute VP was 0.25 m/s and the temporal gap was 100 ms for all subjects. .......................... 121

5.1 Multi-level exercise data encoding scheme .......................... 128

5.2 An illustration of the elbow point concept. Red points are elbow points when $\kappa > 0.05$. Blue points with curvature less than 0.05 is removed from the trajectory. .......................... 130

5.3 The overview of the tele-rehabilitation system .................. 135

5.4 Data flow model in therapist side (except for on-line review) .... 136

5.5 Example of making model motions of the therapist side with Kinect. 137

5.6 Data flow model in patient side (with on-line review for therapists) 138

5.7 Example of the patient side. .......................... 140

5.8 The conceptual Architecture for analysis oriented decision support system. .......................... 141

5.9 Numerical experimental result .......................... 145

5.10 Setup of the real-data experiment .......................... 146

5.11 Result of real-data experiment for the multi-level encoding scheme. 147
List of Tables

1.1 Examples of musculoskeletal injuries in joints of upper extremities. “Inj” refers to injury and “ST” refers to studies using physical rehabilitation to treat the conditions or gain recovery (the same as that in Table 1.2). ..................................................... 7

1.2 Examples of musculoskeletal injuries in joints of lower extremities . 8

1.3 Comparison of basic technical specifications between two versions of Kinects .............................. 14

2.1 Example human movements ................................. 59

2.2 Comparison (in the presence of missing information) of Accuracy and execution times between the proposed RLF and the ones introduced in the references [186] [187] and [188]. .......................... 63

3.1 Sensor motion types ......................................... 84

3.2 Comparison between SG and LS. .......................... 85

3.3 Shape and dynamics for four motions. In each left figure, the colour changes from black to grey with time. ...................... 87

3.4 Parameters used to generate the standard deviation for filters processing trajectories with temporal (first four rows) and spatial (last four rows) indexing scheme. Here, L and P are for the linear and planar motion respectively. (the same in the rest tables). .......... 91
3.5 The comparison of curvatures and torsions for four trajectories indexed by temporal and spatial indexing schemes and filtered by SG, LS and KF. Here C is for curvature and T is for torsion. 92

3.6 Trajectories and decomposition of complex motions. In each top figure, the colour changes from black to grey with time in the trajectories. 94

4.1 The definition of three kinematic severity levels of involuntary movements and jerks, as well as their corresponding abilities in performing reaching tasks in daily living. 102

4.2 Parameters used to simulate two groups of trajectories. These two groups of trajectories correspond to two severity levels of involuntary movements. The first 25 trajectories belong to the second level with two involuntary movements and two jerks. The last 25 trajectories are in the third level with four involuntary movements and jerks. To simulate the severity in one level, the amplitudes of involuntary movements and jerks increase with the index. 107

4.3 Demographic data of subjects 111

4.4 Cohen’s Kappa ($p < 0.05$) between various automated approaches and the human observer. 120

5.1 Notations used in the process of estimating power consumption for data encoding and transfer. Here $i = 1, 2, 3, 4$ for motion tele-rehabilitation and $i = 1, 2, 3$ for respiratory tele-rehabilitation. 131

5.2 Power consumption of various encoding scheme and encoded data transfer for motion tele-rehabilitation. 133
Bibliography


Bibliography


M. Alt Murphy, C. Willen, and K. S. Sunnerhagen, “Responsiveness of upper extremity kinematic measures and clinical improvement during the first three months after stroke,” *Neurorehabil Neural Repair*, vol. 27, no. 9, pp. 844–53, 2013.


