Human Skill Capturing and Modelling using Wearable Devices

By

Yuchen Zhao

A Doctoral Thesis

Submitted in partial fulfillment of the requirement for the award of

Doctor of Philosophy of Loughborough University

April 2017

Acknowledgement

Foremost, I would like to express my sincere gratitude to my supervisor Dr. Yee Mey Goh for the continuous support of my Ph.D and related research, for her patience, enlightment and knowledge. She opened the door for me to fulfil my enthusiasm about the research object. Her guidance helped in all the time of the research and writing of this thesis.

Beside my supervisor, I would like to thank Dr. Niels Lohse, Dr. Laura Justham and Prof. Mike Jackson, for their encouragement, insightful comments and hard questions which inspired me to widen my research from various perspectives.

I thank my fellow labmates for all the fun we have had in the last four years. In particular, I am grateful to Ali for the stimulating discussions and for the days that we are working together before deadlines. I enjoyed the time that we were helping and teaching each other in all aspects of the research.

My sincere thanks also go to all my friends and family members for the generous help and the fun at all time. In particular, I would like to thank and hug my wife Dixi Fu, for her constant companion and support without reservation.

Last but not the least, I would like to thank my parents Ying Li and Zhi Zhao, for giving birth to me at the first place and supporting me spiritually throughout my life.

List of Publications

[1]. Y.C. Zhao, A. Al-Yacoub, Y.M. Goh, L. Justham, N. Lohse, M.R. Jackson, "Human Assembly Skill Capture A Hidden Markov Model Analysis of Force and Torque Data in Peg-in-Hole Assembly," IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, 2016.

[2]. Y.C. Zhao, T. Johnson, Y.M. Goh, L. Justham, N. Lohse, M.R. Jackson," A Sensor Design and Data Analysis for Automatic Drum Beater Winding (2014)", International Conference on Engineering Design (ICED), 2014.

[3]. Y.C. Zhao, Y.M. Goh, N. Lohse, L. Justham, M.R. Jackson, "A Robust Hybrid VICON and IMU System for Tracking Human Forearm Motions," submitted to IEEE Transactions on Sensors, under review.

[4]. Y.C. Zhao, A. Al-Yacoub, Y. M. Goh, L. Justham, N. Lohse and M. Jackson, "Surface EMG-based Force Torque Prediction in a Peg-in-hole Assembly Context for Human Tacit Knowledge Interpretation," submitted to IEEE Transactions on Systems, Man, and Cybernetics, under review.

List of Acronyms

1D	One-dimensional
2D	Two-dimensional
3D	Three-dimensional
BIC	Bayesian Information Criterion
CMM	Coordinate Measuring Machine
CAD	Computer Aided Design
DOF	Degree-of-Freedom
DWT	Dynamic Time Warping
DMP	Dynamic Movement Primitives
F/T	Force Torque Sensor
FIR	Finite Impulse Filter
GMM	Gaussian Mixture Model
GMR	Gaussian Mixture Regression
HMM	Hidden Markov Model
IMU	Inertial Measurement Units
IIR	Infinite Impulse Filter
ICA	Independent Component Analysis
LfD	Learning from Demonstration
LWR	Locally Weighted Regression
LWPR	Locally Weighted Projection Regression
Mocap	Motion Capturing
MSE	Mean Square Error
MTU	Musculotendon Units
PiH	Peg-in-hole
PCA	Principle Component Analysis
PLI	Power Line Interference
R	Regression coefficient
sEMG	surface Electromyography
SVM	Support Vector Machine
SME	Small and Medium-sized Enterprises
TDNN	Time Delayed Neural Network

List of Symbols and Definitions

Macroscopic variables		**Usage(s)**
N	Number of samples	Chapter 3
R	Homogeneous rotation matrix	Chapter 3
q	Quaternion	Chapter 3
M	Cumulated matrix	Chapter 3
t	Time stamp	Chapter 3
a	Scaling factor	Chapter 4
b	Time scaling factor	Chapter 4
$\boldsymbol{\Phi}$	Basis function	Chapter 4
i, j	Dynamic time warping indices	Chapter 4
I	Identity matrix	Chapter 4
$\boldsymbol{\sigma}$	Standard deviation	Chapter 4
Ψ	Uncertainty in the probabilistic model	Chapter 4
$\boldsymbol{\pi_k}$	Prior probability of the *k th* Gaussian component	Chapter 4, 6
$\vec{\boldsymbol{\mu}}_{\boldsymbol{k}}$	Mean of the *k th* multivariate Gaussian	Chapter 4, 6
$\boldsymbol{\Sigma_k}$	Covariance of the *k th* multivariate Gaussian	Chapter 4,6
$\boldsymbol{f_{pca}^1}$	1st PCA of the force vector	Chapter 4,5,6
$\boldsymbol{h}$	Hidden component	Chapter 4
$\boldsymbol{\theta}$	Model parameters	Chapter 4
$\boldsymbol{S_{BIC}}$	Value of Bayesian Information	Chapter 4
n	Number of free parameters	Chapter 4
$\boldsymbol{\Pi}$	Prior state's distribution	Chapter 4,6
$\boldsymbol{T}$	Transition probability	Chapter 4,6
$\boldsymbol{E}$	Emission probability	Chapter 4,6
$E[*]$	Expectation	Chapter 4
$\boldsymbol{\beta_k}$	Mixture weights of the *k th* components	Chapter 4,6
$\vec{\boldsymbol{x}}^{\boldsymbol{d}}$	Desired retrieved trajectory	Chapter 4
$\boldsymbol{\xi^d}$	Generic desired retrieved trajectory	Chapter 4, 6
$\boldsymbol{\Sigma_{\xi\xi}^d}$	Generic desired retried uncertainties	Chapter 4, 6
$\boldsymbol{\mathcal{N}}(*)$	Normal Gaussian distribution	Chapter 4, 6
$\boldsymbol{\alpha_{i,t}}$	Forward variable	Chapter 4, 6
$\boldsymbol{q_d}$	Desired angle	Chapter 4,6
$\boldsymbol{\dot{q}_d}$	Desired angular velocity	Chapter 4,6
$\boldsymbol{q}$	Angle	Chapter 4,5,6
$\boldsymbol{\dot{q}}$	Angular velocity	Chapter 4,6
$\boldsymbol{\ddot{q}}$	Angular acceleration	Chapter 4,6
$\boldsymbol{k_v}$, $\boldsymbol{k_p}$	PD control gain	Chapter 4,6
$\boldsymbol{f^{F/T}}$	Composed raw force data	Chapter 5
emg	Raw sEMG signal	Chapter 5,6
db	Daubechies basis wavelet	Chapter 5, 6
$\boldsymbol{N_d}$	Number of time delays	Chapter 5, 6
$\boldsymbol{N_h}$	Number of hidden units	Chapter 5, 6
$\boldsymbol{D_l}$	Wavelet decomposition level	Chapter 5, 6
F1,F2,F3,F4	Decomposed force	Chapter 6
$\boldsymbol{f_{raw}}$	Raw force measurement from TMU	Chapter 6

TABLE OF CONTENTS

List of Figures

List of Tables

Abstract

Industrial robots are delivering more and more manipulation services in manufacturing. However, when the task is complex, it is difficult to programme a robot to fulfil all the requirements because even a relatively simple task such as a peg-in-hole insertion contains many uncertainties, e.g. clearance, initial grasping position and insertion path. Humans, on the other hand, can deal with these variations using their vision and haptic feedback. Although humans can adapt to uncertainties easily, most of the time, the skilled based performances that relate to their tacit knowledge cannot be easily articulated. Even though the automation solution may not fully imitate human motion since some of them are not necessary, it would be useful if the skill based performance from a human could be firstly interpreted and modelled, which will then allow it to be transferred to the robot.

This thesis aims to reduce robot programming efforts significantly by developing a methodology to capture, model and transfer the manual manufacturing skills from a human demonstrator to the robot. Recently, Learning from Demonstration (LfD) is gaining interest as a framework to transfer skills from human teacher to robot using probability encoding approaches to model observations and state transition uncertainties. In close or actual contact manipulation tasks, it is difficult to reliabley record the state-action examples without interfering with the human senses and activities. Therefore, wearable sensors are investigated as a promising device to record the state-action examples without restricting the human experts during the skilled execution of their tasks.

Firstly to track human motions accurately and reliably in a defined 3-dimensional workspace, a hybrid system of Vicon and IMUs is proposed to compensate for the known limitations of the individual system. The data fusion method was able to overcome occlusion and frame flipping problems in the two camera Vicon setup and the drifting problem associated with the IMUs. The results indicated that occlusion and frame flipping problems associated with Vicon can be mitigated by using the IMU measurements. Furthermore, the proposed method improves the Mean Square Error (*MSE*) tracking accuracy range from 0.8˚ to 6.4˚ compared with the IMU only method.

Secondly, to record haptic feedback from a teacher without physically obstructing their interactions with the workpiece, wearable surface electromyography (sEMG) armbands were used as an indirect method to indicate contact feedback during manual manipulations. A muscle-force model using a Time Delayed Neural Network (TDNN) was built to map the sEMG signals to the known contact force. The results indicated that the model was capable of estimating the force from the sEMG armbands in the applications of interest, namely in peg-in-hole and beater winding tasks, with *MSE* of 2.75N and 0.18N respectively.

Finally, given the force estimation and the motion trajectories, a Hidden Markov Model (HMM) based approach was utilised as a state recognition method to encode and generalise the spatial and temporal information of the skilled executions. This method would allow a more representative control policy to be derived. A modified Gaussian Mixture Regression (GMR) method was then applied to enable motions reproduction by using the learned state-action policy. To simplify the validation procedure, instead of using the robot, additional demonstrations from the teacher were used to verify the reproduction performance of the policy, by assuming human teacher and robot learner are physical identical systems. The results confirmed the generalisation capability of the HMM model across a number of demonstrations from different subjects; and the reproduced motions from GMR were acceptable in these additional tests.

The proposed methodology provides a framework for producing a state-action model from skilled demonstrations that can be translated into robot kinematics and joint states for the robot to execute. The implication to industry is reduced efforts and time in programming the robots for applications where human skilled performances are required to cope robustly with various uncertainties during tasks execution.

Keywords: Manufacturing automation, Force based control, Motion Capturing (Mocap), Learning from Demonstration (LfD), surface Electromyography (sEMG).

Chapter 1. Introduction

1.1 Background and motivation

Robots are an important resource in many industrial applications. They have replaced numerous dull, repetitive, dirty and dangerous manual manipulations. Robots have been implemented in manufacturing processes such as vehicle assembly [1][2] and polishing [3]. These task-specific processes are typically programmed off-line and manually by an engineer with the assistance of the Computer Aided Design (CAD) [4]. Due to the general skill shortage in manufacturing industry [5] and desire to increase productivity, robots that can augment and replicate human skills in more challenging environments are needed [5]. This means that the robots have to be upskilled to deal with more complex tasks such as performin delicate tasks (such as handling a soft component), higher-level task (such as making sequential decisions within a context), and interactive task with human operator (such as collaborative manipulations). To fulfil these increasing need of robotic applications, a new generation of the robot should be more flexible and human-driven designed [6].

The cost of deploying an automation system can be split into 20% to 25% for the robot, 20% to 30% auxiliary hardware, and 45% to 60% system integration [7]. A significant investment in effort and capitals are required to integrate the robots and sensors into the manufacutring workflow, however the resuse of the software from one application to another is very limited [7]. In general, the cost of installation and the supporting hardware/software infrastructures in a cell is normally up to 10 times of the cost of the robots. Therefore, an automation solution is more cost effective if the robot cell and program can be reusable. Programming a robot could take a long time and require programming skill. For large industries such as automotive and aerospace, the programs are required to be more adaptive to deal with complex task and increasing autonomy. For SMEs who have not adopted robotics in manufacturing, the reusability of the program is becoming even more important because the installation and infrastructure costs are unlikely to drop in the near future [7]. The increasing complexity of the manipulation tasks is soon making the automation solutions even harder to achieve by using the conventional robot programming method.

When a human expert is facing complex tasks, most of the time, the decisions are derived from their tacit knowledge. In contrast to rule-based performance, tacit knowledge is difficult to articulate and difficult to model [8] but can be acquired through training. Humans are adaptive to the variations by using predominantly visual and haptic feedback, even without knowing explicitly the exact mathematical or physical problems. For instance, a skilled operator can pick up a workpiece of any shape from anywhere on the table and assemble it in the right place with appropriate force. The skills are embedded in the *control policy which mapes the current state to the actions*. For example, in the context of assembly task, the control policy maps the current reaction force and positions to the corrective motions. The capability to model such tacit control policies are useful to automate the process.

Robot learning from demonstration (LfD) is such an approach to learn the control policies without too much programming effort. The concept is similar to an experienced worker trying to teach an apprentice. The experts will be demonstrating the task rather than explicitly explaining the rules behind their low level actions. The precise mathematical formulations of the task can be avoided using LfD allowing non-expert robot programmers to transfer their domain knowledge to the robot. A frequently used LfD framework is shown in Figure 1-1. In general, LfD contains three stages: demonstration, model, and reproduction. The goal of LfD is to have a robot 'watch' a teacher's demonstrations of the task to be performed [9]; The robot does not simply replicate one of the demonstrations rather a state-action policy is learned and computed by optimising the reward function from a small number of demonstrations with variations. The LfD approach has been studied and applied on various robotic problems such as Peg-in-Hole (PiH) [10], Pick and Place [11] and ball sorting [12] which are closely related to the daily life manipulations.

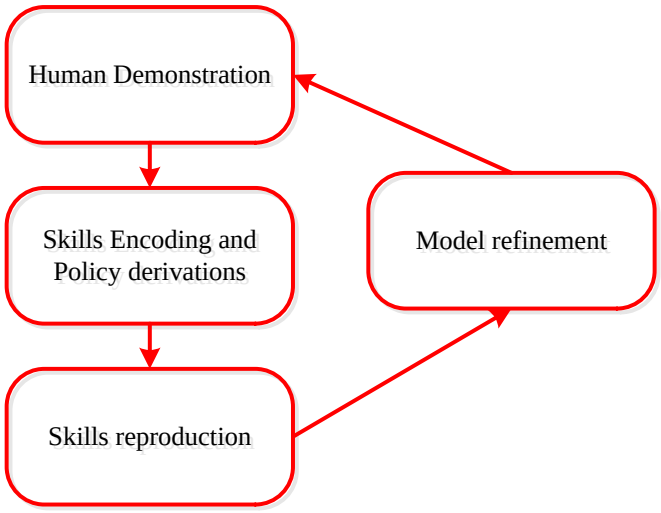


Figure 1-1 The framework of transferring human skills to the robot learner via demonstration

In order to model the control policies, the human motions and haptic feedback need to be recorded. There are various technological solutions available in the market for human motion tracking performance. Wearable sensors [13] have gained the interest of researchers and developers because they are portable and non-intrusive to the natural motions. Among all the choices, photometry and Inertial Measurement Units (IMUs) based techniques are dominants for human motion tracking. The photometry technique relies on the cameras to record people's movements and analyse them off-line. The current technology enables pose estimations of objects with the help of 3D camera or placing reflective markers on the subject's body. Different from the camera-based approach, the IMUs does not need the line-of-sight and use freely in the space for body tracking. Some active application areas are rehabilitation [13], teleoperations [14] and entertainments [15].

The LfD approach has been largely used for learning human kinematics, however, most contact-based manipulation problems are relying on force/haptic feedback. Therefore, it is more important to develop force based control policies for applications that require skills performance from the workers. To generate accurate haptic feedback, force-toque (F/T) sensing or tactile sensing devices are ideal choices. In manufacturing, the F/T sensor attached to the robot end effector generates force feedback to the robots. However, it would be impractical to mount an F/T sensor on the human operator whilst they are carrying out skilled tasks. Therefore, tactile gloves are often used to generate force feedback directly[16]. However, the touch feedback is still compromised by using the glove. As an alternative, an indirect method is to use the measurement of the muscle

activations to indicate the level of the contact force. Wearable surface Electromyography (sEMG) is the device to measure the electrical activity of the muscle fibers during a contraction [16]. It has been used for gesture recognition [17], force [18] and pose estimations [19] and controlling prosthesis [20].

Based on the above context, this thesis aims to capture, model and transfer the manual manufacturing skills from a human demonstrator to the robot to reduce the associated programming efforts using the LfD framework. A hypothesis in this thesis is the wearable sensors for both human motion tracking and haptic feedback can be used to capture the skills of an operator and provide a model of the state-action policies required to control the robot.

1.2 Problem statement

To date, an accurate wearable human motion tracking system is usually based on photogrammetry. For example, a multi-cameras Vicon system [21] can be used to track objects with multiple reflective markers attached. To track an object in three dimensional space, the minimal requirement is two cameras; the accuracy can be improved with more cameras but the setup will be expensive. Although the accuracy might be enough for human motion tracking, occlusion is a problem when line-of-sight is unavailable. Inertial Measurement Units (IMUs) are promising alternative in free space motion tracking but they suffer from the drifting problem due to magnetic distortion [14].

The human motion tracking systems are more responsible to the gross motions and the kinematic based control. But the fine motions tend to rely on additional tactile feedback, which are difficult to measure. The F/T sensor is widely used for capturing haptic feedbacks, but its measurements are not easy to collect due to the sensor size and physical restriction of the fixtures, which often obstruct the physical interactions between the operator and the workpiece. One potential solution is to install a static F/T sensor however, if the application changes, F/T sensor has to be retrofitted into a different position. Therefore, this restricts the implementation of the LfD approach on industrial applications that require accurate and intricate manipulations. As an alternative, indirect measurements of haptic feedback using muscle activations can be potentially useful. However, muscle-force model is subject specific and can be challenging to build [22]. Supported by the study of anatomy, the researchers in

biomechanics have provided explicit muscle-force model using simulation approaches [23]. However, this method requires a long time to calibrate the individual subject's model from the standard model and the setting up phase usually takes a long time. A data-driven approach might be more appropriate, but tuning of the non-parametric muscle-force model could be challenging.

After gathering the motion and haptic data, the signals have to be decomposed to extract essential features of the given tasks especially those related to the tacit knowledge that are performed without conscious control. Researchers who are focusing on building explicit control policies using parametric models derived from physical or mathematical formulations face the problem of uncertainties, which is exacerbated when the task is more complicated with many states and actions. To this end, the robot learning from demonstration approach tries to address uncertainties by constraining and modeling them through a small number of human demonstrations. The approach assumes that the meaningful features from the experts are encoded in the state-action policies since not all the teacher's executions contribute to optimal solutions of the task. The challenges relate to how to derive control policies with generalisation capability and how suitable it is for the robot to reproduce the required performance.

One of the common ways of demonstrating a task to a robot is through kinaesthetic teaching (i.e., human teaches the robot by holding the end-effector and all the data are recorded from the sensors on the robot). However this approach limits the ability to represent the underlying human skills reliably. Therefore, as oppose to this teaching method, the sensor-on-teacher approach has been chosen in this thesis because this approach allows the operator to demonstrate fine manipulation skills naturally during the demonstrations. The wearable systems used in this thesis become essential in this context. After demonstration step, extra efforts are needed to map the state-action policy from the teacher to the student [24]. Once the state-action policy has been built based on human skilled executions, the challenge relates to the transferring and deployment on the robot due to the correspondence issue in between the teacher and learner [25].

Among all the mentioned challenges, the scope of this research is framed as the following: i) build a reliable wearable system suitable for tracking human operators during manual manipulation task demonstrations that require both haptic and pose

feedbacks; ii) verify the skill model reproducibility against unseen human demonstration episodes; iii) evaluate the robustness of the skill transfer framework on a different case study. The correspondence issue is outside the scope of the study, since the model is validated from the unseen human dataset. Four challenges to be addressed in this thesis are described in the following sub-sections.

1.2.1 Reliable tracking of human motion with minimal interference

Various wearable techniques have been proposed in the literature to enable human motion tracking. To achieve the aim of this research, the teacher should deliver his/her skill as reliable as possible with minimal interference. The user wearing mechanical trackers made from rigid or flexible goniometers are therefore considered inappropriate. These trackers directly measure the joint angles of the wearer. However, the installation of these body-based linkages is not trivial and requires extra efforts from the operator to get used to them. It is difficult to track the full body motions in multiple degrees of freedom as a result of these constraints. Additionally, the fixture may obstruct user comfort thus, the data may not be representative of their actual motions. Vision based tracking removes those fixture and allows the subject to move in three dimensions without constraints. However, vision systems may lose tracking when the objects are occluded or partially occluded, for instance, when the operator moved his/her hand inside a hole, which makes the markers invisible from the camera. To date, miniature IMUs with embedded accelerometers, gyroscope and magnetometer are available, but they suffer from drifting problem and unreliable for long time usage. From the above, the tracking system needs to robustly record the human motion with reliable measurements and minimal interference, but the current technologies have their disadvantages as stated above.

1.2.2 Skill capturing, encoding and generalisation

The quality of LfD approach highly depends on the information provided by the demonstration dataset. This means a poorly composed dataset leads to a poorer learner performance. Ideally, a teacher should demonstrate optimal skills to the robot. However, the reality is teachers' executions might be suboptimal [26] which means the sensing of the state and its corresponding action might not be sufficient or necessary for the robot to learn. One potential solution is to remove the irrelevant or unnecessary actions that do not contribute to the task. Another solution is to smooth and generalise the suboptimal solutions by multiple demonstrations or multiple teachers [24-25]. In

summary, the challenge in skills capturing, encoding and generalisation are the difficulties in deriving suitable and reusable policy due to the problems stated above.

1.2.3 Muscle-force model based on sEMG measurement

To generate haptic feedback without installing F/T sensors, wearable sEMG sensors are potentially useful [29]. sEMG driven muscle-force model has been developed for lower limbs [30] and upper limbs [31]. But most of the researchers are concerned with one degree of freedom movement [32][33]. This is because multiple degrees of motions make the modeling difficult. Normally, the muscle-force model is stationary and can be modeled using one Gaussian distribution within a short interval of isometric contraction. However, when the person moves to a different pose, muscle's behavior changes [34]. Therefore, to collect haptic feedback from human, a dynamic muscle-force model with more than one degree of freedom would be required to represent complex manipulations.

1.2.4 Evaluation of the motion productions based on the learned state-action policy

Once the state-action policy model is available, the robot could reproduce the motions. The reproduced motion is the action after the learner's perception of its state. Well-established policy helps the learner smoothly and accurately switching control strategies when the state transition happens. However, suboptimal solutions degrade the learner's reproduction performance. In this thesis, applying the control policy onto the actual robot learner is beyond the scope of the study, therefore, regardless the correspondence problem; the challenge is to evaluate the performance of the policy from human demonstrations before transferring the model to a robot learner.

1.3 Research aim and objectives

This thesis aims to address the knowledge gaps with regards to capturing, modeling and transferring the manual manufacturing skills from a human demonstrator to the robot with the help of wearable sensors. The application domains are manual manipulations using forearm. To pursue this aim the following objectives are identified:

1.3.1 To develop a wearable system that reliably track human motions

To develop a hybrid system that reliably tracks the human body segments using photometrical and inertial-based wearable sensors with minimal interference of the process. This is because with a limited number of cameras and occlusion of the line-of-

sight problem as stated in Section 1.2.1, the camera-based approach is accurate but not appropriate for tracking the subject in a complex environment. Drifting problem associated with IMUs makes it unsuitable for tracking over a long period. Therefore, a hybrid method that fuses the advantages from both systems is developed to track human forearm in a natural way, and the system performances are validated.

1.3.2 To build state-action policy models from human demonstrations that relate to industrial manual manipulations.

To understanding the tacit human skill in industrial manipulations, a well-established state-action policy for human demonstrations is derived. An approach that can generalise across different executions and teachers to reproduce a smooth trajectory of actions based on the recognised states is developed and tested for Peg-in-Hole task (chapter 4) and drum beater winding task (chapter 6).

1.3.3 To develop a muscle-force model to predict the forces generated from forearm muscle activations using wearable sEMG devices.

To eliminate the need to install F/T sensor during demonstrations, an indirect method is developed based on muscle activation signals generated from sEMG sensors worn on the forearm to allow the force feedback to be collected. A dynamic muscle-force model is built by mapping the sEMG signals to the known forces and the performance is validated for Peg-in-a-Hole task (chapter 5) and drum beater winding task (chapter 6). The case studies are briefly described in section 1.4.

1.3.4 To verify and evaluate the robustness of the proposed framework.

To verify the framework, the trained model is tested with additional samples, and evaluation metrics are applied to compare the results of various parameter configurations and different subjects. To evaluate the robustness of this framework, a second case study on drum beater winding task (chapter 6) was used. The results from the proposed indirect methodology for human-robot skill transferring are analysed and discussed.

1.4 Case studies

To meet the research objectives set out in this thesis, two cases were studied. Both of them are manual manipulations which dominantly rely on human forearm control. The first case study, as shown in Figure 1-2, is a Peg-in-Hole (PiH) insertion

task. This is a typical example used in the literature to demonstrate the robot's capability to perform an industrial assembly task under uncertainty that requires the teacher to demonstrate his/her tacit knowledge of the insertion skills [35]. The uncertainties of the PiH are mainly from the different peg clearances, the grasping strategy of the operators, and the different insertion paths. Compliant grippers [36] had been designed to constraint these, but they cannot eliminate all the variations. Therefore, active compliant control using multiple sensors becomes important to further adapt to the uncertainty. A hybrid approach is also promising, but the main focus of this work is learning the compliant control policy from the human demonstrations. Researchers have taught the robot PiH skills by guiding the robot arm [37]. The kinematic data are directly recorded and the F/T sensor is installed on the end effector. But these approaches do not capture the underlying human skills, therefore sensor on teacher approach is considered more suitable in this work. The human demonstrations might introduce more uncertainty such as the different experiment length, the different muscle groups and the different sensor placements. These require additional validation step and a careful calibration strategy.

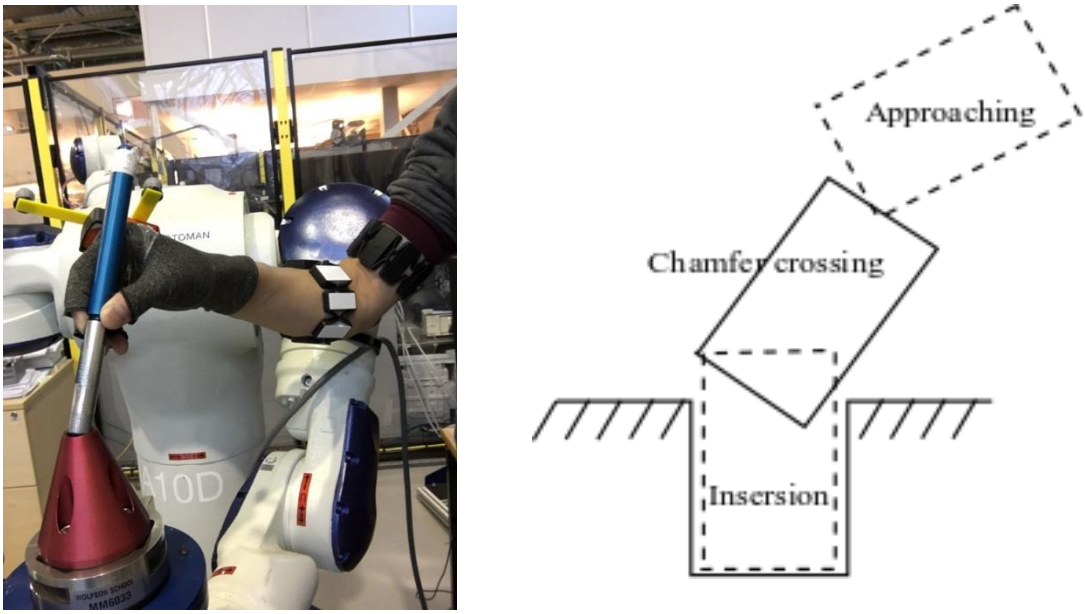


Figure 1-2 A peg-in-hole task.

The second case study is a drum beater winding process where the operator winds synthetic yarn onto a spherical rubber attached to one end of a wooden stick to create the drum beater head. The skills required in this task have been found to be largely procedural[38], but maintaining tension while the geometry is changing is one

of the key skills required to accomplish the task. Therefore, these tacit skills were demonstated and learned in this work. As shown in Figure 1-3, the operator is holding the beater head using her left hand and winding using her right hand. To simplify the case study, the left hand is assumed fixed and the right hand delivers the winding skill. In reality, the translation movements in the right hand are small and the dominant features are hand orientations. The main haptic feedback in this case study is the tension produced in the yarn. To measure the tension, a tension measurement unit was firstly designed and made. This device is not intended to be installed on the human body but it was used for building a muscle-tension model. Once the model is built, the teacher can demonstrate the winding task without having to use the tension measurement unit. The proposed LfD methodology was demonstrated and verified for both case studies.

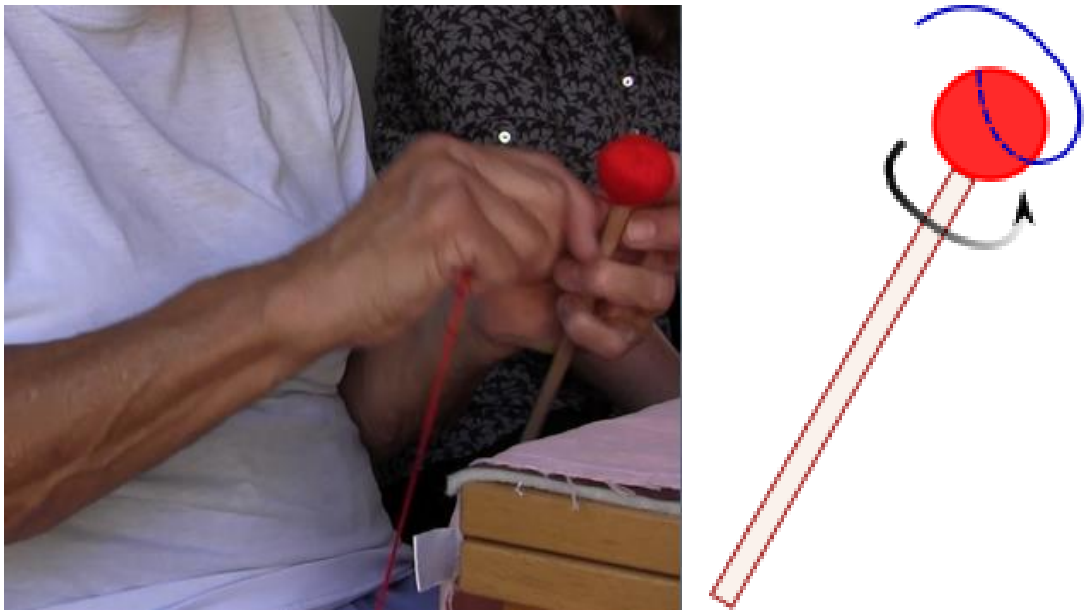


Figure 1-3 One frame of the winding process.

1.5 Thesis structure

An overview of the proposed methodology is shown in Figure 1-4. From left to right, the human demonstration (i.e., PiH task) was tracked, encoded and control policy was reproduced and evaluated. The details in each block are explained in the following chapters. The thesis structure is shown in Figure 1-5.

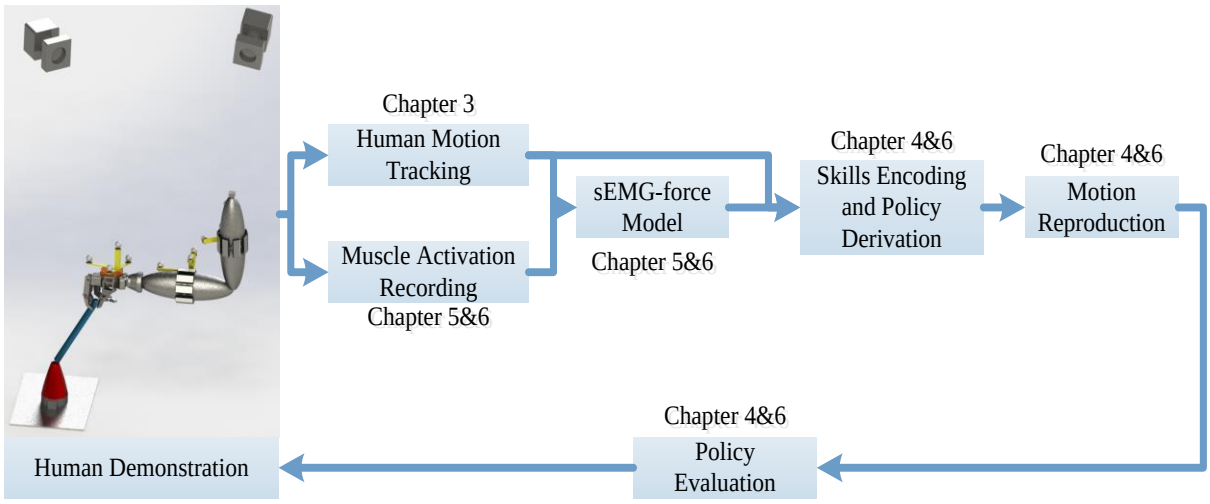


Figure 1-4 An overview of the proposed methodology.

Chapter 1 provides an overview of the research background, problems and motivation needed to understand the research gaps, and the approach undertaken to address them through four measurable objectives.

Chapter 2 provides a literature review of the LfD, human motion tracking and muscle-force modeling using wearable sensors. The research gaps pertaining to this thesis were identified from this chapter.

Chapter 3 presents a robust and reliable hybrid method to track the human forearm motions using Vicon system with two cameras and IMUs. Research objective one (section 1.3.1) is addressed in this chapter. A low cost and automatic data-driven approach has been developed to align the IMU and Vicon local frames to improve tracking reliability. The proposed hybrid forearm tracking system has overcome drifts and occlusion issues associated with the individual system.

Chapter 4 presents a method to build reproducible GMM-HMM state action policy models for industrial manipulation i.e. peg in a hole. A thorough evaluation of the time sequence motion reproduction were performed against further human experiments with an average accuracy less than 2.5 degree. Research objective three (section 1.3.2) is addressed in this chapter. The expert skills for fine/dexterous task can be automatically encoded from and reproduced from the human demonstrations and proposed probabilistic encoding approach.

Chapter 5 focuses on modeling the muscle-force relationship by using the TDNN in the PiH scenario. Research objective two (section 1.3.2) addressed in this

chapter. A predictive force modelling approach has been proposed by using the sEMG signals. A Wearable framework has been proposed through offline mapping of muscle activations to the force measurements. A thorough evaluations on PiH process with a repeatability of 3 ± 0.5 N was achieved.

Chapter 6 applies the methodology from chapter 4 and 5 to a beater winding task. Research objective two and three are addressed in this chapter. The contribution of this chapter is to apply the proposed skill transfer framework on a different industrial manipulation case study i.e. beater winding.

Research objective four is addressed from chapter 3 to chapter 6. It includes the verfications and validations on the built models, and evaluations on a different case study.

Chapter 7 summarises the key contributions of this thesis and offers directions for future work.

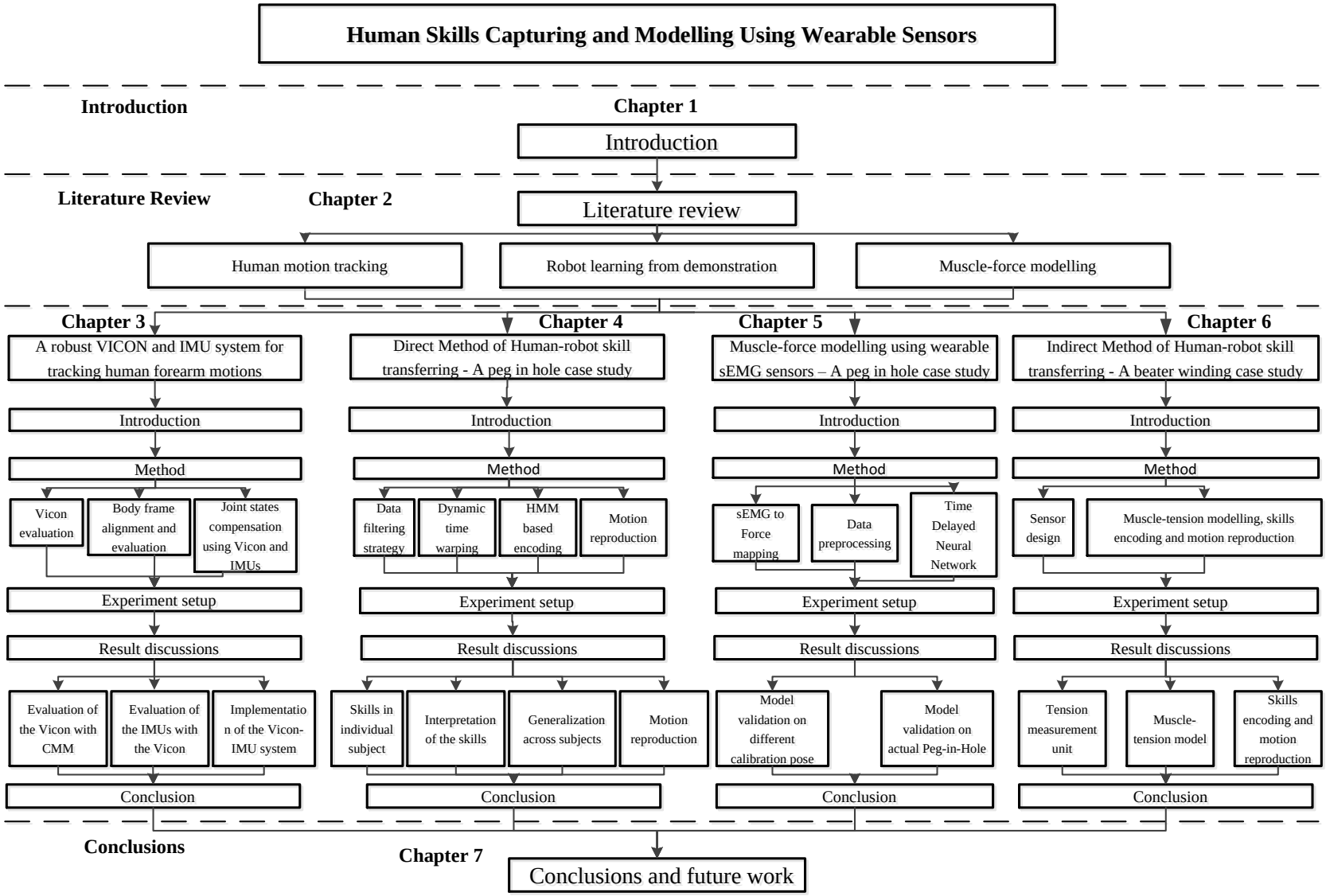


Figure 1-5 Thesis structure

Chapter 2. Literature Review

This chapter reviews the existing literature and identifies the research challenges related to transferring the human skills using wearable devices to robots. Section 2.1 provides a review of the human motion tracking approaches, focussing on the state-of-the-art technologies to reliably and accurately track human motions and their applications. Section 2.2 focuses on the methods to build the sEMG-force model using the signal processing and the modeling techniques. Finally, section 2.3 provides a review on the human skills extraction and transferring by using the robot learning from demonstration approaches.

2.1 Human motion tracking

Human motion tracking technologies enable the compliant movements to be recorded and analysed. A reliable and accurate motion capturing system is critical for the applications concerned in this thesis because the robot student has to reproduce the correct state-actions regarding its velocity or acceleration commands. Motion capture and analysis is a popular research field with many applications in areas such as, computer animations, rehabilitation, surveillance and human-machine interaction [39][40]. Figure 2-1 is an overview of the available tracking system where the human motions can be recorded using on-body and visual sensors [41].

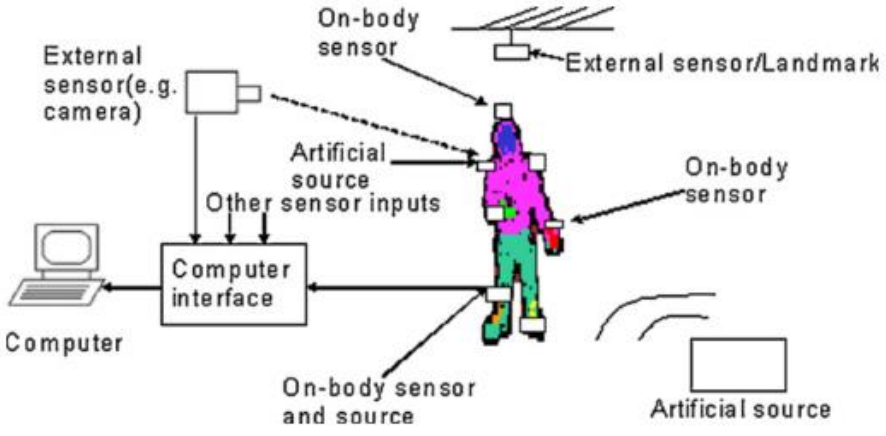


Figure 2-1 An overview of the human motion tracking system [41]**.**

Mainly, motion capturing systems can be classified into three categories: visual based tracking, non-visual based tracking, robot-aided tracking. In visual based tracking,

both marker-free and marker based approaches have been investigated. In the marker free approach, the 2D and 3D cameras are directly used to monitor the actors' motion and fit a model from the data points in the view. In the marker based approach, the actors need to wear obtrusive devices such as reflective markers or fixtures to assist motion capturing. Different from the visual based tracking, a non-visual based tracking uses systems such as Inertial Measurement Units, data gloves and acoustic sensors without requiring 'line-of-sight'. In the robot aided tracking system, the human motions are tracked through the exoskeleton systems. In the following sections, various motion tracking systems have been reviewed in depth.

2.1.1 Visual based tracking system

From the literature, the marker-based approach is mainly using reflective markers with more than one monocular camera [42]. A large amount of commercially available real-time motion tracking systems is marker-based which requires the operators to wear devices, which might obstruct the operator when carrying out skilled work. The marker-based tracking system can be passive, active or hybrid. In the passive case, the markers do not generate any light, only reflect the incoming light. In contrast, the active marker can produce light (i.e. infrared) that can be detected by the camera system.

Qualisys [43] as shown in Figure 2-2, and Vicon [44] as shown in Figure 2-3 are commercial products which use passive markers. CODA [45] and Polaris [46] systems are the commonly used active visual tracking products. The advantages of the marker based system is due to the contact-less sensing, six degree-of-freedom (6 DOF) measurement, high sampling rates, multiple simultaneous segments tracking and high accuracy and precision measurements [47]. The accuracy of such system is dependent on various factors such as the resolution of the monocular camera, the size of measuring volume, the cameras configuration around the measurement volume and the accuracy of the extrinsic and intrinsic parameters computed from the calibration procedure for each camera [47]. Some researchers have evaluated and reported the marker based system accuracy in small working volumes [47].

Figure 2-2 Qualisys system with 5 camera configuration [43]**.**

Figure 2-3 An operating Vicon system [44]**.**

Yang *et al*. [48] used a Vicon MX system with five F40 cameras to measure the bone deformation in a 400×300×300 mm^3 volume. In the optimal conditions, a displacement of 20 μm had 1.2μm -1.8 μm absolute error.

Liu *et al*. [49] used the Qualisys ProReflex-MCU120 (658×500 pixels, CCD) cameras to measure the micro displacements of teeth. The field of views was 68.18 mm ×51.14 mm, and the accuracy of displacements was ±1.17%, ±1.67% and ±1.31% in axis wise terms. The corresponding standard deviations were ±1.7μm, ±2.3 μm and ±1.9 μm. The measurements range from 20μm to 200μm.

Windolf *et al.* [50] systemically evaluated a five Mcam-60 cameras (1012×987 pixels, CMOS) Vicon system. The experiment used an 180×180×150 mm^3 volume to capture small magnitude biomechanical motion. The samples were collected in 294 positions according to a 7×7×6 grid with 30mm uniform spacing. Each measurement

was compared with an XYZ scanner with 15 μm linear encoder accuracy. The evaluations were conducted for various variables such as camera positions, manual versus scanner based dynamic calibration, accuracy outside the calibration volume and marker size. The optimal set of variables achieved an overall accuracy of 63 ± 5μm. It was concluded that the main factors are the camera placements, the marker size (larger markers increase the accuracy) and lens filtering to sharpen the target edge. Also, the accuracy in the calibrated volume was significantly larger than un-calibrated volume.

For larger working volume, a six-camera Vicon system with a 6.8×3.8×3.8 m^3 calibrated volume has been evaluated for static positional accuracy [47]. The true ground measurements were provided by a highly accurate laser tracker: Leica Absolute Tracker AT901B. According to Windolf *et al* [50], the cameras were deployed in the optimal placements such that at least two cameras were available at any point of the volume for triangulation. A systematic approach had been proposed to evaluate the measurements in the dense space over a large volume using large markers of diameter 38.1mm; the effects of the calibration artifact in the system accuracy. It was found that the mean errors in the active and passive calibration approaches were 1.48mm and 3.95mm respectively (the maximum errors were 4.03 mm and 7.15mm respectively).

The manipulations studied in this thesis had a working volume from middle to large size (range from half metre to a couple of metres). In this range, Vicon systems are extensively used in the applications such as gait analysis [51], rehabilitation [52], animation in the entertainment [53] and improving tracking accuracy in robotics [54].

Yang *et al*. [52] proposed a cost-effective and portable system for motion analysis and post-stroke impairment assessment, using a single camera. The reason of using single camera was that the system did not require three-dimensional tracking. The markers were attached to the human upper limb. The results showed that the proposed decision support system had the capability to offer stroke survivors and clinicians an affordable, accurate, and precise assessment method suitable for home healthcare.

Lee *et al*.[44] used a twelve cameras system for real-time control of three-dimension avatars in the computer games and virtual environments. They proposed an approach by obtaining a large dataset of candidate human motions, which was sufficient for exploitation of the real-time control strategies for the avatars. They

showed the flexibility of the approach by using four different applications and evaluated the avatar motions with the recorded human motion.

Schmidt *et al*. [55] proposed a protocol that can measure the free rotations of the wrist and elbow movements in three-dimensional space. The method can be used for diagnosis and treatment of disorders in the upper extremities. The experiment setup as shown in Figure 2-4 was intuitive and suitable for reducing the influence of the skin movements because the markers were rigidly fixed on the plate. More importantly, this setup can also be used for analysing the manual manipulation works such as assembly and polishing tasks.

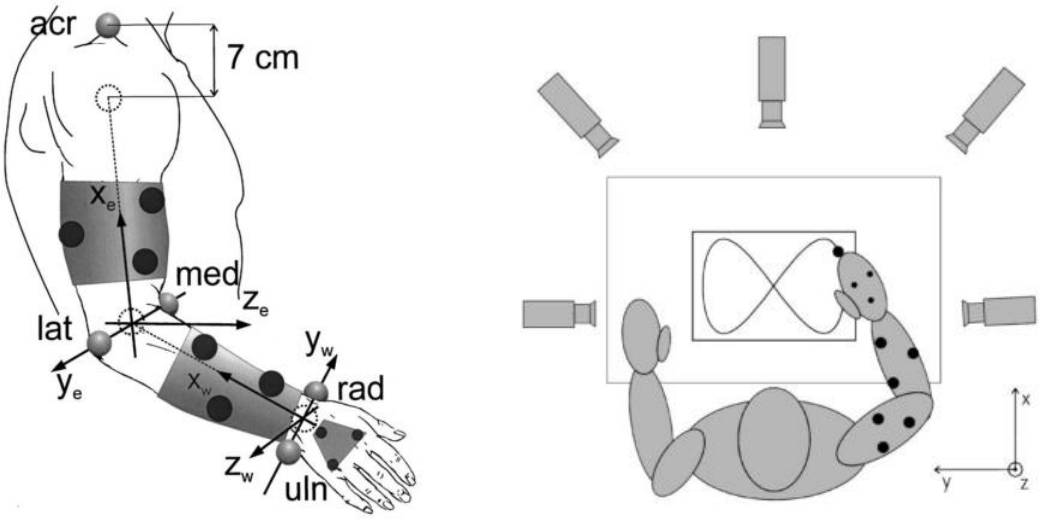


Figure 2-4 A protocol for tracking human upper extremities using Vicon [55]**.**

For lower extremities, the marker-based approach has been used for gait simulation using AnyBody Modeling system [56]. As shown in Figure 2-5, 15 markers were installed on both legs to reconstruct the gait motions. They can be used jointly with other sources of measurements i.e. force plate on the ground and muscle activation levels. The simulation system generates comprehensive analytical results for clinicians.

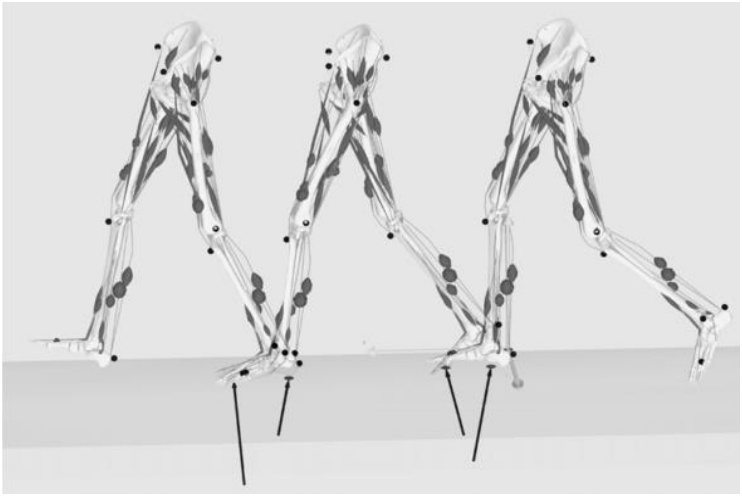


Figure 2-5 Marker placements in the lower extremities [56]**.**

The marker-based approach can be accurate, but it has some disadvantages: (1) the marker can displace during movement; (2) the markers can become removed from the skin; (3) the skin can move and introduce noise to the data; (4) The body landmarks can vary in between subjects so that they make the measurement unreliable. Therefore, the following review of the marker-free approaches is necessary.

Nowadays, the tracking of moving objects can be highly accurate due to the high resolution of the camera with over a million pixels. The cameras are portable, and the parameters are easy to configure by the user. Therefore, this makes a 2D camera suitable for surveillance applications. However, challenges arise when the computation of the 3D localisation with an optimal latency of the data is required. Also, due to the limited bandwidth for accurate data representation, high-speed cameras are required [57].

Different from the marker-based method where the objects can be reliably tracked by the reference markers, the marker-free method typically uses 2D and 3D approaches that directly capture the object by fitting the models. In the 2D approach, the main research interest are focusing on identification of the explicit shape models by introducing prior knowledge of human body segments [58]. The active shape model [59] without using explicit shape model is also commonly used. In 3D approach, model-based tracking e.g. stick figure [60] and volumetric modeling [61] are commonly used to overcome the self-occlusion and collision problems. Feature-based tracking is another popular research area where global features [62][63] and local features [64][65] are utilised to be matched across images.

Zhang *el al*. [66] tracked a 22-DOF human kinematic model by fusing multiple depth cameras. They seamlessly registered multiple depth images from depth cameras into one single joint point cloud in which the pose estimation is derived. To track the high-dimensional human poses, particle filtering algorithm was employed. The particle likelihood is computed based on the distance of each observed point to a parameterized human shape model. The results indicated that the proposed method is considerably more robust for unconstrained motions and under occlusions.

Shotton *et al.* [67] proposed a novel approach to predict the positions of the human joints by only using one depth camera (Kinect) without time information. The estimations of the joint positions were derived from a large training dataset which contains sufficient variations for the classifier to recognize body parts invariant to pose, body shape and occlusions, etc. The system ran at 200 frames per second and achieved highly accurate tracking performance by using both synthetic and real test data.

Ganaphathi *et al.* [68] proposed a marker-less tracking method for pose estimations of the whole human body segments using time-of-flight camera. The filtering algorithm first encoded the probability model of the articulated body parts by using Bayesian network; then the pose inference was done by given the current frame of the range images. They had evaluated the method by 28 real world sequence using ground-true values from maker based motion capturing system.

Sminchisescu *et al.* [69] presented a robust method for recognizing the 3D human pose from monocular video sequences by considering the joint limits, non-self-intersection constraints, a search strategy guided by a rescaled cost-function covariance and robust image feature matching. They demonstrated that the mentioned considerations were essential for reliable human motion tracking to overcome problems such as self-occlusion, model imperfection, and high dimensional features.

Agarwal *et al.* [70] proposed a learning-based approach for detecting 3D human body pose from monocular sequences. Neither explicit model nor prior labeling of the body parts was required by this method. The pose was detected by nonlinear regression method (Support Vector Regression) using histogram descriptors derived from the silhouette shapes. A 4-6 degree of the mean angular error was obtained in the walking sequences.

2.1.1 Non-visual based tracking systems

One of the disadvantages of the visual based tracking system is that it requires "line of sight." Therefore, it is difficult to consistently and effectively monitor the operators' movements during the manipulations in the ambient working environment with clutters. In this case, non-visual based tracking systems such as inertial, magnetic and ultrasonic systems are often used and are available as commercial products. Data gloves are also gaining more attentions due to the modern sensing techniques.

Inertial sensors have been extensively used in the sports industry, healthcare, navigation and human modeling in the augmented reality [71]. The data from the sensors can be wirelessly transmitted to the workstation for further processing. The biggest advantage of the inertial sensor is that there are no restrictions in the working areas so the operator can freely move about. However, the position and angle of the inertial sensor suffer from drifting due to the fluctuations of offsets and measurement noise [72]. Therefore, the main challenge of the inertial sensor design is to overcome the drifts.

Mtx is an Inertial Measurement Unit (IMU) that measures 3D orientation plus acceleration and angular velocity [73]. The newly designed Xsense MVN motion capture suit, as shown in Figure 2-6, is an easy-to-retrofit, cost efficient for full body tracking applications with the integration of several miniature Mtx sensors [73]. In the homogeneous earth-magnetic field, the individual sensor has 0.05° root-mean-square (RMS) angular resolution; ≤1.0° static accuracy; and 3° dynamic accuracy.

Figure 2-6 Single Mtx sensor (left) and Xsense motion capture suite (right) [73]**.**

Zhou and Hu [74][75] introduced a novel tracking strategy for human upper limb motion. The upper limb motion was represented by using six joint variables which form a kinematic chain. A simulation annealing approach was implemented to reduce the measurement error. Furthermore, a Kalman filter based method was used to depress the noise in the measurements by using the on-board accelerometers and gyroscopes [76]. The experiment indicated a reduction of the drift and noise.

Yu *et al.* [77] installed IMUs to track the full human body motions of a skier and characterised ski turns and performance. The goal of the research was to identify the optimal sensor installation locations. The validation results indicated that the proposed method could effectively evaluate the skier performance with minimal interference of the skier's motion. Moreover, it can be used for routine training of the professional skiers.

Sessa *et al.* [78] proposed a systematic approach to evaluate the performance of the IMU devices with a Vicon system. They used Vicon as a ground true reference system and a Dynamic Time Warping (DTW) algorithm to solve the sensor misalignment issues. They used an Extended Kalman filter to estimate the tri-axes orientations. The proposed method enabled the comparisons of the different IMUs by using the same Vicon system as a reference.

Kang *et al.* [79] proposed a design and implementation of a real-time human body motion capture system using IMUs. Each body segments were captured by a single IMU. A lie group setting was used to represent the kinematic tree of the full body

configurations. A golf swing motion was used in the simulation environment to validate the feasibility of the proposed approach.

Roetenberg *et al.* [80] used multiple IMUs to capture human motions for gesture recognitions. In total, six different gestures were used, and data were collected from eleven participants. A Support Vector Machine (SVM) and Neural Network-based methods were implemented and compared on the same dataset. The results indicated the classification accuracies are satisfactory and the speed of gesture recognition was acceptable for interactive usage.

Qiu *et al.* [81] proposed a wearable sensor approach to capture human lower limbs as shown in Figure 2-7. The goal was performance evaluation of the specific walking and stair ascent capabilities. The Denacit-Hartenberg (DH) convention was used to set up the kinematic chains when the foot stayed stationary on the ground and produced state constraints to reset the estimation error on the knee's position. The method to solve the drifting problem when the IMU had to operate in the long term was developed. The Vicon system was used as the ground true reference measurements. The results were satisfactory and consistent for tracking of human lower limbs.

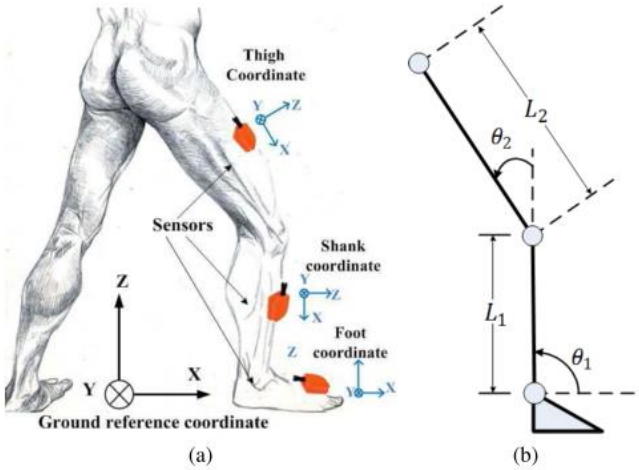


Figure 2-7 Schematic plot of the human lower limbs. (a) Sensor attachment. (b) D-H models. [81]

The magnetic based motion tracking system requires the performer to wear an array of the magnetic receiver which tracks location with respect to a static magnetic transmitter. The advantages of this type of sensor are due to its high sampling rate, being invariance to occlusions and wearability, which make it suitable for virtual reality and motion capturing. However, the magnetic sensor also suffers from latency due to

the asynchronous nature of the sensor measurements and jitter when there are ferrous or electronic devices in the surrounding causing noise in the measurements.

One of the commercial products, as shown in Figure 2-8 is MotionStar produced by the Ascension Technology Corporation in the United States [82]. It has ±3.05 m translation range; ±180° for Azimuth and Roll, ±90° for Elevation in the angular range; static resolution (position) is 80 μm at 1.52 m range; static resolution (orientation) is 0.1 RMS at 1.52 m range. It uses direct current (DC) electromagnetic tracking technology, which generates less metallic distortion of the measurement than the alternating current (AC) electromagnetic tracking systems. Another system is LIBERTY from Polhemus [83]. The update rate is 240 frames per second per sensor. The latency is 3.5 ms, and the resolution is 38 μm and 0.0012° orientation at 300 μm range, A method has been proposed in [84] to convert the magnetic sensor measurement to the human anatomical rotations.

Figure 2-8 a wireless MotionStar system [82]**.**

Acoustic system uses a transmitter and receiver to collect sound wave signals. The flight duration of the ultrasonic pulse is timed and measured. It is widely used in medical applications but rarely used for motion capturing. The drawbacks are: (1) large device is required because the efficiency of the acoustic transducer relies on the size of active surface area; (2) the frequency of the ultrasonic wave has to be low to improve the detection range, but this affects continuous measurements due to latency; and (3) "line of sight" is required by the acoustic system.

Data glove has been studied for analysing hand gestures since the late 1970s. It transduces the finger flexion and bending into electrical signals to estimate hand pose.

The device can be used for hand therapy due to the flexibility, ease to wear, accuracy and being lightweight. The principle of the glove is translating the amount of light that escape from the fiber optic cable to the amount of bending in the finger. Therefore, a recalibration is required for each user. One commercial product is CyberGlove system with one CyberGlove unit (as shown in Figure 2-9). It has a serial connection to the host PC, a virtual simulator for the hand postures, and calibration software [85]. A PowerGove designed by Abrams-Gentile Entertainment operates with flat plastic strain gauge fibres that are coated with conductive ink along the sensor stripe. The degree of flex of the finger can be measured from the changes in the resistance when the finger bends. Another research introduced an approach to use materials of Lycra and Nylon blend on each of the five fingers. The repeatability test indicated average variations of 2.96% in the hand gripping position [86].

Figure 2-9 A CyberGlove [85].

2.1.2 Robot-aided tracking system

The robot-aided tracking system uses exoskeletons (as shown in Figure 2-10) to enable a human to complete difficult tasks, i.e. lifting and moving heavy loads, and arm rehabilitations. In its early and simple form, Taylor [87] proposed a 2DOF robot arm wore by the patient to allow movements of a shoulder and elbow in a horizontal plane. He further proposed a five exoskeleton system to enable daily living operations. The results were validated against with the goniometer for arm pose identifications. Another early example of the exoskeleton system was “Handyman” with 10 DOF electrohydraulic arms [88][89]. Even though the system had potential for many applications, it is still not commonly commercial available due to weight, safety hazards [90] and limitations in the functional anatomy of the human arm. Therefore, it

is unsuitable for the implementation scenarios of this work because the additional fixtures will make the operator uncomfortable while completing the manipulations.

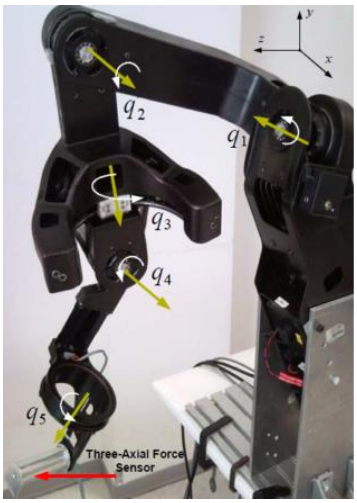


Figure 2-10 An example of a 5 DOF exoskeleton [91]**.**

In this section, motion tracking systems have been reviewed. Table 2-1 summarises the literatures which fall into different categories. Among the choices, marker-based Vicon system and IMU system are more suitable candidates due to their accuracy and portability. Table 2-2 is a comparison of the two. Both systems are able to track human kinematic motions with sufficient angular accuracy (±3˚), but the measurements from Vicon are more reliable without drifts. The IMUs are more flexible and suitable for free space movements. Therefore, both limitations need to be addressed in a hybrid system will provide a more reliable solution.

Table 2-1 A classification of the motion tracking systems.

Human motion tracking						
Non-visual tracking				**Visual tracking**		**Robot-aided tracking**
Inertial-based	**Magnetic based**	**Other sensors**	**Glove based**	**Marker-based**	**Marker-free based**	
[77][92][78][79][73][80][81]	[72][82][83][84]	[93][94][95]	[85][86] [96]	[55][47][52][44][48][49][50][51][53]**[31]**[56][97][43]	[66][67][68][69][70][98][99][100][58][59] [57]	[87][88][89][90][91]

Table 2-2 A comparison of the Vicon and IMU system

	Vicon	IMU
Pros	Tracking without drift	More suitable for free space movements
Cons	Line-of-sight occlusion	Tracking with drift

2.2 sEMG-force modelling

Electromyography signals are electrical activity of the muscle's motor unit, which is produced during muscle contraction and controlled by the nervous system. The signal represents the physiological and anatomical properties of muscles. From the literature, both non-invasive and invasive electrode to collect the surface and intramuscular electromyography signals respectively [101]. Due to its simplicity, it is preferable to use surface EMG (sEMG) signal to record the intensity of the superficial muscle activities [102]. They have been applied in many applications, including prosthesis or rehabilitation devices, human machine interaction [106-107], entertainment and clinical/ biomedical. This section starts with the review of the existing signal processing techniques for sEMG feature extractions and recognition, followed by the state-of-art techniques to build sEMG-force models. Although sEMG is an important indicator of the muscle activation time and fatigue index which is useful for muscle rehabilitation [105], the muscle-force relation is the primary focus of this research because the model will be used to replace the role of the F/T sensor.

2.2.1 sEMG signal processing and feature extraction

The raw sEMG signal contains various type of noise which makes it difficult to analyse, especially when movements with multiple DOF occur [106]. To effectively use it, an accurate signal processing is essential. When receiving sEMG signal, various background noises are recorded due to the presence of electronic equipment and physiological factors [101]. At the beginning of the section, an overview of the various noise sources and ways to overcome them are discussed.

a) Noise sources

Different types of noise can be found in sEMG signals. Depending on the subjects, factors that have influence to the signal are: the individual skin formation, the blood flow velocity, the measured skin temperatures, the tissue structure, and the measurement positions, etc [101]. They influence the efficiency of the feature extractions and prevent the sEMG signal from practical usages. The amplitude range of the sEMG signal is 0-10 mV (+5 mV to -5mV) before amplification. It is important to characterise the electrical noise, which can be categorised into the following types:

- The inherent noise in the electronic devices has frequency components from 0 Hz to several thousand Hz. The commonly used non-invasive electrode is made of silver

chloride. It has been found this material provides adequate signal-to-noise ratio and generates a stable electrical signal. When the electrode size becomes larger, the impendence decreases. But it is not preferable to have a very large electrode for practical reasons. A high impendence will decrease the signal-to-noise ratio, which degrade the quality of the signal. In general, the only way to eliminate the inherent noise is to use high-quality instruments [101].

- The movement of the cable connections in between the electrode and the amplifier and the interface between the skin surface and the electrode creates motion artifacts. Whenever muscle activates, the muscle fibers generate electrical activities [107]. Noise occurs when the muscle, electrode, and skin move respect to each other. A conductive gel layer can help to improve the contact so that the movement artifact noise can be reduced. Another type of movement artifact is due to the different skin layers. The gel cannot reduce such noise, but it can be attenuated by skin preparations to reduce the skin impedance [109-111].
- The ambient noise mainly from electromagnetic radiations will affect the sEMG signals. It is almost impossible to eliminate noise from the ambient environment, which magnitude may be much greater than the sEMG signal. The noise is also called power line interference (PLI) with 50 Hz frequency. If the frequency contents are within the sEMG signal, it is important to remove it and its harmonics [110]. Methods such as adaptive notch filter, FIR notch filter, IIR filter and Laguerre filter have been applied to attenuate the noise from PLI [101].
- The sEMG signal is random in nature due to the firing rate of the motor units which in most cases, fire in the frequency range from 0 to 20Hz. This noise is unwanted, and the removal of the noise is essential.

In order to analyse the sEMG signal, the three factors that affect the signal quality need to be considered. Firstly, the causative factor which can be divided into two classes:

- Extrinsic: The main factors are the electrode structure and placement [105]. For instance, the detection surface, shape of electrode, distance between electrode detection surface, location of electrode with respect to the motor units in the muscle, location of the muscle electrode on the muscle surface with respect to the lateral

edge of the muscle, and orientation of the detection surfaces with respect to the muscle fibres are the main factors in this category.

- Intrinsic: This is due to the complex nature of the physiological, anatomical and biochemical structure of human being [111]. These factors are affected by the number of active motor units, fiber type decomposition, blood flow, fiber diameter, depth and location of active fibers and amount of tissue between the surface of the muscle and the electrode.

Secondly, the intermediate factors which are the physiological and the physical phenomena affected by one or more causative factors. Some reasons are behind these factors including, for instance, the band-pass filtering aspects of the electrode along with its detection volume, the superposition of action potentials in the detected sEMG signal, conduction velocity of the action potential that propagate along the muscle fiber membrane. The cross-talk [112] in between the muscle groups can be another cause.

Thirdly, the deterministic factors are the outcomes of the intermediate factors. The motor firing rate, the number of active motor units, motor firing rate, and mechanical interaction between muscle fibers have a direct impact on the recorded sEMG signal and the generated force. The amplitude, duration, and shape of the motor unit action potential can also be responsible to this relation.

To optimise the sEMG signal quality, two approaches are required: firstly, to enlarge the signal-to-noise ratio as much as possible; secondly, minimise the distortion of the sEMG signal caused by the unnecessary filtering.

During the processing stage, both half-wave and full-wave rectification [111] are commonly used. The half-wave only retains the positive data and discard the negative data. The full-wave is preferred since the absolute value of each data point is used so that all the information is self-contained.

b) Signal processing

Many literature have proposed advanced methodologies, including wigner-ville distribution (WVD), wavelet transform (WT), empirical mode decomposition (EMD), Independent component analysis (ICA), and higher-order statistics (HOS) for analysing the sEMG signal [101]. In signal processing, one of the most fundamental concepts is time-frequency analysis. The Wigner-Ville distribution (WVD) is one of the popular

methods for analysing sEMG signal. It has an excellent localisation property due to the high concentration in the instantaneous frequency and time of the signal. In [113], they used WVD to present the frequency ranges of the motor unit. But the cross-term effect and noisy treated signal means it is not suitable for sEMG signal processing with multi-component.

Wavelets transformation as an alternative to the Fourier transform method has gained more attentions. The wavelet transform has two forms: discrete and continuous. It can efficiently transform the signal with various resolutions in both time and frequency domains. The Discrete Wavelet Transform (DWT) has been implemented to analyse non-stationary sEMG signals, but it produces a high-dimensional feature vector [114].

The basic analytical expression for wavelet transform is presented in equation (2-1) below. The wavelet corresponding to scale *a* and time location *b* is given by:

$$\psi(a,b) = \frac{1}{\sqrt{|a|}}\psi(\frac{t-b}{a}) \quad \text{.....(2-1)}$$

Where $\psi(t)$ is the "mother wavelet" which takes the form of a band-pass function. The factor $\sqrt{|a|}$ is to ensure the energy preservation. *a* and *b* are discretizing timescale parameters and each pair generate a different type of wavelet transform.

Successive low-pass and high-pass filtering in the discrete domain computes the general equation of DWT, is given below in equation (2-2):

$$x(t) = \sum_{k=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} d(k,l)2^{\frac{k}{2}}\psi(2^{-k}t - 1) \quad \text{.....(2-2)}$$

Where k is related to a as $a = 2k$; l is relate to b as $b = 2kl$; and $d(k,l)$ is a sampling of $W(a,b)$ at points (k, l).

Daubechies [115] analysed the non-stationary time series at many different frequencies by using wavelet transform. The various types of wavelets functions have different time-frequency structures. Farge [116] considered the various factors that should be considered when selecting the wavelet function. Guglielminotti and Merletti [117] showed a good energy localisation capability of the wavelet transform in the time-scale plane when the shape of the Motor Unit Action Potential (MUAP) is matched with the wavelet function. Laterza and Olmo [118] used wavelet transform as

an alternative method to other time frequency representations to overcome the resolution problem. The results showed that the WT is not affected by the cross term which is useful for dealing with multi-component signals.

The wavelet de-noising algorithm has been widely used in signal pre-processing stage for sEMG upper- and lower- limb movement recognitions over the past few years [119]. Phinyomark *et al*. [120] demonstrated that using wavelet de-noising algorithm can effectively remove the interference of random noises from sEMG signals (i.e. white Gaussian noise (WGN)). They proposed a basic idea of a wavelet-based de-noising procedure. Five processing parameters needed to be selected: (1) the type of wavelet basis function; (2) the threshold selection rule; (3) the scale; (4) the threshold rescaling method, and (5) the thresholding function. Among the parameters, selection of the wavelet function types are of most critical and it needs to be justified based on the type of applications and characterisation of the signal. Five wavelet functions (*db2*, *db5*, *sym5*, *sym8* and *coif5*) were compared in [120] and their performances were validated by using mean square error. The results indicated that a scale level 4 provided the better performance when compared with other scale levels. Moreover, a fifth order of *Coiflet* function provided the best reconstruction for sEMG signal [121]. Jiang and Kuo [122] assessed four classical threshold estimation functions and demonstrated that the sEMG signal was invariant to the selection of the threshold estimation function. Kumar *et al.* [123] determined the muscle fatigue by using the *Symlet* function (*sym4* and *sym5*) with decomposition level 8 and 9. Hussain and Mamun [124] showed that a wavelet form with *db45* showed the best contrast when they analysed the sEMG signal using both bi-spectrum and power spectrum compared to the other four wavelet forms (*db2, Haar, sym4* and *sym5*) within the range 50 to 70 Hz.

A summary of the wavelet function with their families are listed in Table 2-3. Chowdhury *et al.* [101] concluded that analysing sEMG signal using Daubebchies's function provides successful results. They recommended using a *db* function (*db2, db4, db6, db44* and *db45*) at decomposition level 4. Also they applied them to the raw sEMG signal from the right rectus femoris muscle during maximum walking speed.

Table 2-3 List of 324 wavelet function from 15 wavelet families [101]**.**

Wavelet Family	Wavelet Subtypes
Haar	db1
Daubechies	db2-db45
Coiflet	coif1-coif5
Morlet	morl
Complex Morlet	cmor
Discrete Meyer	dmey
Meyer	meyr
Mexican Hat	mexh
Shannon	shan
Frequency B-spline	fbsp
Gaussian	gaus
Complex Gaussian	cgaus
Biorthogonal	bior
Reverse Biorthogonal	rbio
Symlet	sym

Higher order statistics (HOS) are defined as spectral representations of higher order cumulants of a random process. The spectral representation of HOS, such as moments and cumulants of the third order and above, are known as polyspectra or higher order spectra. To effectively process the sEMG signal, HOS is applicable because it is capable of deviation identification from linearity, stationarity or Gaussianity in the signal [125]. The advantage of HOS is due to the accurate phase reconstruction is possible in the HOS domain, while the second-order-statistics (SOS) is phase-blind [126]. In 2004, Shahjahan Shahid [127] proposed to use HOS for electromyography processing and characterization due to its advantages over SOS. He modelled the bispectum of a time series signal as the output of a linear system for the purpose of identification and characterization of the given system using "Bispectrum of Linear System". Also, HOS is useful for modelling nonlinear and non-Gaussian processes. Kaplanis [128] used HOS to characterize the Gaussianity of the sEMG signal by using the bicoherence index. As a matter of the fact, the distribution of the sEMG signal is highly non-Gaussian at low and high levels of force, whereas the maximum Gaussianity is achievable at the mid-level of Maximum Voluntary Contraction (MVC). They used the HOS technique to extract new parameters (power spectrum median frequency) that could enhance the characterization of sEMG signal. In the literature, HOS has been widely applied for diagnosis of neuromuscular disorders, where the parameters, such as amplitude, spike duration, number of phases, number of turns, etc. should be taken into consideration. The HOS method is able to characterize and detects the non-linearity of the sEMG signal and estimate both the amplitude and phase information successfully. Kanosue *et al*. [129] used HOS as an important signal

processing method for quality neuromuscular diagnosis to obtain information on innervation pulse trains and MUAPs characteristics.

Other methods have been proposed by various researchers for processing sEMG signal. Some of the methods are briefly described here.

Karlsson and Nystrom [130] proposed a real-time system for electromyography signal analysis. They aimed to develop a system for clinical use with the characteristics of graphics feedback, flexible parameter selection, and flexible addition processing and standard method. The short-time Fourier transform was proposed to produce a time-frequency representation of a signal. The drawback of this method was the stationary assumption of the signal. In fact, myoelectric signals are non-stationary due to the inherent physiology of the organs even without voluntary change of muscle state.

Empirical mode decomposition (EMD) is a moderately new, data-driven adaptive technique for analysing the non-stationary and non-linear signals. EMD provides the underlying notion of instantaneous frequency and insight into the time-frequency signal features. Huang *et al.* [131] firstly proposed EMG method as a sifting process that estimates intrinsic mode functions (IMFs). The aim of EMD is to decompose a multi-component signal into a number of virtually mono component IMFs plus a non-zero-mean value of the residual component. Andrade *et al.* [132] firstly used EMD method for decomposing the electromyography signals into a set of IMFs. During the signal processing, EMD is useful to filter background activities due to the fact that it is a non-linear method that can deal with non-stationary data. Also it does not need the assumption of the basis function like the wavelet transform. Comparison results in between EMD method and several wavelet functions (*db2*, *db3*, and *db4*) showed the efficiency of the EMD approach. However, the disadvantage is that computing IMFs takes a lot of time when compared to wavelets. Another drawback of the EMD is that it is more sensitive to the presence of noise, and suffers a mode-mixing problem. To accommodate this, Ensemble EMD (EEMD) was introduced to remove these side-effects [133].

Independent Component Analysis (ICA) is another popular signal processing techniques. As a statistical method, ICA assumes the original signal from the mixture of signals. Comon [134], firstly proposed this approach for transforming an multivariate random vector into components that are statistically independent from each

other. The ICA is feasible for signal decomposition of a sEMG signal into a number of independent components. Different types of ICA have been proposed in the literature, such as the Fast ICA, the joint Approximate Diagonalization of Eigen-matrices (JADE), and maximum likelihood algorithm. Among these choices, the Fast ICA is a very popular method due to its simplicity, satisfactory results and fast convergence. Nakamura *et al.* [135] reported that ICA is very useful for decomposing sEMG signal into MUAPs from different muscle sources. Furthermore, the Fast ICA outperformed the Principle Component Analysis (PCA) in discrimination of the MUAPs for sEMG signal decomposition. The fast ICA has problems in automatic decomposing sEMG signals, therefore, JADE algorithm [136] was proposed and proved to be more effective than the fast ICA method. Garcia et al. [137] used JADE method to solve overlaps of MUAPs, and showed that this method is not affected by the added noise. However, the drawback is the inter-channel delay.

In this section, various methods have been reviewed for sEMG signal processing. The key feature of these methods is to decompose the complex and noisy sEMG signal into its simplified forms with various information. The information might be the true sEMG signal in different level of details, Power Line Interference (PLI) due to signal-to-noise ratio and artefacts. Among the choices, the wavelet based transformation is chosen in this research due to its time and energy localisation property and implementation simplicity.

c) sEMG features extraction

sEMG signal is often used for classifications, however, the raw signals are normally not suitable for this purpose, thus, features are used instead to improve the classification efficiency. Researchers have proposed different types of sEMG features as input to the classifier. To select the optimal features, various properties are considered, i.e. Classification accuracy, computation complexity and robustness [138]. There are mainly three types of features in different domains: time domain, frequency domain and time-frequency domain features. Time domain features are developed by Hudgins et al [139]. They used Mean Absolute Value (MAV), Slope Sign Changes (SSC), Mean Absolute Value Slope (MAVS), Waveform Lengths (WL) and Zero Crossings (ZC) for sEMG feature representations [140]. A careful selection of these "Hudgins features" as input provides a higher classification rate than the raw data [141]. Englehart *et al.* [142] compared the Time Frequency Domain features (TFD) with the

time domain “Hudgins features” in myoelectric signal pattern classifications. The results indicated that the features based on Wavelet packet transform (WPT) were the most effective method with small classification error. The time-frequency domain features are effective feature sets especially for transient myoelectric signal pattern classification. But these features normally have high dimensionality and high resolution problem which require dimension reduction to complement the feature extraction [143]. Angkoon Phinyomark and Limsakul [140] concluded that features based on Mean Frequency (MNF), Median Frequency (MDF), Mean Power (MNP), Mean Peak Frequency (PKF), Time-to-peak Force (TTP), Spectral Moments, Frequency Ratio (FR), Power Spectrum Ratio (PSR), and Variance of Central Frequency (VCF) are generally not good in sEMG signal classification.

In [17], Karlik tested various time domain and time-frequency domain features with different machine learning algorithm for characterisation of sEMG signals for myoelectric control of human arm prosthesis. The results indicated a near perfect recognition performance (95% to 98% rate of success). Both Auto-regression (AR) and WT features give better results. But the feature computed from Auto-regression (AR) coefficients needs less computing resources.

In [20], Ma, Thakor and Matsuno used a Nonnegative Matrix Factorisation (NMF) algorithm to control a prosthesis hand by using both time and time-frequency domain features. Nine movement gestures as shown in Figure 2-11 are used for both offline and on-line recognitions. They concluded that for offline experiment, the time domain features are suitable for NMF, but the frequency domain ones may not be. The MAV and RMS features only perform well when the signal is zero mean, but WL features are generally robust for signal with either zero or nonzero mean. For online experiment, the performance is nearly 100% with minor misclassifications due to low activation strength levels.

In [144], two novel mean and median frequencies (MMNF and MMDF) features are presented for robust sEMG feature extraction and evaluated against sixteen existing features in noisy environment. The results indicate that MMNF shows better recognition performance. Table 2-4 shows the commonly used sEMG feature extraction method.

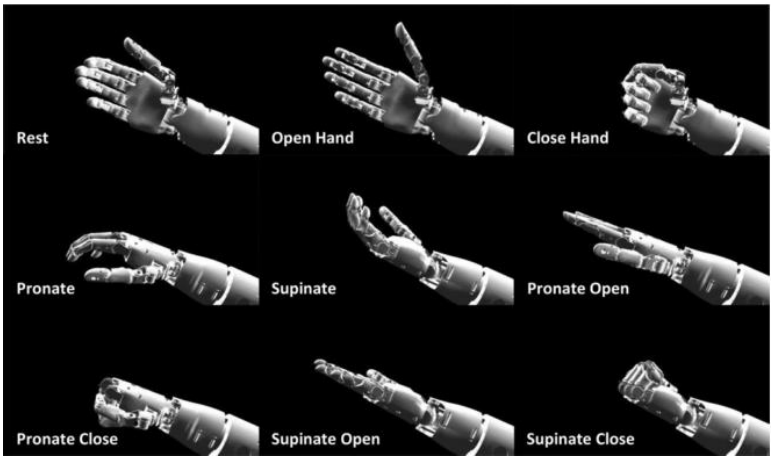


Figure 2-11 Virtual prosthesis generated by the musculoskeletal modeling software (MSMS). Nine gestures are demonstrated [20]**.**

Table 2-4 Mathematical representation of commonly used sEMG features [101]**.**

Feature Extraction	Mathematical Equation
Integrated EMG(IEMG)	$IEMG = \sum_{n=1}^{N} \lvert x_n \rvert$ Here N denotes the length of the signal and x_n represents the sEMG signal in a segment.
Mean Absolute Value (MAV)	$MAV = \frac{1}{N}\sum_{n=1}^{N} \lvert x_n \rvert$
Modified Mean Absolute Value 1 (MMAV1)	$MMAV1 = \frac{1}{N}\sum_{n=1}^{N} w_n \lvert x_n \rvert$ $w_n = \begin{cases} 1, & if\ 0.25N \le n \le 0.75N \\ 0.5, & otherwise \end{cases}$
Modified Mean Absolute Value 2 (MMAV2)	$MMAV2 = \frac{1}{N}\sum_{n=1}^{N} w_n \lvert x_n \rvert$ $w_n = \begin{cases} 1, & if\ 0.25N \le n \le 75N \\ \frac{4n}{n}, & if\ 0.25N \le n \\ \frac{4(n-N)}{n}, & if\ 0.75N \le n \end{cases}$
Simple Square Integral(SSI)	$SSI = \sum_{n=1}^{N} \lvert x_n \rvert \hat{}\, 2$
Variance of EMG (VAR)	$VAR = \frac{1}{N-1}\sum_{n=1}^{N} x_n \hat{}\, 2$
Root Mean Square (RMS)	$RMS = \sqrt{\frac{1}{N}\sum_{n=1}^{N} x_n \hat{}\, 2}$
Waveform Length (WL)	$WL = \sum_{n=1}^{N} \lvert x_{n+1} - x_n \rvert$
Willison Amplitude (WAMP)	$WAMP = \sum_{n=1}^{N} f\lvert x_{n+1} - x_n \rvert$ $f(x) = \begin{cases} 1, & if\ x \ge threshold \\ 0, & otherwise \end{cases}$
Log detector (LOG)	$LOG = e^{\frac{1}{N}\sum_{n=1}^{N} log\lvert x_n \rvert}$
Slope Sign Change (SSC)	$SSC = \sum_{n=2}^{N} f\left[(x_n - x_{n-1}) \times (x_n - x_{n+1})\right]$ $f(x) = \begin{cases} 1, & if\ x \ge threshold \\ 0, & otherwise \end{cases}$
Zero crossing (ZC)	$ZC = \sum_{n=1}^{N-1} [sgn(x_n \times x_{n+1}) \cap \lvert x_n - x_{n+1} \rvert \ge threshold]$ $sgn = \begin{cases} 1, & if\ x \ge threshold \\ 0, & otherwise \end{cases}$
Multi-scale amplitude modulation–frequency modulation (AM–FM)	$f(k) = \sum_{n=1}^{M} a_n(k) cos\emptyset_n(k)$ Here n = 1, 2,…M indexes the AM–FM components, a_n represents the nth instantaneous amplitude, and ϕ_n represents the nth instantaneous phase. Here, AM–FM components are extracted over a dyadic filter bank.

2.2.2 Methods for sEMG-force modeling

In recent decades, the sEMG-force relationship has been extensively investigated. The estimation of the force generation from muscle activations is not only gaining interest in the biomechanical studies but also more and more in clinical applications in which the information about the muscle force supports the physician's decisions on diagnosis and treatment [145]. The sEMG signal is measuring the degree of the activations in the skeletal muscles, henceforth; it is highly correlated to the muscle force. However, due to the difficulties to directly measure the task related muscle activities without introducing noise factors mentioned in section 2.3.1, an accurate sEMG-force relationship has been investigated by many researchers. Both Disselhorst-Klug [145] and Staudenmann [146] discussed the problems associated with sEMG-force estimation and proposed solutions. The prediction accuracy on the force amplitude is the main methodological issue. The stochastic nature of the sEMG signal which is a result of a series of constructive and destructive superimpositions causes the lack of prediction precisions. Novel methods such as multi-channel monopolar EMG and high-pass filtering or whitening of conventional bipolar EMG make the variable estimation easier and provide better estimation results. These methods are able to (1) reduce effects of phase cancellation, and (2) provide sufficient representation of the heterogeneous activities of motor units within a muscle. More importantly, highly accurate predictions of force are achievable even for minor force fluctuations that occur during an isometric and isotonic contraction. For dynamic contractions, the force estimations need to consider the muscle length and contraction velocities. Therefore, a valid force estimation requires sEMG amplitude prediction combined with modelling of the muscle contraction dynamics [146]. Concentric and eccentric contractions are two simple types of dynamic movements, but their sEMG-force relationships are different. If the same force applies, the myoelectric signal strength increased in concentric contraction compared to static isometric contraction, and it is lower in eccentric contraction. An example of elbow joint dynamic movement was a rhythmic flexion-extension [141] performed in a decelerated and accelerated manner by the biceps and triceps, as shown in Figure 2-12.

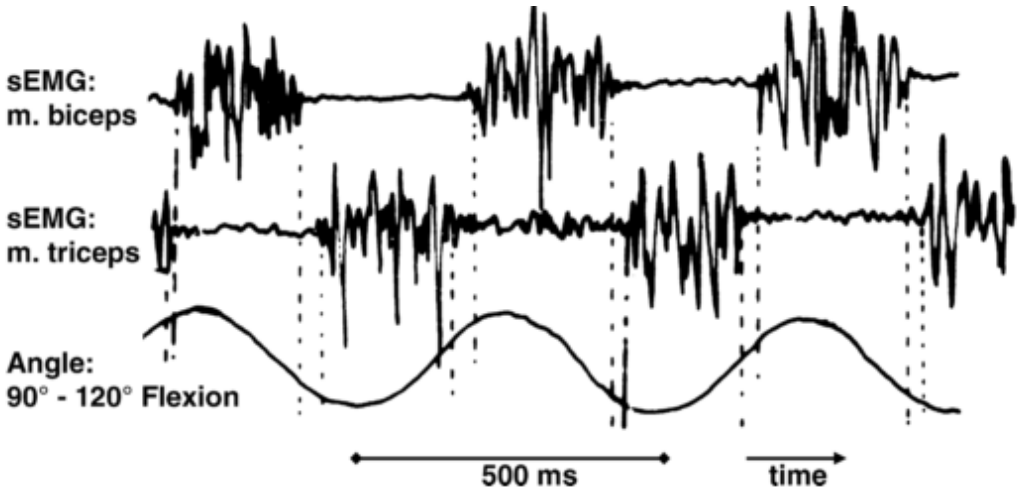


Figure 2-12 Elbow joint angle changes as a function of time and sEMG activations during a rhythmic flexion-extension [141]**.**

In the literature, to model the sEMG-force relationship, both implicit and explicit methods have been implemented. The implicit model estimates the complex relationships between myoelectric signals and dynamic forces without any explicit function descriptions, whereas the explicit model predicts the force based on the phenomenological muscular models. A number of researchers have contributed in both direction of this muscle force modelling.

Savelberg and Herzog [147] proposed using an Artificial Neural Network (ANN) with back-propagation algorithm to predict dynamic tendon forces from electromyography signals. Tendon force and EMG signals are recorded for cat walking. The cross-correlation coefficients between the predicted and actual force ranged from 0.72 to 0.98, which indicated that the ANN approach was a powerful technique to predict dynamic tendon force.

Loconsole *et al.* [91] proposed a sEMG-based method for on-line estimations of the torque and control of robot joints. A light Exoskeleton was used as assistance to patients in the execution of functional tasks such as reaching with the impaired arm. Myoelectric signals from five individual muscles of the shoulder and elbow were recorded. The Mean Absolute Value (MAV) feature was extracted and the joint torque of the exoskeleton robot was collected as input and target dataset. The Time Delayed Neural Network (TDNN) was applied to modelling the dynamic and non-linear relationships. This approach showed the possibility to support patient movements during therapy. However, the approach was used for motion in one plane and validated

on one subject. Thus, it requires considerations with more degree of freedom and subjects.

Jali *et al.* [148] used ANN to model sEMG-torque relations through arm rehabilitation device. The network composed of two layer feed-forward network and trained with back propagations. The experiment was conducted in one plane and the operator was asked to lift the dumbbell. The results indicated that the ANN can represent sEMG-torque relationship well for arm rehabilitation device control.

Apart from ANN, various machine learning algorithms have been implemented for sEMG characterizations. However, more papers are found in the field of sEMG-based classification for myoelectric control compares to sEMG-force regressions. In principle, the algorithm for solving a classification or a regression problem has little difference. Karlık [17] compared a number of machine learning algorithms based on different features, including Nearest Neighbour Classifier, Multi-Layer Perception with Back-propagation, Fuzzy Clustering Neural Network, Linear Discriminant Analysis, Artificial Neural Fuzzy Inference System, Learning Vector Quantization Neural Network and Support Vector Machine.

Some literatures focused on pose estimations rather than force predictions, but in principle the methodology is interchangeable. Kwon *et al.* [149] proposed using ANN to estimate human motions to facilitate natural cooperation and safety of the human within human-machine cooperation systems. sEMG and joint angular velocity were used as input and target training dataset. The experiment results indicated acceptable results in flexion-extension of the limb in the 2D plane with normalised root mean square error (NRMSE) < 0.15 and correlation coefficient (CC) >0.9 under non-contact condition. Zhang *et al.* [150] proposed a multiple input and single output autoregressive structure with exogenous input (ARX) model to represent the sEMG-pose relationship. Experiments were conducted on two subjects performing a single elbow flexion-extension movement, and the results showed improvements compared with previous studies with RMSE within 8.3% - 10.6%.

Different from the machine learning based implicit approach described above, the phenomenological based explicit model was implemented for human locomotion studies. Sartori *et al.* [151] demonstrated a comprehensive framework for an excitation primitive (XP)-driven musculoskeletal model. As shown in Figure 2-13, it contained

five components: (A) Musculotendon Kinematics; (B) Musculotendon Excitation-to-Activation; (C) Musculotendon Dynamics; (D) Moment Computation and (E) Model Calibration. After calibration the model operates in open-loop. The excitation primitives were mapped to 34 individual Musculotendon Units (MTU) and went through the muscular-tendon dynamics model and generated toque commands to the individual joints. The human musculoskeletal geometry was scaled to match the individual subject's anthropometry by using OpenSim software [152]. The MTU was modelled by the Hill-type muscle model [153], and the corresponding parameters such as maximum isometric muscle force, optimal fibre length, and maximum muscle contraction velocity can be found in [154], and other biomechanical parameters can be found in [30]. This approach was presented and validated with a comparison to the previous EMG-driven modelling approach in [155][156].

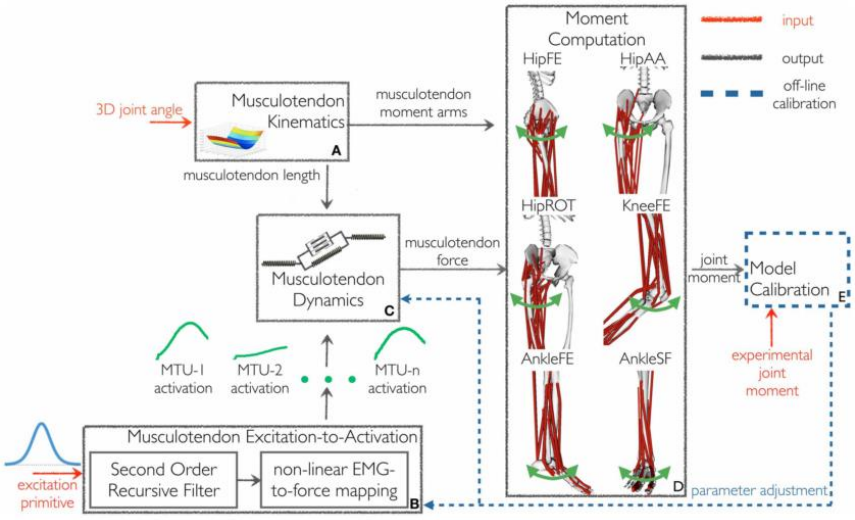


Figure 2-13 The schematic structure of the excitation primitive (XP)-driven musculoskeletal model [151]**.**

In this section, various methods for sEMG-force modeling have been reviewed. The machine learning based approach can be accurate after tuning the parameters and properly process the sEMG signal. It is relatively easy to implement but lack of explicit descriptive functions. The phenomenological based approach takes longer time to prepare but provides explicit explanations. The learning based method is selected in this research due to its promising force estimation capability and implementation simplicity.

2.3 Robot learning from demonstrations

Learning from Demonstration (LfD) is an attractive research topic for many robotics applications. It defines a mapping between world state and action. This mapping is actually a policy which enables a robot to select an action based upon its current perception of the world state. It is challenging to develop these policies by hand due to the task complexity, and as a result machine learning approaches have been implemented.

Within LfD, the teacher plays an important role to demonstrate examples for policy learning [35]. The examples are defined as sequences of state-action pairs that are recorded during teacher's demonstration of the desired robot behaviour [24]. This dataset is utilised by the LfD algorithm to derive policy that reproduces the demonstrated behaviour. Different from other approach such as Reinforcement Learning [157] in which the policy is learned from data acquired through exploration, the LfD approach derives a policy only in those states encountered, and for those corresponding actions taken, during the example executions [24].

In this section, the review of the LfD approach is focussed specifically on robotic applications. The problem is segmented into two fundamental phases: gathering the examples, and deriving a policy from such examples. Argall *et al*. [24] established a structure for concretely placing the relevant works that applies LfD within the community. In general, they proposed a categorical structure for aiding comparative assessments among applications.

Although LfD is widely used in robotics, this thesis considers it as an important method for skills capturing and encoding. As mentioned in the introduction, the conventional robot programming will soon become impossible as the complexity of the skilled based task increases. Also, the knowledge from the non-robotic expert is required for control policy designs. Therefore, LfD is a potential tool for fulfilling such demands in which the state-action examples are learned.

2.3.1 Challenges and problem definition

The traditional ways of learning a robotic control policy are through building a dynamic and mathematical model. But this approach depends heavily on the accuracy

of the world model. The model requires considerable expertise to develop, and approximation such as linearization is often introduced for computational tractability. However, the performance is often degraded because of this. Other approaches such as Reinforcement learning requires an agent to continuously interact with the environment and receive reward feedback given particular states. But exploration from scratch is often inapplicable for physical robot in the industrial environment. Furthermore, designing a reward function requires expert knowledge.

Due to these challenges, LfD offers many advantages for both leaner and teacher. Firstly, the formulations in LfD do not need expert knowledge of the domain dynamics, which removes the compromise of model simplifications. Secondly, it is more accessible for non-robotic expert (with knowledge of the process) to get involved in the design of the robot control policy. Lastly, it learns task specific constrains without initial explorations, which is suitable for practical implementations.

LfD is a subset of supervised learning. In supervised learning, the input data are labelled and the agent learns an approximation to the function which produces the input-target relationship. Within LfD, the training dataset composes the example executions of the task by teacher's demonstrations (Figure 2-14, top).

According to [24], the LfD is formally defined as follows. The state S and action A mapping in the world were defined by a probabilistic transition function $T(s'|s,a)$. The states were assumed not fully observable, henceforth, instead the learner had access to the observed state Z, through the mapping $M: S \rightarrow Z$. A policy $\pi: Z \rightarrow A$ selected the actions based on the observations of the world state. A single cycle of policy execution at time *t* was shown in the bottom of Figure 2-14.

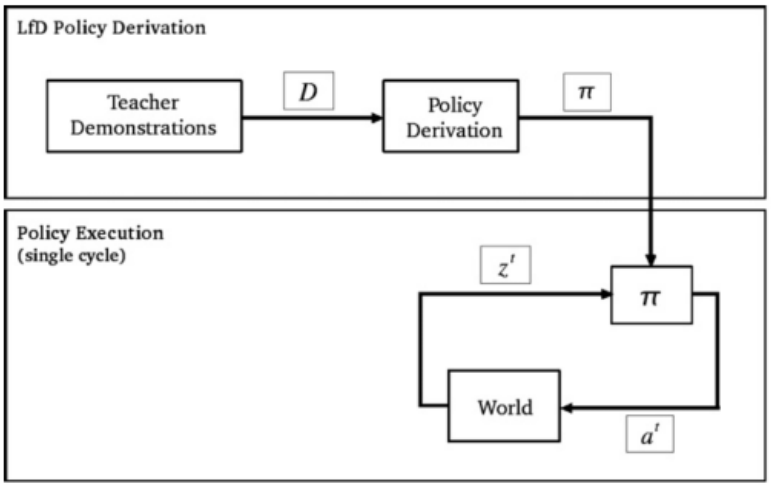


Figure 2-14 Control policy derivation and execution [24]**.**

The action set A ranges from low-level motions to high-level motions. In the simulation environment, the state might be fully transparent, but in other applications, the states are not fully transparent and observable. Therefore Z is the observations from the sensors and used for representation of the real world states S. The teachers demonstrations $d_j = \{(z_j^i, a_j^i)\} \in D$ formally defines k_j pairs of observations and actions where $z_j^i \in Z, a_j^i \in A, i = 0, \dots, k_j$. The set D is demonstrated to the learner, and the LfD enable the learner to select an action based on the current state.

Many researchers have reviewed the LfD by categorisations. For example in Billard *et al*. [158] and Schaal et al. [159], the discussions concentrated on who, what, when and how to imitate for a robot learner. Argall *et al.* [24] focused on the formulations and techniques required to implement LfD system. Other useful related survey was [160] which provided a foresight of integration research from those studying imitation in humans and other animals, and those studying computer software. The discussion involved a diverse interdisciplinary field including animal behaviour, artificial intelligence, computer science, neuroscience, primatology, and linguistics. Another review presented in the chapter "Robot Programming by Demonstration" [158] highlighted the techniques for LfD approach and provided a comprehensive historical review of LfD. The focus of this section is to illustrate how the dataset within the LfD is composed and the recent techniques to derive policies and their corresponding applications. For the remainder of this section, the key design decisions for an LfD system will be discussed in 2.3.2. Methods for gathering demonstration examples will be presented in 2.3.3, followed by the core techniques for policy derivation 2.3.4

2.3.2 Design choices

All LfD applications share some common aspects. One of them is that the teacher demonstrates a desired behaviour. Another is that the learner provided with those demonstrations, and from them derives control policies to reproduce the behaviour.

However, some design choices need to be considered when developing a new LfD system. For example, the choices of a discrete or continuous action representation are determined by the application. Other choices might be dependent on the designer's preference. But in general, these design choices have strong influence on how the learning problem is structured and solved. In this section, the most significant design choices are highlighted. A Peg-in-Hole task is presented as an example in which the robot is asked to (1) approach the hole (2) insert the peg and (3) extract the peg.

a) Demonstration approach

In the context of the teacher's demonstrations, two decisions must be made: the choice of the demonstration, and the choice of demonstration technique. These decisions are affected by factors such as the complexity of the robot and task. For instance, teleoperation is rarely used in humanoids with high degree of freedom, since the complexity of the motion is difficult to control through a joystick.

i. The choice of demonstrator

To date, most LfD works use the human demonstrators, although some techniques also suggested the use of robotic teacher, hand-made control policies and simulated planners [24]. The choices of the demonstrator further break down into (1) who controls the demonstration and (2) who executes the demonstration.

For instance, consider a robot learning a PiH task, as described above. One approach could be a robotic teacher approach, insert and extract the peg using its own body. In this case, a robot teacher controls the demonstration and its body executes the tasks. An alternative approach could be a human teacher tele-operate the robot learner through the task of PiH. In this case, a human teacher controls the demonstration, and a robot learner executes the task. For example, the human teachers demonstrate a collaborative manipulation task, namely lifting an object by teleoperation [161]. The robot follower tried to learn the collaborative manner produced by the human.

Kormushev *et al.* [162] taught the robot an ironing task and a door-opening task via human kinaesthetic teaching. Both the positioning and force profiles were learned and reproduced by a robot learner. From the above, the choice of demonstrator has a significant impact on the type of learning strategies that can be applied. Further discussion on the similarity between state and action spaces of the teacher and learner determines the types of the algorithm to process the data can be found in section 2.3.4.

ii. Demonstration techniques

The demonstration technique is the strategy for generating dataset to the learner. One option is to perform batch learning, in which the policy is derived only once all the demonstration sequences are gathered. Alternatively, an interactive approach allows the policy to be updated incrementally as new demonstration sequences become available. Billard *et al.* [158] introduced a practical guidance of the incremental implementation of the probabilistic approach. The advantage of this approach was that the model parameters can represent the learned dataset so the old data can be removed and the model only updated from the new data. A hybrid of batch and incremental learning strategy was also presented where the model was initiated with batch learning and further adapted to the new demonstrations. If the time domain feature was not as important as the spatial feature, it was important to attenuate the time line in each demonstration so that there was no time shift in the spatial domain. A Dynamic Time Warping (DWT) [163] is a distance based approach to align multiple time series data in multiple dimension. However, to enable such approach, a reference demonstration is required. According to Billard *et al.* [158], one can simply use the first demonstration as a reference signal or use the generalised representation from the Gaussian Mixture Regression (GMR) of a set of demonstrations.

b) Problem space continuity

It is important to consider the continuity of the state-action representation. In the PiH task, the environment can be represented by discrete state such as the chamfer crossing, one point contact, two point contact and line contact [164]. Alternatively, a continuous state could be represented by 3D positions of the robot's end effector and the hole centre. Similarly, the discrete or continuous representations could be applied for the robot's actions.

In designing a domain, many factors affect the continuity of the problem space, such as the behaviour of interest, the set of the available actions and whether the world is simulated or real. The selection of the various policy derivation techniques heavily relied on the continuity of the given problem.

Different from the state-action continuity problem, LfD can be applied at a variety of action control level [24]: low-level actions for motion control, action primitives control, and complex behavioural actions for high-level control. The different level of control actions can be both discrete and continuous. An LfD framework is applicable in any of this control level therefore the continuity is a major determinant on the selected method.

c) Policy derivation

The main consideration of the policy derivations are (1) the general technique used to derive the policy and (2) whether the performance can improve beyond the teacher demonstration. In this section, according to [24], a short summary is provided for the three core approaches to policy derivation, which were defined as mapping function, system model, and plans (as shown in Figure 2-15):

- Mapping function: A direct approximation function mapping (π) from the robot's state observations (Z) to actions (A) from dataset $D = [(z', a')]$.
- System model: A world dynamics ($T(s'|s, a$)) is determined by the demonstration data, and possibly with a reward function ($R(s)$). A policy is then derived.
- Plans: A sequence of actions is planned by learning the rules that associate with the pre-and post- conditions ($L(\{preC, postC\}|a)$), and possibly a sparsified state dynamic model ($T(s'|s, a)$).

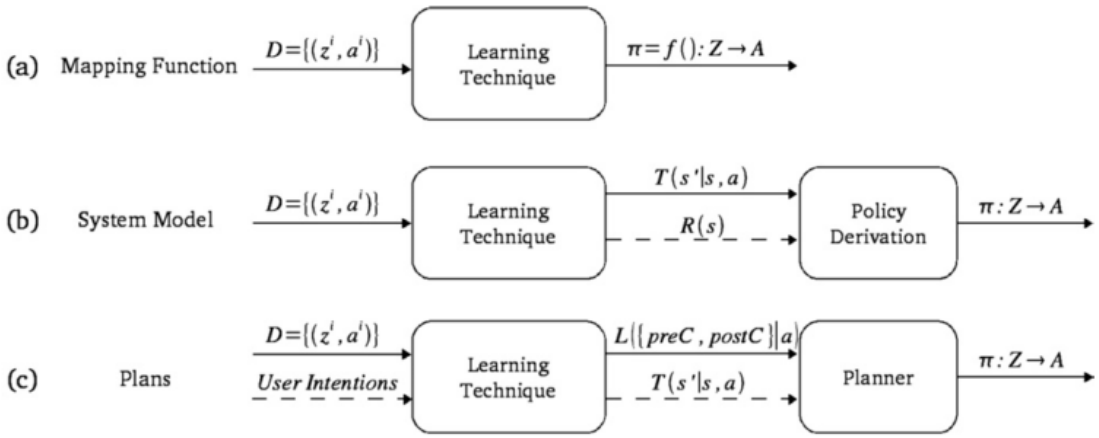


Figure 2-15 Policy derivation methods. (a) An approximation to the state-action; (b) dynamic model of system and (c) a plan of sequenced actions [24]**.**

Considering the PiH example, a mapping function approach can be used derive the policy, for example the 3D orientation of the end effector, to an action which guide the robot to align the peg with the hole direction. In the system model approach, a state transition model is learned first, for example taking the peg in the approaching state with the robot holding action results in the chamfer cross state. Using the model, the learner knows the best action to take when given a state. Finally, in the planning approach, the pre- and post-conditions of the executions are learned from the demonstrations. For instance, the insertion action needs the chamfer cross condition and peg inserted condition. A planner uses this learned model to generate a sequence of actions that ends with robot goal state.

d) Data limitations

The training samples from the demonstrations are intrinsically limited by the performers. In many cases, for example in the industrial manipulations, it is possible that the experienced manual worker performs in a suboptimal way when compared with the ability of the learner. For example, a human teacher cannot be physically as quick or accurate as a robot learner. Therefore, the performance of the learner is also limited by the policies derived from those demonstrations. Approaches such as reinforcement learning may be used to improve the existing model beyond what was provided by the demonstration dataset.

2.3.3 Correspondence issue

Various techniques for executing and recording demonstrations are discussed in this section. As mentioned before, the LfD dataset is the state-action pairs recorded

during teacher executions of the desired behaviour. However, how they are recorded, and what platform the teacher uses for the execution, varies greatly across applications. Methods range from the sensor on the robot learner body which are passively tele-operated by the teacher, to an external camera recoding a human teacher when he/she executes the behaviour with his own body.

In an ideal situation, to successfully implement LfD, the state and actions in the dataset should be usable by the learner. This requires the state and action from the teacher executions to be mapped directly to the leaner. However, in reality, it is almost impossible to have that direct mapping, as the learner and the teachers are likely to differ in their sensing or mechanics. For instance, a robot's camera is not working the same as the human eye, nor will its gripper grasp the objects in the same manner as a human hand. The challenges arise from these differences are referred to as correspondence issues [165].

The correspondence issues in the LfD are closely related to the two mapping in between the teacher and the learner: Recording Mapping and Embodiment Mapping The terminology used in this section is referred to [24].

The recording mapping maps the teacher executions to the recorded executions. An identity $I(z, a)$ means the data recorded during teacher demonstrations are directly recorded in the dataset. Otherwise, some record mapping function will be required to encode teacher information. The embodiment mapping maps the recorded dataset to the leaner. An identity $I(z, a)$ means the recorded state-action pairs are exactly those that the learner would observe/execute. The embodiment mapping is important and it should be accurate as the real robot needs to physically execute what teacher showed.

Recalling the PiH example, a human teacher uses his own body to demonstrate the insertion technique, and a camera to record his motion. The human joint angle information needs to be extracted from the image data, henceforth, a recording mapping is needed. Furthermore, the physical embodiment of the human teacher is different from that of the robot; therefore, the actions from both embodiments need a mapping function as well.

To avoid the terminology confusions, the LfD framework can be split into two major categories: Demonstration and Imitation due to the differences in the mapping functions.

In the demonstration case, there is no embodiment mapping issue at all which means what teacher has shown is exactly what happens on the robot. Two approaches are common in this case: Teleoperation and Shadowing. In teleoperation, the teacher operates the robot learner platform and uses the robot's sensor to record the examples. This method provides the most direct way for information transfer from teacher to learner. A joystick is the most commonly used teach-in device for teleoperation. It has been applied in many fields of interest, including flying a robotic helicopter [166], object grasping [167][168], robotic arm assembly tasks[169], and navigation [170]. Haptic device takes advantage of the joystick with additional force feedbacks. Calinon et al. [161] they used a haptic device to teach robot collaborative manipulation tasks. The proposed approach was generative and could be used to retrieve the tasks by reproducing the dynamics of the task, namely lifting an object and adapting to the human user's hand motion. Soediono [12] taught the robot a ball balancing task, in which the human teacher demonstrated his skill to guide the ball to a designated hole by using haptic device. The robot was asked to reproduce the performance in a blind way by only relying on the force-torque sensing. Alternatively, kinaesthetic was another teleoperation approach wherein the human demonstrated the task by holding robot end effector. Kormushev *et al.* [162] used both haptic and kinaesthetic based approach to teach a robot an ironing task. Tang [171] used the kinaesthetic approach to teach an industrial robot a PiH task. Abu-Dakka et al. [37] used the kinaesthetic based LfD approach combined with exception strategies to teach a robot a PiH task. Various types and sizes of the hole and pegs were tested within the framework. In [172], as shown in Figure 2-16, a robot was taught to hit a ball with reproduction of a drive stroke (top) and topspin stroke (bottom), and the HOAP-3 humanoid robot was taught to feed a robotic doll.

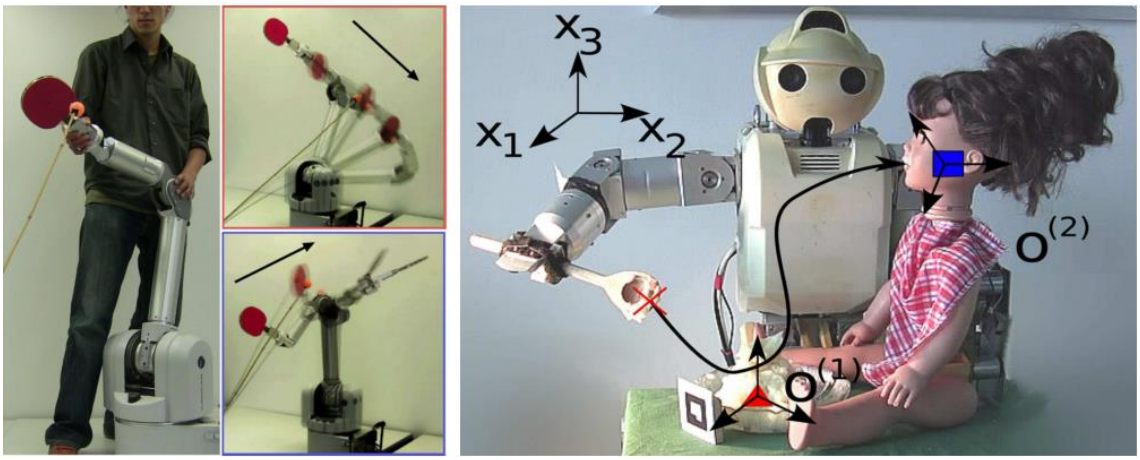


Figure 2-16 An example of kinesthetic learning [172].

In shadowing, the robot learner mimics the teacher's demonstrations while recoding using its own sensors. The record mapping is not direct, because the state-action during the demonstration is not recorded. Instead, the robot records its own mimicking execution, so the teacher's state-action examples are indirectly encoded within the dataset. Different from passively tele-operated by the teacher, this method requires the robot to actively shadow the teacher. Applications such as having a robot follow an identical platform robot teacher through a maze [173], follow a human teacher passing sequences of colored markers [174] and to mimic the trajectories determined from observations of human teacher executions [175].

In imitation approach, the embodiment issue exists and indicates the presence of the mapping function to map the demonstrations recorded to the learner's state-actions. Depending on whether the recording is identical or not, this approach can be further divided into two types: sensor on teacher, and external observation.

In the sensor on teacher approach, sensors are installed on the human body, and all the collected demonstrations are directly relating to the recorded executions. This alleviates one potential source of the correspondence problem. The advantage of this approach is that the teacher provides accurate measurements of the state-action pairs. However, it also requires specialized sensors, such as motion tracking suites or controlled environment with fixed cameras. A review of those tracking techniques can be found in section 2.1.

When working with humanoid or anthropomorphic robots, the human teachers commonly use their own bodies to execute the task by using wearable sensors. In

[176], drumming patterns recorded from a human joint angles were demonstrated to a 30-DoF humanoid robot, and later in Nakanishi *et al*. [177], walking patterns were studied to design a controller for natural human-like locomotion. Calinon *et al.* [178] taught a humanoid robot various gestures by wearing the motion sensor on his body (left), and refining the movements by kinesthetic teaching (right) during robot reproduction, as shown in Figure 2-17. Robot benefits from both human demonstrations and its own kinematic capabilities by using this teaching approach. Another approach by Aleotti and Caselli [179], a simulated human was controlled by the human teacher wearing sensors on his body, and then the simulated movements were mapped to a real robot arm.

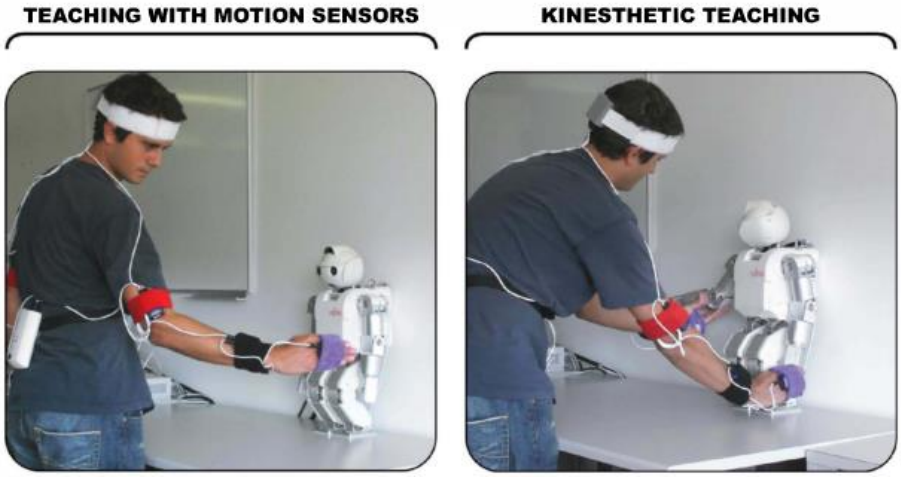


Figure 2-17 Human teacher teaches a humanoid robot by both sensor on body (left) and kinesthetic teaching (right) approaches. [178]

Different from sensor on body approach, the external observation approach relies on data recoded by sensors located externally to the executing platform, which means that a recording mapping exist. Since the actual state-action pairs from the human demonstration are not directly recorded, they must be inferred from the recorded data and extracted the useful state-actions for the robot learners. Compared to the sensor on body approach, this approach is more general and is not limited by the overhead of specialised sensors and settings.

Visual based tracking is the predominant technique for this external observation approach. In the early work [180], stereo vision was used to teach a robotic arm pole balancing task. Nowadays, marker-based approach as discussed in section 2.1.1 is the most popular technique. Marker based technique was applied to teach human motion and manipulation tasks [181][182]. In Pollard and Hodgins [183], the motion

primitives and language expressions were integrated and learned for a humanoid robot. Both natural language module and motion primitives were represented in probabilistic model, so that a robot could infer the commands and plan his move by reading a sentence. Tang *et al.* [171] installed reflective marker on the object of holding, and taught an industrial robot a PiH manipulation task. Calinon and Billard [184] used a color-based stereoscopic vision system to track the 3D position of a marker placed on the teacher's hand.

Other methods such as background subtraction was used to extract teacher motion from images [185]. Skeleton models produced by the depth cameras were also useful to extract the state-action pairs from a steam of images. The external sensing approach can be also used jointly with other systems such as force-sensing gloves for grasping teaching [186].

2.3.4 Policy derivations

The techniques to derive a policy from the demonstration data were discussed in this section. As mentioned from the beginning, three core approaches composed the LfD framework. It involved simply learning an approximation to the state-action mapping, or learning a model of the world dynamics. Alternatively, the planner can generate a sequence of action after learning a model of action pre- and pose- conditions. In all these learning techniques, it is desirable to spend less training time with minimal parameters tuning and require few training examples.

a) Mapping function

The mapping function approach aims to calculate a function that approximates the state to action mapping for the demonstrated behaviour. The robot learner is able to generalise across all the training samples and reproduce the underlying control policy, which is usually unknown beforehand.

Many factors have influences on the details of the function approximations. These include whether the state or the action are continuous or discrete, whether the approximation function takes the data from the time of execution or from prior the time of execution, whether the learned dataset can be discarded or not, and whether the algorithm is online or offline.

In general, this mapping approximation techniques fall into two categories: classification and regression, depending on whether the prediction outputs are continuous or discrete.

i. Classification

Classification approach groups the similar input variables and classifies them into discrete classes. The input of the classifier is the state observations, and the outputs are the discrete robot actions. A summary of classification methods are applied at three action level (low-level actions, action primitives, and complex behaviours) is discussed in this section.

In the low-level actions, the basic commands include moving-forward or turning. Gaussian Mixture Models (GMMs) was used to control a car within a simulated driving domain using [187]. Bayesian network [188] and k-Nearest Neighbours (kNN) classifiers [189] were used to learn obstacle avoidance and navigation behaviours.

When motion primitives are recognized from the state, they are then composed or sequenced together to become a complete task. For example, Pook and Ballard [28], taught a robot egg flipping manipulation task. They classified primitive membership using kNN and then recognized each primitive from the demonstrated task via Hidden Markov Models (HMMs). HMMs have been applied in [190][191] to teach robot a basic assembly task and motor-skill tasks by identifying and generalizing upon the intention of the user.

Similar approaches have been applied on the high-level behaviours. The behaviours themselves are generally either handcrafted or learned prior to task learning. This means the learner knows what it is expected. Rybski and Voyles [192] represented gestures into eigenvectors, and within this framework, HMMs was used to classify the demonstrations into gestures for a box sorting task with a Pioneer robot. Lockerd and Breazeal [193] used Bayesian likelihood method to select actions for a humanoid robot in a button pressing task, and Support Vector Machine (SVM) was used for a robotic ball sorting task [194].

ii. Regression

Regression approaches map the state observations to the continuous action spaces, and most approaches to state-action modelling estimate a time dependent model of the trajectories. Modelling methods such as exploiting variants along the concept of spline decomposition [179] [195] or through an explicit encoding of the time-space dependencies [196], were effective and precise in the description of the actual trajectory. However, the explicit time dependency of those models requires additional time alignment process to handle spatial and temporal perturbations. As an alternative, modelling the intrinsic dynamics of motion was considered in other approaches [197] [198] [199]. These approaches had benefit that the system did not depend only on an explicit time variable and was able to reproduce actions with similar dynamics in unexploited areas which not covered by training. The state of the art approaches proposed so far are Hidden Markov Model (HMM) [172], Gaussian Mixture Regression (GMR) [200], Locally Weighted Regression (LWR) [201], Gaussian Process Regression (GPR) [202], and Dynamic Movement Primitive (DMP) [203].

GMR: Gaussian Mixture Regression, as introduced in [200], uses time as an explicit input variable, and the demonstrations are first aligned though DTW. Then, the distribution of temporal and spatial variables $\{t, x, \dot{x}\}$ is encoded using a Gaussian Mixture Model (GMM). At each time step, by given the current time step, a desired position $\hat{x}$ and a desired velocity $\hat{\dot{x}}$ are retrieved by estimating $P(x, \dot{x}|t)$. In [204], a generalization of the demonstrated trajectories is calculated by using GMR for a chess-moving and a cup grasping tasks. The optimal trajectory is generated by additionally considering the task and robot kinematic constrains. This method can reproduce smooth trajectories but require alignment through the demonstrations. Calinon *et al.* [205] extended the approach in [204] to a more generic procedure handling simultaneously constraints in task space and joint space by combining directly the probabilistic representation of the task constraints with a simple Jacobian-based inverse kinematics solutions. In order to retrieve the dynamics of the task, some modifications are needed. Calinon *et al*. [161] extended the GMR by its analogous HMM. The weighting function was no longer calculated based on position information only, but was replaced by the forward variable corresponding to the probability of partially observations of being in the current time step and current state.

LWR: Locally Weighted Regression is a memory based probabilistic approach [201]. It can estimate the desired position $\hat{x}$ and desired velocity $\hat{\dot{x}}$ at each time step. The influence of each data point is estimated by assuming a Gaussian kernel with fixed diagonal covariance matrix centred at the current point of interest.

LWPR: Locally Weighted Projection Regression is an incremental regression algorithm that performs piecewise linear function approximation [201]. The algorithm does not require the storage of the training data and has proved to be efficient in solving high dimension problem. By detecting locally redundant or irrelevant input dimensions, the dimension of the input data is locally reduced by finding the local projections though Partial Least Squares (PLS) regression.

DMP: Dynamic Movement Primitives approach was originally proposed by Ijspeert et al in [203]. The method allows a target to be tracked by modulating a set of mass-spring-damper systems. The velocity is guaranteed to vanish at the end of the movement. A phase variable acts as a decay term to make sure the system is asymptotically converged to the end point.

b) System models

The system model approach to learn a policy uses a state transition model of the world, $\mathrm{T}(\mathrm{s}'|\mathrm{s},\mathrm{a})$, and from this derive the policy π: $Z \rightarrow A$. This approach is typically formulated and structured within the Reinforcement Learning (RL) approach [24]. $\mathrm{T}(\mathrm{s}'|\mathrm{s},\mathrm{a})$ is derived from the demonstration data and additionally explorations with reward functions $R(s)$, which is either learned or defined by the user. The goal of RL is to maximize the cumulative reward over time. The accumulated future reward under the current policy π given the current state s and action a is calculated based on the state value function *V(s)*. The Value function maybe updated by using the Bellman equation in the following form:

$$V^{\pi}(s) = \int_a \pi(s,a) \int_s T(s'|s,a)[r(s) + \gamma V^{\pi}(s')] \text{……..(2-3)}$$

Where $V^{\pi}(s)$ is the state value under policy π and given state s, γ is the discount factor for the future rewards. Unlike the function approximation approach, RL approach does not generalize state and every state must be presented with demonstrations. A comprehensive review of the RL approach can be found in [157]. The design of the reward function is critical to implement RL in the practice to prevent

the agent from extensive explorations. Both hand engineered and learned approaches have been proposed in the literature. Kober and Peter [190] proposed a policy learning approach by weighting exploration with the returns for the motor primitives (PoWER) in swing-up and ball-in-a-cup applications. During the policy update, the key is to calculate the policy derivations. Peters and Schaal [207] showed various popular policy derivation approaches in the robotics applications.

c) Plans

Plan is an alternative to the mapping function approach in which the states are directly mapped to actions. The policy is represented as a sequence of the actions that lead from the initial state to the final goal state. Actions are often defined in terms of pre-conditions, in which the state must be demonstrated before the action can be performed, and post-condition, in which the state is the result from the action's execution. Different from the other LfD approach, the plan is not only relying on the demonstrated samples but also depending on additional information in the forms of annotations or intentions from the teacher [24].

In all, the robot learning from demonstration methods was reviewed in this section. Among the candidate choices, the mapping function techniques, which use the HMM based approach, are the most suitable to the context of this thesis.

2.4 Summary and research gaps identification

In this chapter, various techniques have been reviewed for human motion tracking, sEMG-force modelling, and robot learning from demonstrations. In human motion tracking, advanced approaches including inertial and marker-based tracking are widely used. However, because they are both restricted to specific working conditions such as "line-of-sight" and drifts, it is preferable to have a hybrid system for human motion tracking.

The tactile and haptic force feedbacks are required to be collected alongside with the motion data. To remove the bulky Force Torque sensor, various sEMG-force modelling approaches have been reviewed. In fact, the sEMG signals are noisy which means the identification of the noise source and filtering strategies are particularly important. The method needs to have good performance in time localisation property and maintain the most relevant information. Among the choices, the Wavelet Transform is selected as a promising filtering technique. Two different approaches

(phenomenological modelling approach and learning based approach), have been review for building the relationship between the sEMG and the contact forces. The phenomenological modelling approach provides the explicit musculoskeletal model but takes longer time to prepare the parameters needed. The learning based approach is quick to prepare and the literatures indicates a good prediction performance in many application scenarios. Henceforth, the Time Delayed Neural Network which considers the time series natural of the dataset has been chosen as the modelling method.

The state-action examples which enable the learning of the human demonstrations need to be prepared from the previous procedures. In order to extract the skills from the demonstrated task and reproduce them on a robot learner, various policy derivation approaches have been reviewed. The case studies in this thesis belong to the category of solving regression problem in the mapping function. A HMM based algorithm is selected to encode and reproduce the dynamics of the manipulation tasks due to its generalisation capability and learning efficiency (easy to implement and quick to learn). The limitations of these LfD approaches are actually the limitation of the human demonstrations, in which situations including undemonstrated state, poor quality data and suboptimal or ambiguous demonstrations might occur. Therefore, experiments need to be properly designed to alleviate these issues followed with extensive result discussions.

From the above, the research gaps are identified as follow:

- The existing motion tracking system had either 'line-of-sight' or drifting problem in free space motion tracking for long term use. A hybrid method is needed to produce reliable (±3˚) tracking performance.
- The existing sEMG-modelling methods heavily focussed on gait analysis, whereas fewer studies are focusing on forearm manipulation tasks. In addition, no comparable results had been reported for modelling the sEMG-force relationship for industrial manipulations, therefore, investigations on how feasible and accurate to use the model free techniques to build such relationships are needed and evaluated with case studies.
- The existing control policy for robotic applications requires interaction with the physical world; however it is not practical to learn the policy from scratch through some random trial and errors. Therefore, it is essential to capture the human skills

and derive the control policy as a seed or initialisation to guide the robot to explore and learn better policy.

These identified research gaps are systematically addressed in the thesis using two case studies, namely peg-in-hole and beater winding processes.

Chapter 3. A Robust Hybrid VICON and IMU System for Tracking Human Forearm Motions

3.1 Introduction

A robust human motion tracking system is important for this research. This allows the subject to move in the working area with less constraints and deliver the skills in a more natural way. This chapter aims to address the research objective one to develop a wearable system that reliably tracks human motions for an appropriate length of demonstration time. The learning from demonstration framework can benefit from this since it reduces the errors introduced from the sensor measurements. To achieve this, the state of the art motion tracking system is first evaluated then a robust human forearm tracking system proposed and validated.

Tracking and understanding a static body posture of a human operator can be easily achieved and transferred to a machine if the kinematic information from body tracking is available [80]. However, compared to understanding static body postures, dynamic motion data can generate even more valuable knowledge for a machine/robot. These motion data can teach the robot the complexity and stochastic nature of the variations in the task if they are properly tracked. In order to track human body motions, various measuring techniques have been proposed in the literature. The user for instance wears mechanical trackers made from rigid or flexible goniometers. These trackers directly measure the joint angles of the wearer. However, installation of these body-based linkages is not trivial and requires extra efforts from the user to get used to it. It is almost impossible to track the full body motions in multiple degrees of freedom as a result of these constraints [73]. As alternatives, the vision-based and inertial based tracking systems are widely used for analysing human motions. However, both systems have limitations. The vision-based camera system cannot track objects while the line-of-sight is occluded. The inertial-based system suffers from signal drifts due to the ambient magnetic field.

To better learn from human demonstrations, an accurate and reliable (as suggested in [208]) dynamic tracking of the human movements is required. In the literature, there is a lack of systematic methods to evaluate the performances of the Vicon and IMUs for Mocap and to fuse their data to compensate for their respective

disadvantages. Therefore, in this chapter, I have evaluated the Vicon system with an accurate tri-axis Coordinate Measuring Machine (CMM); then the Vicon system was used for evaluation of the IMU measurements. Three IMUs were used to track the human forearm with following segments: upper arm IMU, lower arm IMU and hand IMU. An adaptive method to align the IMU body frame with the Vicon object frames without fixing them on a reference platform (i.e. tripod) is also used in this chapter to achieve a fully wearable system. In order to take advantage of both systems, a novel method is proposed to use the Vicon system to periodically recalibrate the initial frame of the IMUs and use IMUs to fill in missing observations during any occlusions of the Vicon. This chapter provides a systematic way to evaluate and a methodology to merge the Vicon and IMU systems to achieve a reliable (±2˚) human forearm motion tracking.

In this chapter, firstly, the 2-camera Vicon system is evaluated by using a CMM. Secondly, the IMU and the Vicon body frames transformations were estimated. Thirdly, the IMUs were evaluated by using Vicon. In the end, a PiH and Pick-and-Place tasks were used to evaluate the proposed compensation approach.

3.2 Methodology

To enable reliable and accurate dynamic motion tracking suitable for learning from demonstration of industry manipulation tasks, it is important to evaluate the tracking devices and to characterise their performances. The evaluation of the Vicon system requires a more accurate positioning system moving in multiple axes with the markers installed in clearly visible and stable positions. A CMM with tri-axial movements is appropriate for this requirement. The evaluation of the IMU requires a portable and lightweight tracking device without measurement drift. The Vicon system generates accurate pose measurements (±0.5mm) once calibrated; therefore, it is appropriate to evaluate the IMUs' performance. In this section, a framework for the evaluation and orientation compensation of the combined Vicon-IMUs tracking system will be presented. As shown in Figure 3-1, the data comes from three sources: a CMM, the Vicon cameras, and the IMUs. The CMM is used to evaluate the x, y, and z position data from the Vicon. Then the Vicon was used to evaluate the orientation data from the IMUs. The measurements from the IMUs were used to compensate the pose observed from the Vicon system. The compensated measurements can be used for further anatomical calibration on human (not the main topic of this work) [209]. The accuracy depends on the human poses and is affected by the skin movements. The following sub-

sections discuss this methodology in the following order: i) Vicon-CMM evaluation; ii) Vicon-IMU body frame alignment; iii) Vicon-IMU evaluation; and iv) Orientation compensation.

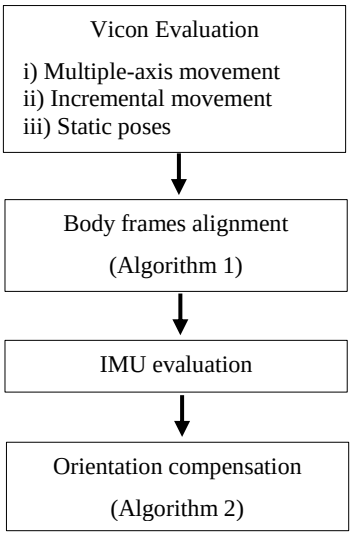


Figure 3-1 The framework of the Vicon-IMUs evaluation and joint states compensation.

3.2.1 Experiment setup

The devices used in this work are the two Vicon Bonita cameras with Tracker v2 software, an industrial CMM with an accuracy of 2.1μm, and three IMUs from X-sense and Thalmic Labs. The Vicon system is carefully calibrated using the active calibration T-wand with 8000 frames collected. The sampling rate of the Vicon and the IMUs are 100 Hz and 60 Hz respectively. The data is streamed in a synchronised way by using ROS approximate time synchroniser.

a) Experiment setup for Vicon-CMM evaluation

The experiment setup for x, y, z positioning evaluations of the Vicon are shown in Figure 3-2 where two Vicon cameras were symmetrically fixed above the work volume, and the working volumes of the CMM are drawn as well. The maximum single axis motion of the CMM is 1000 mm. Here, a 1000×700×600 mm cube was defined as the evaluation volume. Starting from near camera side to the far end, 5 planes (*P1, P2, P3, P4, P5*) are sliced at 1mm, 10mm, 100mm, 500mm and 1000mm. The motions are either on single axis (x-axis), in plane (x, y axes) or 3 dimensional (x, y, z axes*).* In each case, the CMM moves from *P1* to *P5* and the data was rcorded 5 times (*N=5*) at each plane. The markers are firmly fixed at the end of probing head. No alignment of the probing head and the marker plates are needed for position evaluation, since only

the relative moving distance will be compared. The reference frames are the starting frames of both CMM and Vicon system. For single axis motion, the CMM only moves 1000 mm at once from point A to point B. For two axes motion, the CMM moves 1mm, 10mm, 100mm, 500mm and 1000mm from point E to point F. For three axes motion, the CMM moves 1mm, 10mm, 100mm, 500mm and 1000mm from point C to point D. The mean error ($\sum_{i=1}^{N} error/N$) and standard deviation (SD) of the error are used to evaluate the result. The uncertainty can be further evaluated using ($\frac{SD}{sqrt(N)}, for\ N < 100$). Note, the lighting condion and marker configuraton may have impact on the result. The lighting condition will either improve or decrease the calibration accuracy. The marker configuration will make some poses visible and some poses not visible to the cameras. The setup for Vicon-CMM evaluation avoids this problem by having a consistent lighting condition and marker configuration which always visible to the cameras. However, to make a more robust evaluation, further disuccison on these issues need to be addressed in the future work (section 7.3.1).

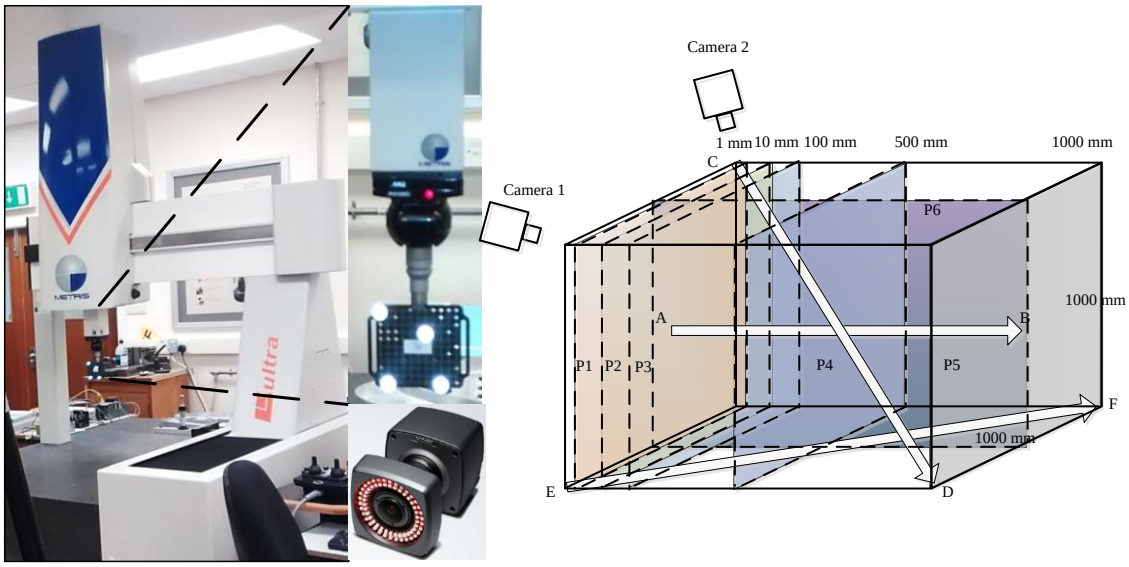


Figure 3-2 Experiment setup for Vicon-CMM evaluation. Left: the actual setup. Right: a schematic plot and annotation of the setup.

b) Experiment setup for Vicon-IMU evaluation

The experiment setup for Vicon-IMUs evaluation is shown in Figure 3-3. Three markers with asymmetric arrangement were installed on the Vicon frame which had been rigidly attached to the IMU body. To evaluate the IMUs on 3 dimensional space, the operator moved both frames in a 3 dimension volume. In the first step, for each IMU, an incremental number of the data samples have been used for the IMU and

Vicon body frame transformation estimation. 10 discrete numbers are selected: 10, 15, 20, 25, 30, 35, 40, 45, 50 and 55. 5 trials have been repeated for each case. In the second step, a sequence of movements that contain left-right waving, up-down tilting, and a “8” shape trajectory is used to emulate multiple axes rotations. This sequence was repeated 3 times and recorded for 300s (5min) for IMUs evaluations.

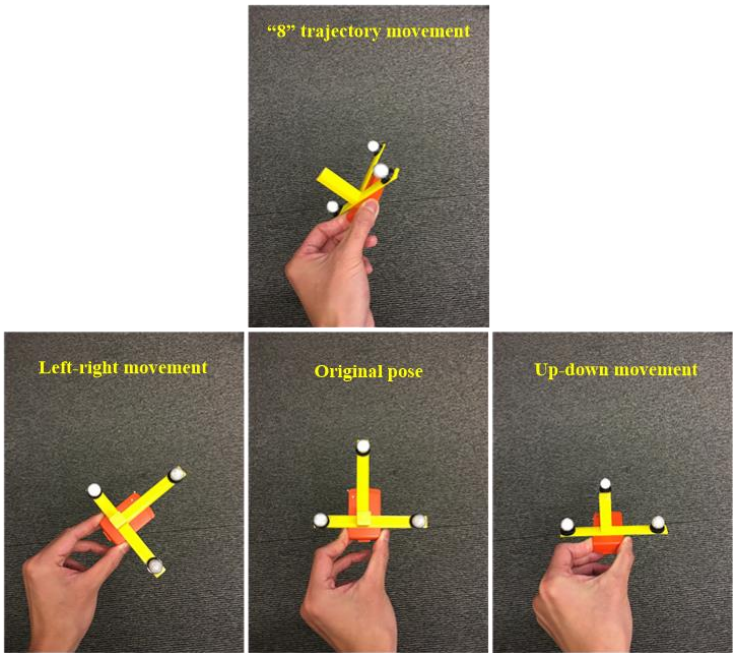


Figure 3-3 Experiment setup for Vicon-IMUs evaluation

c) Experiment setup for compensation method evaluation

Finally, two simply case studies were used to evaluate the IMU-Vicon system. One is a peg insertion (PiH) task, the other is a pick-and-place task (PnP) task. Four subjects were asked to wear the devices as shown in Figure 3-5 and repeated the trial 5 times. They are all male subjects who are researchers in the Intelligent Automation lab aged 25~30 years, the average height is 172cm and the weight is 75kg. They all have engineering degree and manufacturing automation expertise. Each trial contains 20 repetions of the PiH or PnP movements. The Vicon data was used as a reference signal. The proposed Vicon-IMU compensation method was applied to the collected trial data (Vicon and IMU). The Vicon data was deliberately corrupted with flipping frames and occlusions. The index of the flipping frames were randomly generated and covered 5% of the dataset. The occlusions were generated by evenly splitting the dataset into 4 subsets and masking 100 data points starting from each splitting node. Artificially

introducing flipping and occlusions allowed the proposed method to be evaluated against a known ground truth, which is the unaltered data.

3.2.2 Vicon-CMM evaluation

In general, the Vicon tracking system has an accuracy of less than 0.5 mm for positional tracking and ±3° for orientation tracking [210]. But this value only gives the overall performance of the system regardless of the working volume and motion. The question considered here is whether more accurate tracking can be achieved within a constrained movement and workspace with minimal cameras. This would be very interesting for robotic assembly tasks where a little improvement of the accuracy and stability of the positioning system will increase the success rate. This is the motivation of this evaluation. However, a more systematic evaluation can use the ratio between the standard deviation of the position error and the length of the cuboid main diagonal. But the main focus in this section is to identify a working volume which more accurate than the claimed accuracy. Here, three situations were considered:

- The static pose where the CMM stopped at several different locations in its workspace and the measurements were recorded statically for a short interval.
- Multi-axis movement where the CMM moved in x, y, z-axis simultaneously. This is to simulate the manipulation task i.e. an assembly task is likely to be multi-axis.
- An incremental movement where the CMM moved further away from the camera to explore the full working range.

The potential improvement of the Vicon system within a constrained working volume was evaluated by using the mean and the standard deviation of the error. If the mean error is less than 0.5mm and the standard deviation is small (<0.05mm), this means that the Vicon system can be even more accurate in that specific volume.

3.2.3 Body frame alignment and Vicon-IMU evaluation

The reference frames of the IMUs need to be aligned with the Vicon system to allow their measurements to be combined. To achieve this, each IMU has been equipped with four Vicon markers as shown in Figure 3-4. The markers allow the position and orientation of the IMU to be tracked by the Vicon system once the

coordinates of the IMU reference frame has been aligned with the world coordinate (GV) of the Vicon system. All frames used are right-handed. The centre of the coordinate (VB) is calculated based on the positions of the markers. The four-marker arrangement used is not unique; the rule of thumb is to make sure asymmetric geometric arrangements. One example can be found in [47]. The IMU measurements were derived from its inertial frame (GS). Its z axis is defined by the magnetometer. The IMU body frame (SB) depends on the location of the on-board chips. It should be aligned with the IMU case, but minor misalignments might exist. Therefore, it is almost impossible to manually align the body frame calculated by the Vicon (VB) and the body frame of the IMU (SB). A calibration method would be needed to fine tune the alignment.

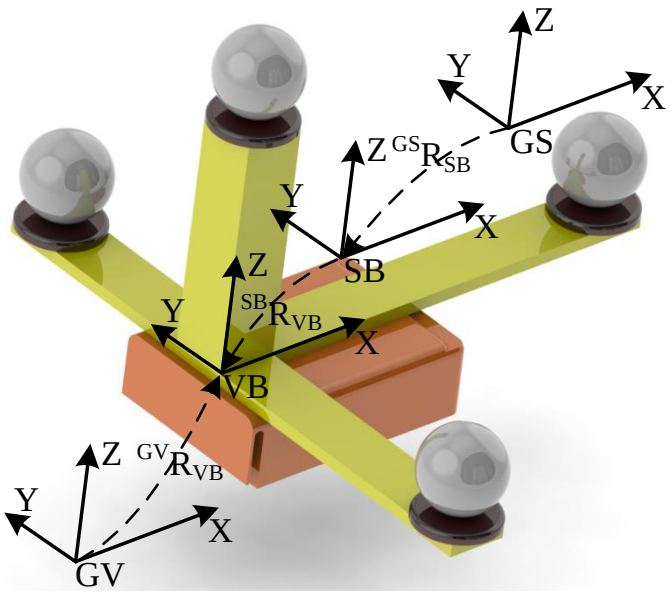


Figure 3-4 Vicon and IMU body frames alignment.

This problem can be formalised as follows: When the orientation measurements of the Vicon ${}^{GV}R_{VB} \in R^3$ and IMU ${}^{GS}R_{SB} \in R^3$ are given, the relative orientation between the Vicon body frame and IMU body frame ${}^{SB}R_{VB}$ needs to be computed. This belongs to the AX = XB problem which has been described in [79][211]. All the measurements are represented in a form of quaternion ($q = w + xi + yj + zk$) where $x, y, z\ and\ w$ are real numbers, and $i, j, and\ k$ are the fundamental quaternion units. The algorithm which has been modified and fitted to this particular problem is illustrated in

Table 3-1

Table 3-1 Algorithm 1: IMU and VICON frame alignment.

Input: ${}^{GS}R_{SB}$, ${}^{GV}R_{VB.}$
Output: ${}^{SB}R_{VB}$
1. Collect 50 [${}^{GS}R_{SB}$, ${}^{GV}R_{VB}$] pairs. Initialise M = []
2. For all i such that $2 \leq i \leq 50$
3. A = ${}^{GS}R_{SB}$ [i-1]$^{-1}$ * ${}^{GS}R_{SB}$ [i]
4. $\emptyset = \text{acos}((Trace(A) - 1)/2)$
5. $LogA = (\emptyset * (A - A^T))/(2 * \sin(\emptyset))$
6. B = ${}^{GV}R_{VB}$ [i-1]$^{-1}$ * ${}^{GV}R_{VB}$ [i]
7. $\emptyset = \text{acos}((Trace(B) - 1)/2)$
8. $LogB = (\emptyset * (B - B^T))/(2 * \sin(\emptyset))$
9. If abs (LogA.norm ( ) – LogB.norm ( ) < 0.01) where abs means absolute value, norm () means matrix norm.
10. Convert $LogA \in SO(3)$ to $w_A \in so(3)$
11. Convert $LogB \in SO(3)$ to $w_B \in so(3)$
12. M = M + $w_B * w_A^T$
13. Else M = M + []
14. End if
15. End for
16. ${}^{SB}R_{VB} = (M^T * M)^{-1/2} * M^T$

After ${}^{SB}R_{VB}$ has been calculated, the orientation measurements from the IMUs and the Vicon can be evaluated. Here, since the human forearm motion is of interest, multi-axis movements are required.

3.2.4 Orientation compensation using Vicon and IMUs

As discussed in the introduction, a Vicon system might lose tracking of the object due to occlusions and the frames might flip suddenly depending on their relative position to the Vicon cameras. In addition, the ambient working environment might affect the local frames of the IMUs because the magnetic field measurements from the magnetometer can be biased. Hence, a frequent resetting of the origin is needed. Therefore, the setup shown in Figure 3-2 Experiment setup for Vicon-CMM evaluation. Left: the actual setup. Right: a schematic plot and annotation of the setup.is proposed to overcome these problems. A similar setup can be found in [55]. The difference is that they put the markers directly on the skin rather than setting them up on a rigid structure to ensure the relative position of the markers is not affected by the muscle movement and hence derive measurements that are more reliable. As shown in the setup, multiple IMUs were used to track each rigid link of the human arm which allowed the pose of the manipulated object to be deduced. This requires additional anatomical calibration for sensor-on-body alignments. However, this approach requires extra human postures to be collected and affected by the skin moves [79]. The

armbands that are worn on the upper and lower arm contain a built-in IMU in the pod where the Vicon marker frame is attached. The GV refers to the global Vicon frame for all the Vicon body frames with the annotations H, L and U for hand, lower arm and upper arm respectively. The GS_U refers to the inertial frame of the upper arm IM. The GS_L refers to the inertial frame of the lower arm IMU. The GS_H refers to the inertial frame of the hand IMU.

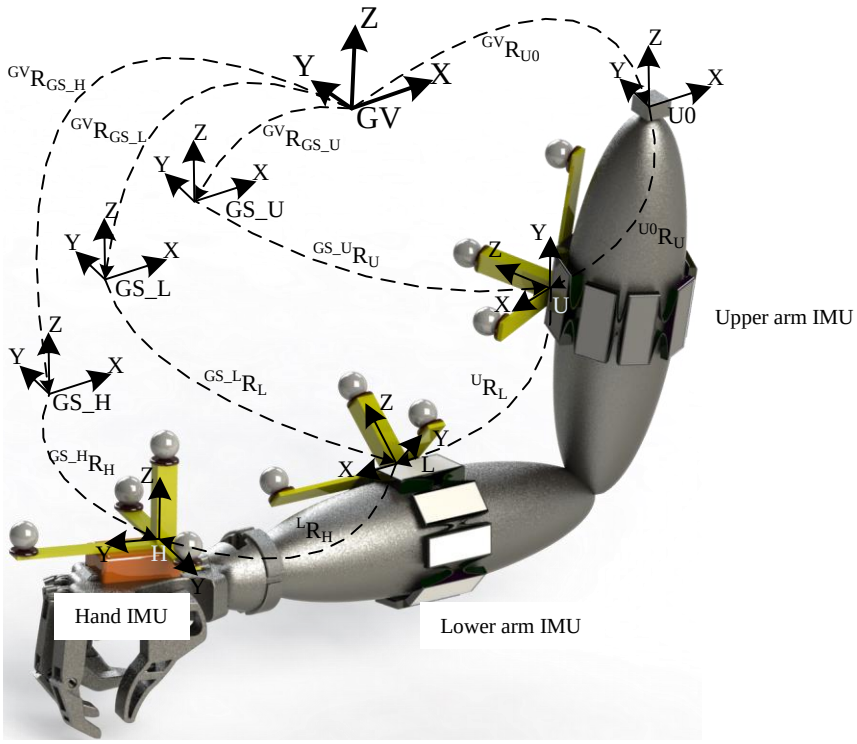


Figure 3-5 Setup for Vicon-IMUs compensation.

The algorithm to fuse the Vicon and IMU data is illustrated in Table 3-2. The inputs are: ${}^{SB}R_{VB}$, ${}^{GV}R_{VB}$, ${}^{GS}R_{SB}$ from the body frame alignments and measurements from both the Vicon system and IMUs. One of the outputs is ${}^{GV}R_{GS} \in R^3$, the Inertial frames of the IMUs. It is periodically calculated as shown in equation (3-1).

$$ {}^{GV}R_{GS} = {}^{GV}R_{VB} * ({}^{SB}R_{VB})^T * ({}^{GS}R_{SB})^T \quad \text{.....(3-1)} $$

where $SB \in [Upper\ arm\ (U), Lower\ arm\ (L), Hand\ (H)]$. ${}^{GS(0)}R_{GS(t)}$ is calculated relative to the initial frame and converted to Euler angles according to equation (3-2). These angles enable the resetting of the IMU inertial frame by comparing them with a threshold value (1˚ in this chapter).

$$\begin{bmatrix} roll \\ pitch \\ yaw \end{bmatrix} = \begin{bmatrix} arctan\frac{2(wx+yz)}{1-2(x^2+y^2)} \\ \arcsin(2(wy-zx)) \\ arctan\frac{2(wz+xy)}{1-2(y+z)} \end{bmatrix} \quad \text{(3-2)}$$

The measurements from the IMUs can be transformed into the Vicon system world coordinates by calculating ${}^{GV}R_{GS} * {}^{GSt}R_{SB} * {}^{SB}R_{VB}$. On the other hand, the measurement from the Vicon system needs to be checked by calculating ${}^{VB(t-1)}R_{VB(t)}$ which inspects the inconsistency of the pose measurements. The subscription *t* in the bracket is the measurement from the current timestamp, whereas *(t-1)* is from the previous timestamp. If the frame flips or the data is not available, the measurement ${}^{GV}R_{VB}$ will be replaced by the value from the IMUs $\overline{({}^{GV}R_{VB(t)})}$. At this point, by assuming that the IMUs are representing each body segment, the relative transformations can be calculated based on forward kinematics [79].

By using the proposed method, ${}^{GV}R_{GS}$ will not update until the next period of checking has occurred and both the IMU and the Vicon measurements are available for the same time stamp. This will correct the bias of the IMUs and keep tracking the body segments while the object is occluded or flipped from the Vicon cameras.

Table 3-2. Algorithm 2: Vicon-IMU compensation method.

Input: ${}^{SB}R_{VB}$, ${}^{GV}R_{VB}$, ${}^{GS}R_{SB}$
Output: ${}^{GV}R_{GS,}$ $\overline{({}^{GV}R_{VB})}$
For $t = 0:T$
  if t == 0
    ${}^{GV}R_{GS(0)} = {}^{GV}R_{VB(0)} * ({}^{SB}R_{VB(0)})^{-1} * ({}^{GS(0)}R_{SB(0)})^{-1}$
  else
    ${}^{GV}R_{GS(t)} = {}^{GV}R_{VB(t)} * ({}^{SB}R_{VB})^{-1} * ({}^{GS(t)}R_{SB(t)})^{-1}$
    ${}^{GS0}R_{GS(t)} = ({}^{GV}R_{GS(0)})^{-1} * {}^{GV}R_{GS(t)}$
    if any roll, pitch or yaw > 1˚
      $\overline{({}^{GV}R_{VB(t)})} = {}^{GV}R_{GS(t)} * {}^{GSt}R_{SB(t)} * {}^{SB}R_{VB}$
    else
      $\overline{({}^{GV}R_{VB(t)})} = {}^{GV}R_{VB(t)}$
    end if
    ${}^{VB(t-1)}R_{VB(t)} = ({}^{GV}R_{VB(t-1)})^{-1} * ({}^{GV}R_{VB(t)})$
    (roll, pitch, yaw) = quat2Euler $({}^{VB(t-1)}R_{VB(t)})$
    if ${}^{GV}R_{VB(t)}$ unavailable or any roll, pitch or yaw > 10˚
      ${}^{GV}R_{VB(t)} = \overline{({}^{GV}R_{VB(t)})}$
    end if
  end if
end For

3.3 Result analysis

3.3.1 Evaluation of the Vicon with CMM

The 2-camera Vicon system claimed to have 0.5mm accuracy. This 0.5mm is a general performance which may improve if the working volume is constrained. To evaluate the accuracy of the Vicon system, targets were attached to a CMM which has 2.1μm accuracy in all axes. A number of dynamic poses were evaluated by moving the single Z axis, both the XZ axes and the XYZ together. The CMM moving distance is equal to x, y and z axis with an incremental value. The total linear distance is equal to $\sqrt{x^2 + y^2 + z^2}$. By recording the initial position, the error is defined as: *e = Total linear distance in CMM – object travel distance in Vicon*. The results are shown in Table 3-3-3-5:

Table 3-3 Evaluation of the Vicon with CMM in the X axis.

Trials	Axis	CMM distance (mm)	Total linear distance (mm)	Mean Error (mm)	SD (mm)
5	X	1000	1000	0.132	0.022

Table 3-4 Evaluation of the Vicon with CMM in X and Y axis.

Trials	Axis	CMM distance (mm)	Total linear distance (mm)	Mean Error (mm)	SD (mm)
5	X and Y	1	1.414	-0.012	0.006
		10	14.142	0.054	0.006
		100	141.421	0.431	0.01
		500	707.107	1.542	0.03
		1000	1414.214	2.346	0.059

Table 3-5 Evaluation of the Vicon with CMM in X, Y and Z axis.

Trials	Axis	CMM distance (mm)	Total linear distance (mm)	Mean Error (mm)	SD (mm)
5	X,Y, and Z	1	1.732	0.004	0.005
		10	17.321	0.083	0.013
		100	17.321	0.194	0.015
		500	866.025	0.703	0.036
		1000	1536.229	1.121	0.058

The main results are:

- In the one axis case, the total moving distance is 1000 mm between the two extremes of the CMM measurement volume; the mean error is 0.132mm with a standard deviation 0.022 mm. The overall performance is better than the claimed accuracy.

- In the two-axes case, the accuracy degrades when the object moves further away from the camera. When the moving distance is 10mm, the mean error is 0.054mm with a standard deviation 0.006 mm. But when the moving distance is 100 mm, the mean error is 0.431mm with a standard deviation 0.01mm. This result is almost equal to the claimed accuracy. At a maximum distance of 1000mm, the mean error is only 2.346mm with a standard deviation 0.059 mm, actually much worse than stated for the Vicon system.
- In the three-axes moving case, the accuracy also degrades when the object moves further away from the camera. The difference is when the moving distance is 100mm, the Vicon system is still within the claimed accuracy. At the maximum distance, the mean error is 1.121mm with a standard deviation 0.058mm. Again, this is worse than the stated accuracy of the system.

From the above results, the mean error of the Vicon system is not as good as the claimed value in multiple axis movements when the object moves away from the cameras. Similar results were also indicated in [47], where the further the object is away from the camera the less accurate the system became. In [212], they also discussed the improvement of the positioning accuracy by increasing the number of cameras used in any photogrammetry system. In general, by having more intersections from multiple cameras, the Vicon system can provide more reliable and accurate results. The 2-camera setup in this chapter is fixed and symmetrically placed, the performance of one diagonal movement is assumed to equal to another diagonal movement. The evaluation of the Vicon system was done with discrete positions, and the results are similar to the previous works. The results indicated that starting from the near camera side of the evaluation volume, more accuracies can be achieved when the object moving distance is less than 100mm. Therefore, by referring to the result in Table 3-3-3-5, the measurements from the Vicon system used in this work can be more accurate (0.069 - 0.488mm) than the claimed accuracy if the working volumes are limited and close to the cameras.

However, the accuracy is not the only characteristic that makes the 2-camera Vicon system suitable for human motion tracking. Once the markers have been attached to the frame and fixed to the IMU body, tracking would be lost at some poses because the human body segments may occlude the markers. These missing information will

cause a problem for human posture tracking as it leaves a blank in the time series data. Also, the frame might flip when only partial view of the markers is available from the cameras hence confusing the correct tracking of the object orientation. An example in Figure 3-6 shows the motion data from the Vicon system that contains frames flipping and occlusion that cause abnormal peaks and missing data.

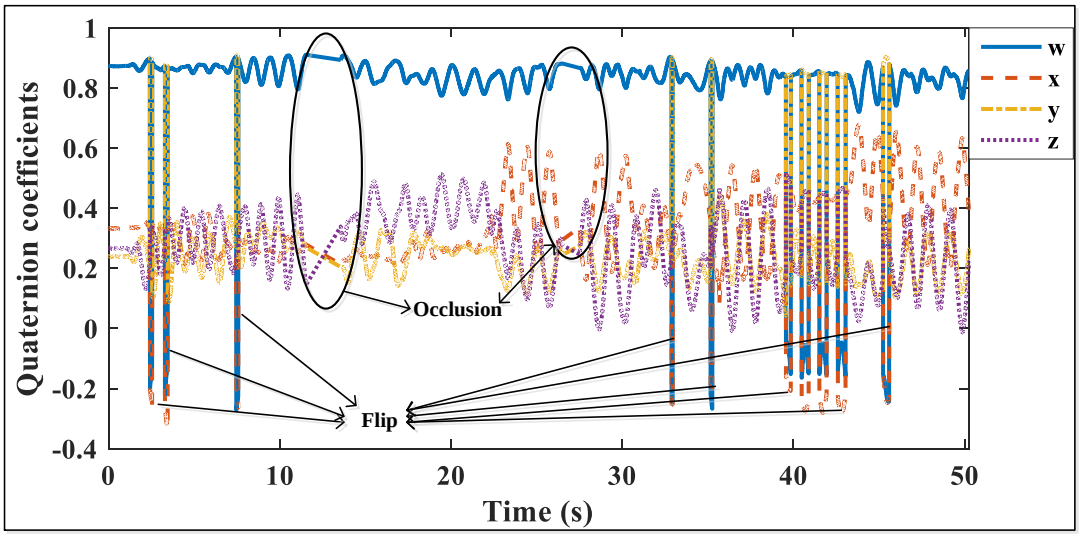


Figure 3-6 An example of Vicon flipping and occlusion problem during motion tracking. The data is represented in quaternion.

3.3.2 Evaluation of the IMUs with Vicon

a) Body alignment results

The body frame transformations ${}^{VB}R_{SB}$ from the IMU inertial frames to the Vicon origin frame were estimated based on algorithm 1 (Table 3-1). The collected orientations contain enough valid data samples (at least 3) that allow them to be discriminated from each other. As described in section 3.2.1, an incremental number of valid pairs of the Vicon-IMU measurements were used for ${}^{VB}R_{SB}$ estimation. The evaluation results are shown in Figure 3-7, Figure 3-8 and Figure 3-9 where the mean and standard deviation values are plotted for yaw, pitch and roll angle rotations from Z, Y, and X axis. The results show that as the number of the frames increase, the estimations tend to be more stable for all the IMUs: the mean and standard deviation have less variation and the error bar (standard deviation of the estimations) tend to be narrower. Therefore, a summary of all the IMUs estimation results from 55 frames were selected and shown in Table 3-6. From the results, even though the Vicon frame was manually aligned with the IMU body (not the inertial frame), none of the Vicon body frames were pre-aligned with the IMU body frames. The estimation indicated that the body frame of the lower arm IMU has a misalignment of [-3.44˚ ± 0.25˚, -3.66˚ ± 0.32˚,

1.28˚ ± 0.39˚] from the Vicon frame in roll pitch and yaw; the upper arm IMU has a misalignment of [-76.10˚ ± 0.09˚, 0.41˚ ± 0.15˚, -0.22˚ ± 0.23˚]; the hand IMU has a misalignment of [98.51˚ ± 0.89˚, -1.65˚ ± 0.73˚, 1.18˚ ± 0.17˚]. The results will be used for IMU evaluations in section 3.3.20 and Vicon-IMU compensations in section 3.3.3.

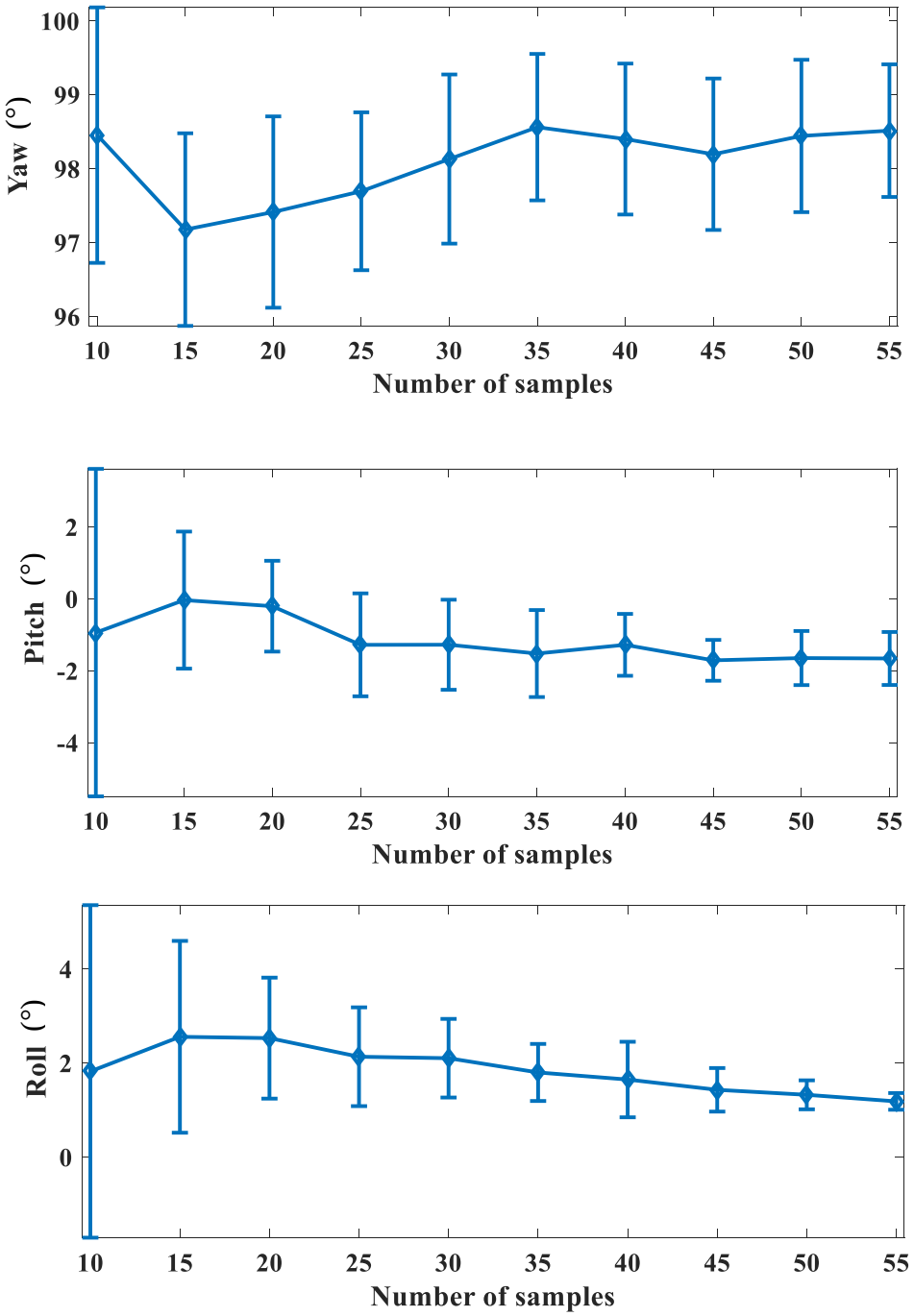


Figure 3-7 Evaluation of the hand IMU- Vicon body frame alignment transformation.

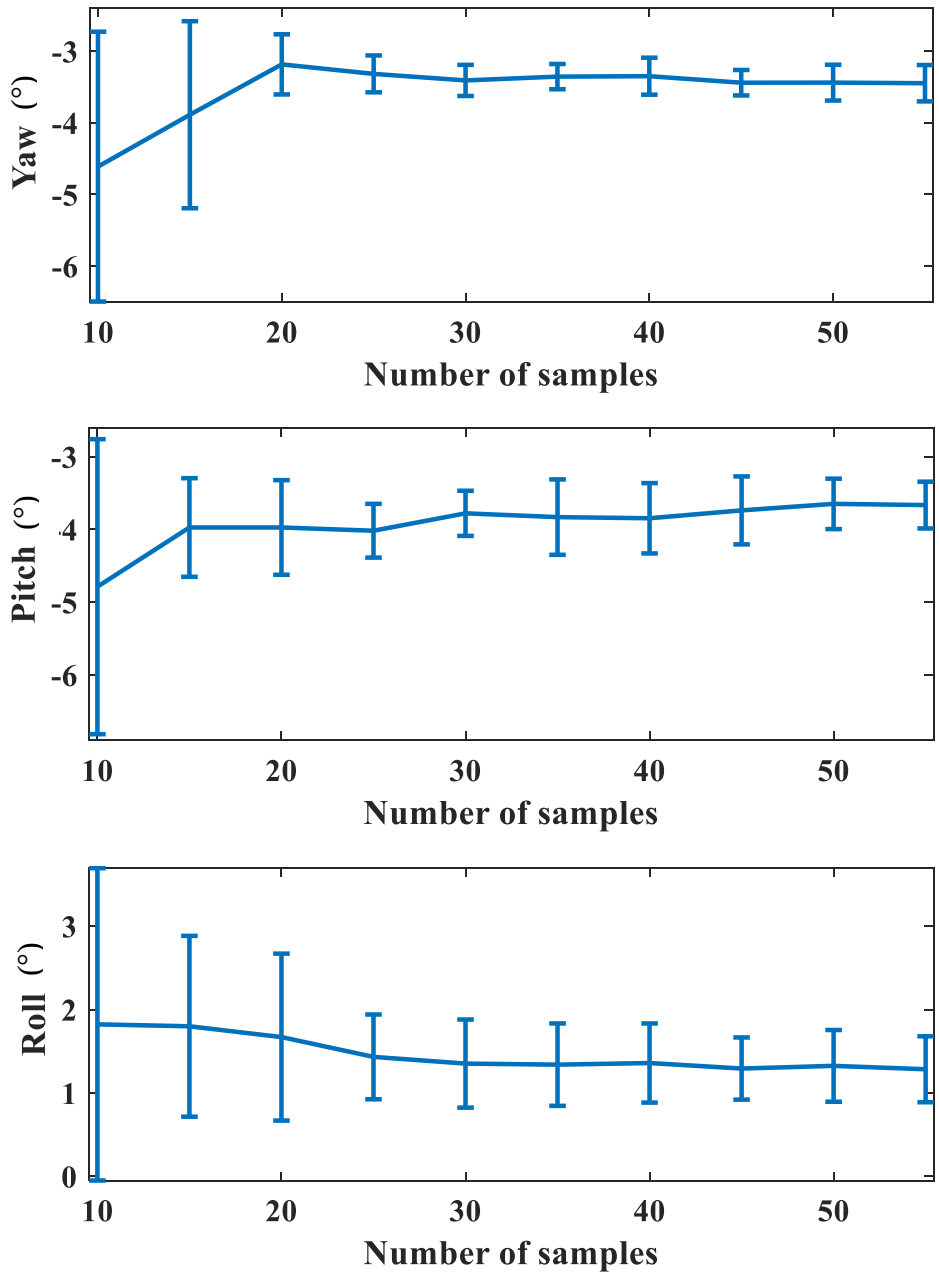
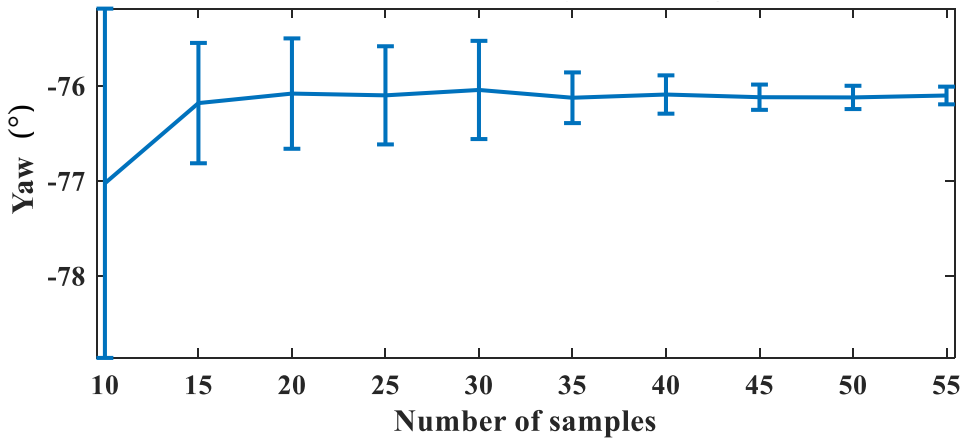


Figure 3-8 Evaluation of the lower arm IMU- Vicon body frame alignment transformation.



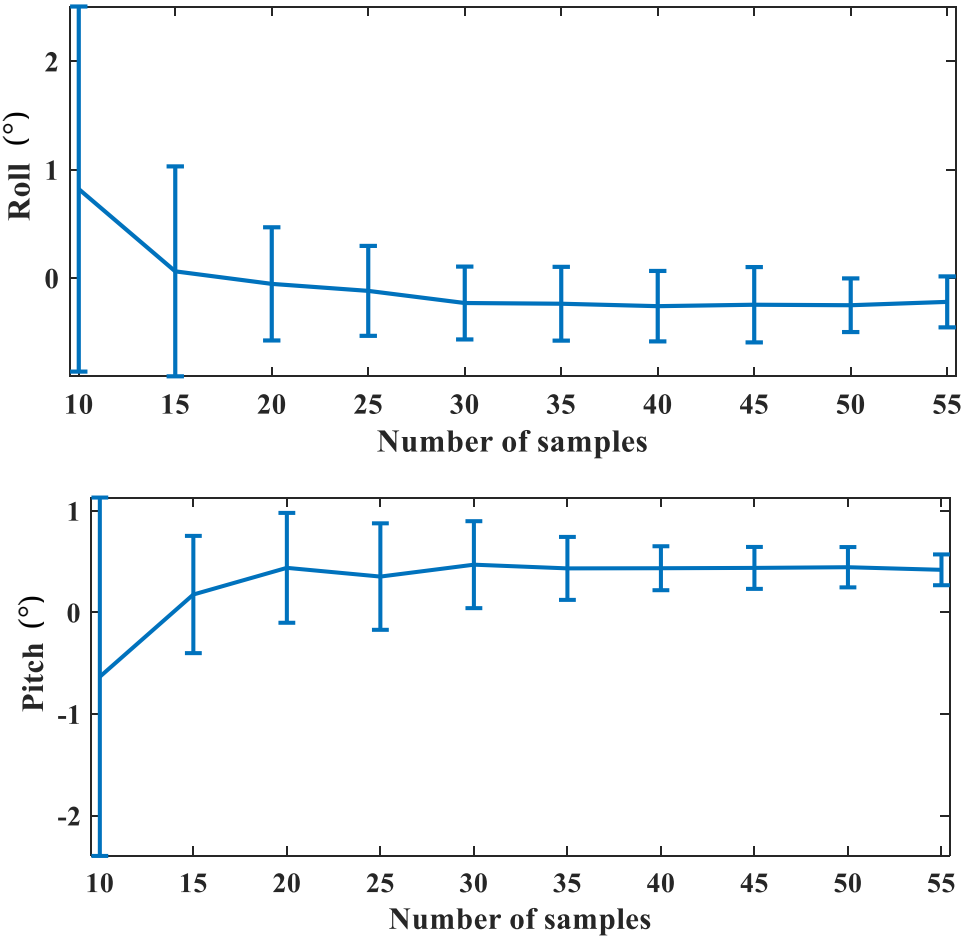


Figure 3-9 Evaluation of the upper arm IMU- Vicon body frame alignment transformation.

Table 3-6 Body frame alignment transformations in Euler angles.

	yaw	pitch	roll
Lower arm IMU	-3.44 ˚ ± 0.25˚	-3.66˚ ± 0.32˚	1.28˚ ± 0.39˚
Upper arm IMU	-76.10˚ ± 0.09˚	0.41˚ ± 0.15˚	-0.22˚ ± 0.23˚
Hand IMU	98.51˚ ± 0.89˚	-1.65˚ ± 0.73˚	1.18˚ ± 0.17˚

b) Evaluation results

After calculation of the body frame transformations ${}^{VB}R_{SB}$ for each IMU in section 3.3.2a), the IMUs can now be evaluated. The frames used in this section have been defined in Figure 3-5. Three trials over a 300s interval each have been collected. In the beginning, the first frames from the first measurements of all the sensors were used as a reference. ${}^{GS0}R_{GS} = ({}^{GV}R_{GS0})^{-1} * {}^{GV}R_{GS}$ is the transformation from the inertial frame from the first measurement to the current inertial frame of the IMUs. As a reminder, ${}^{GV}R_{GS}$ is calculated based on equation (3-1). Since the Vicon world coordinate frame (GV) will not change, therefore, the ${}^{GS0}R_{GS}$ of the IMUs were

calculated to evaluate how much they differ from one to another relative to GV. The Euler angles are interpreted in the yaw, pitch and roll order.

As shown in Figure 3-10, the measurements from IMU and Vicon are not directly comparable without frame alignment. Therefore, it is essential to use the body frame alignment transformations from section 3.3.2a). Figure 3-11shows the evaluations of the hand IMU on trial 1 for visualisation of the IMU inertial frame drifting problem. A smoothed curve and zero reference curves are also shown for comparison purposes. The most significant drifting occurs on the yaw angle. The sensor itself can cancel the drifting in pitch and roll angles oscillate around the zero reference, which indicates that, the drifting. The experiments were carried out according to the specific patterns defined in section 3.2.1. But from the statistics results across all trials in Figure 3-12, the histogram plots shows the distribution of the drifting error. From the plots, the different shapes of distributions indicate that the drifts are not predictable. For instance, the histogram in the yaw drifts in trial 1 is biased towards a positive angle. But histogram is biased to a negative angle in trial 2 and contains two peaks in trial 3. The unpredictable drifts apply on the lower arm and upper arm IMUs, as shown in Figure 3-13-3-16. Especially in Figure 3-13, the lower arm IMU tends to have stable inertial frame for the first 50s, but then the drifts become more significant. For the upper arm IMU, as shown in Figure 3-15, the yaw angle is stable for 20s, but then the drifts become worse. The drifts of the pitch and roll angles seem to periodically get back to zero, but again from the histogram in Figure 3-16, they are not predictable.

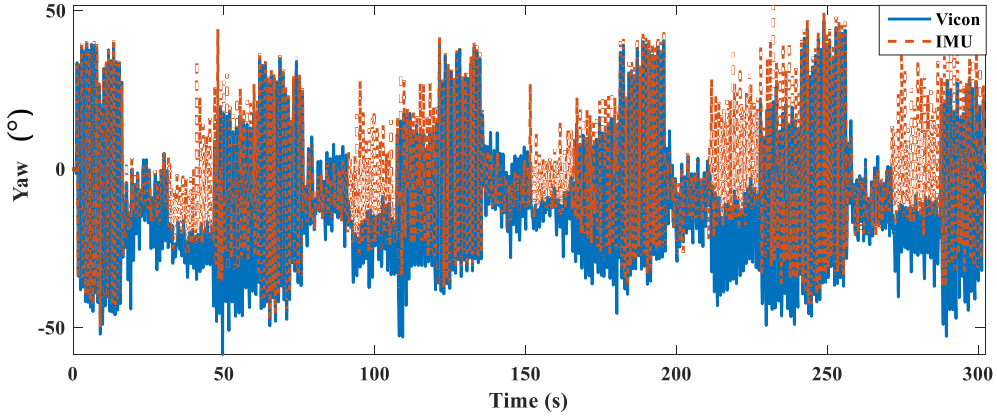


Figure 3-10 An example of IMU evaluation using Vicon without body frame alignment.

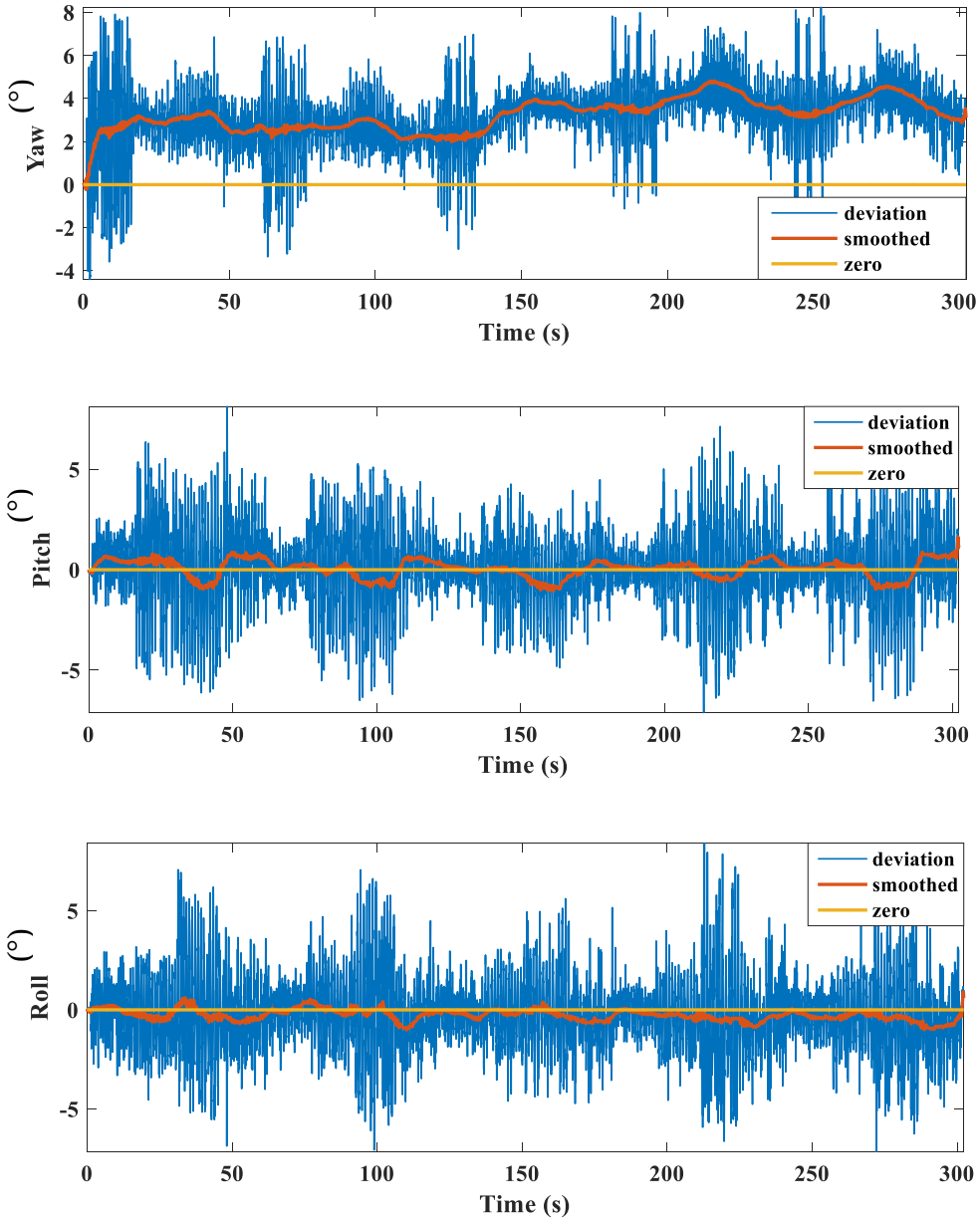


Figure 3-11 The deviation of the hand IMU inertial frame after alignment for trial 1.

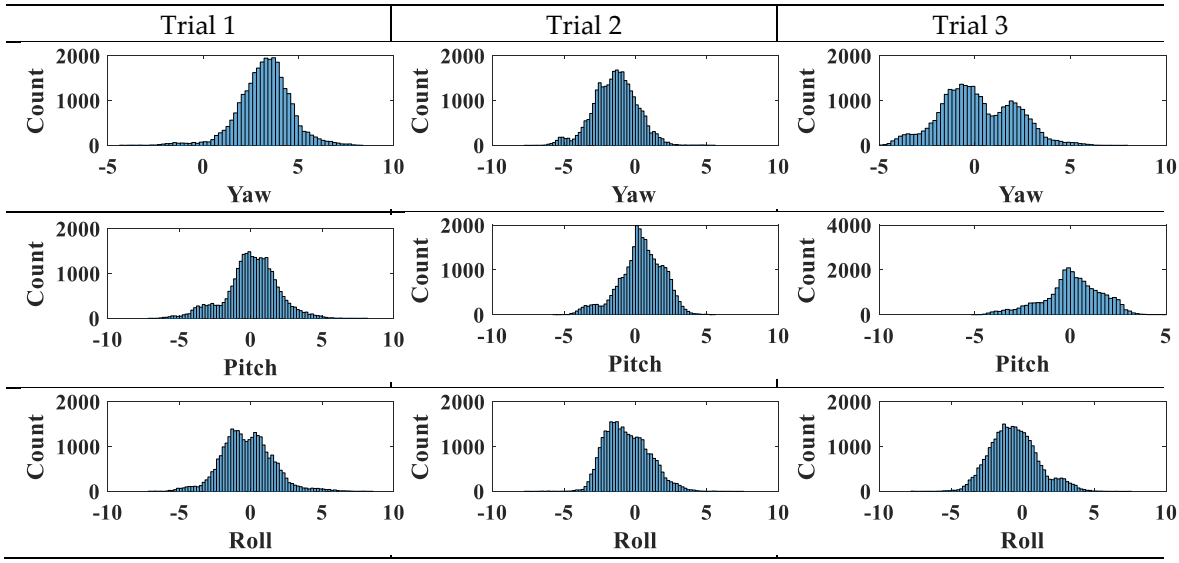
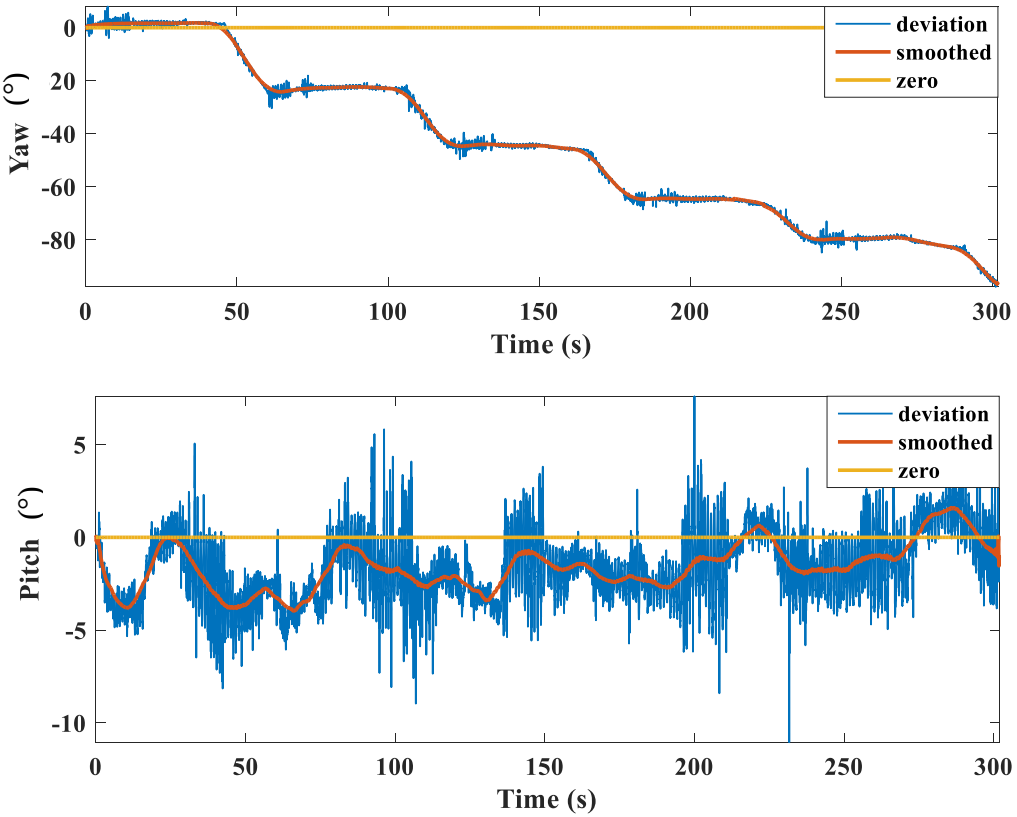


Figure 3-12 Statistics of the inertial frame deviations for the hand IMU.



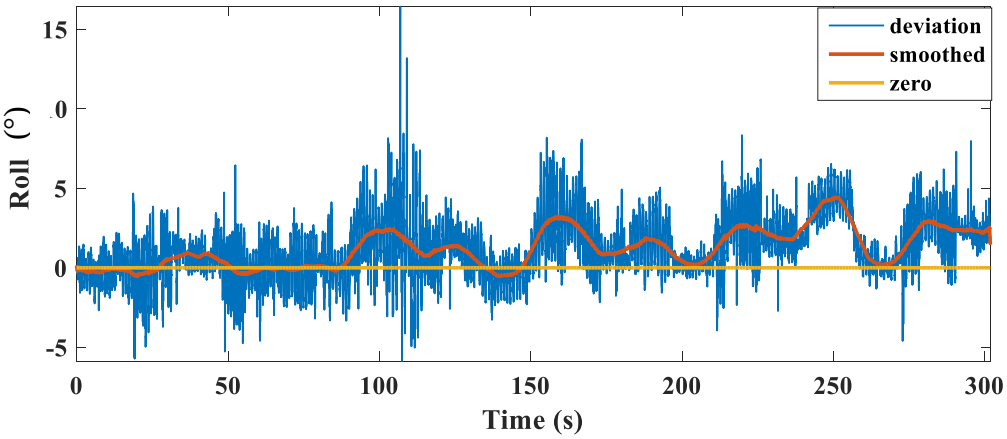


Figure 3-13 The deviation of the lower arm IMU inertial frame after alignment for trial 1.

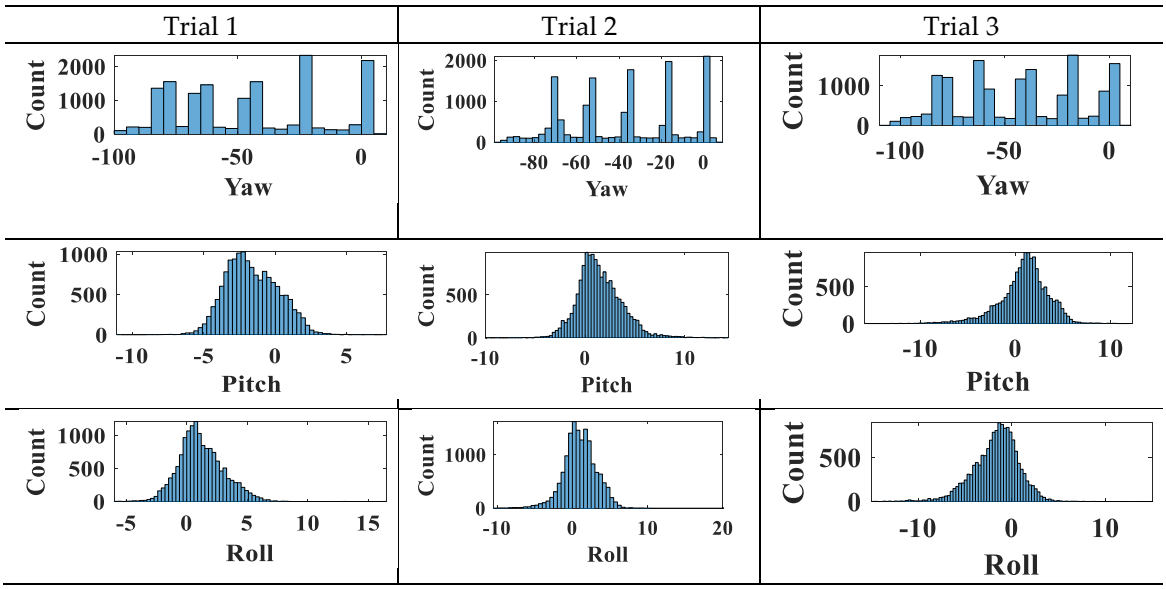
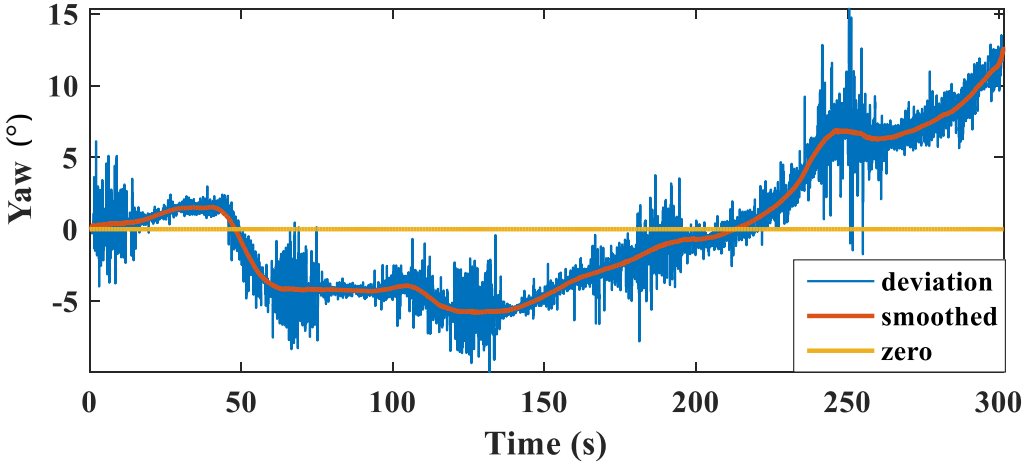


Figure 3-14 Statistics of the inertial frame deviations for the lower arm IMU.



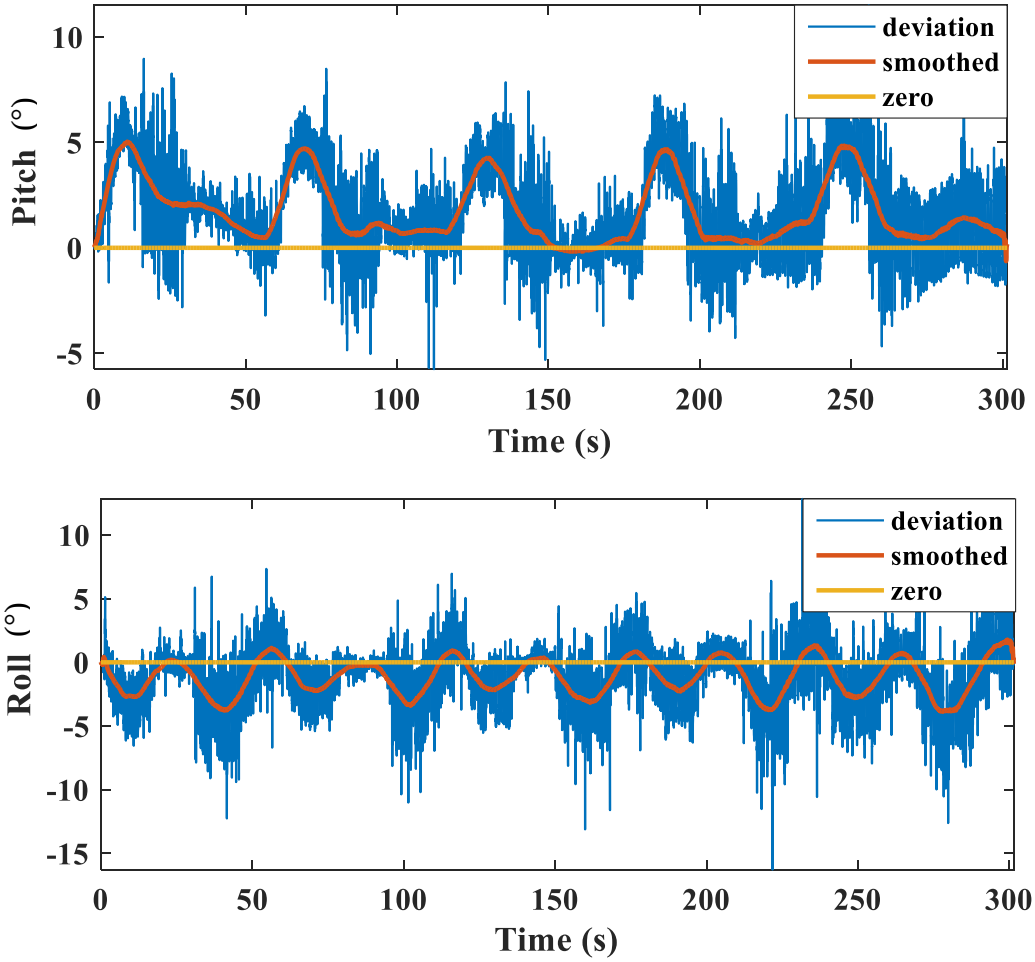


Figure 3-15 The deviation of the upper arm IMU inertial frame after alignment for trial 1.

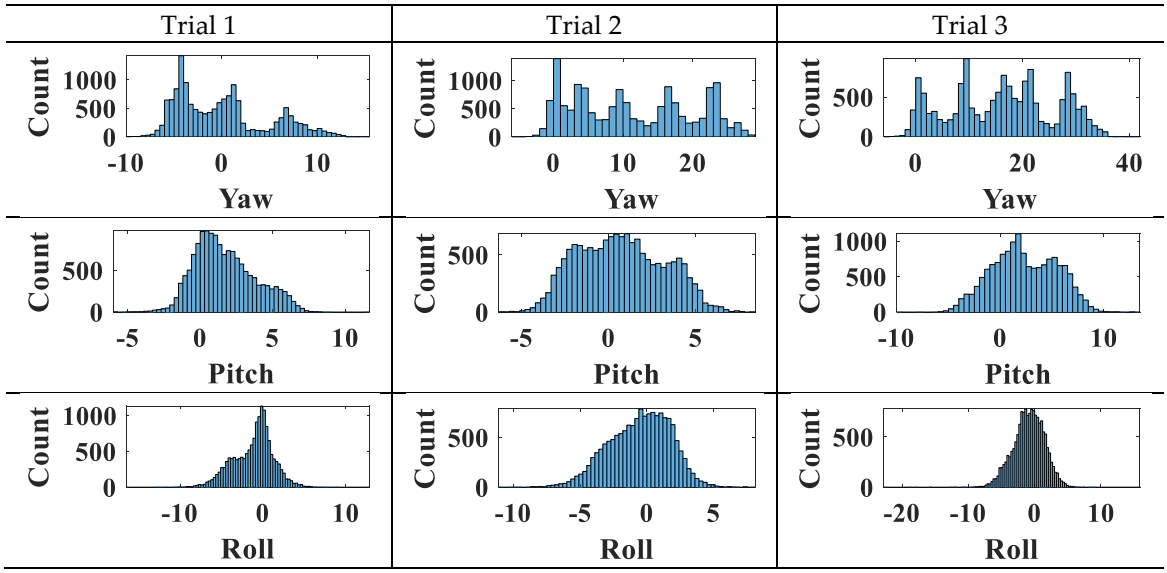


Figure 3-16 Statistics of the inertial frame deviations for the upper arm IMU.

In conclusion, the IMU's inertial frames keep changing in the ambient environment. The drifting problems for yaw angles are larger for all the IMUs used.

The drifts of the pitch and roll angles show some pattern, but they are not predictable and non-systematic. These make it difficult to track human body movements reliably with IMUs alone. Therefore, a reference system is needed to compensate the non-systematic drifting issues of the IMUs.

3.3.3 Implementation of the Vicon-IMU system

To validate the overall approach, four operators have been asked to wear the IMU sensors as shown in Figure 3-5. In this section, two industrial manipulation tasks: a PiH assembly and pick-and-place task were used to discuss the performance of the proposed Vicon-IMUs system.

The results in Table 3-7-Table 3-9 show the performance of the proposed Vicon-IMU compensation method compared to the IMU-only method. The Mean Square Error (MSE), as the performance indicator is calculated in between the reference Vicon signal and the measurements from either IMU-only or with compensation method. The mean and standard deviation of the MSE across 5 trials are used to show the variation of the performance for each subjects. In all cases, the compensated measurements have smaller MSE than the IMU-only method. A golf swing motion was tracked with ±3˚ angle accuracy [79]. Here we are aiming at a better tracking performance (±2˚). The maximum error in the compensation approach is 1.89˚± 0.68˚ from the yaw angle of the lower arm IMU. This error is smaller than the expected value. The error in the IMU-only approach is much larger the expected value, therefore the motion tracking performance is unreliable.

Since the comparison of the results berween the four subjects by using the IMU-only method is not applicable (too much un-systematic drifting), the measurements from the compensation method were used. For PiH task, subject A has a larger error on the hand tracking and upper arm tracking, subject C has a larger error on the lower arm tracking. For Pick and Place (PnP) task, subject C has a larger error on the hand tracking, subject A has a larger error on the lower arm tracking and subject B has a larger error on the upper arm tracking. These indicate that tracking on the different subject contains variations in different body parts. This is partially because the different types of IMUs behave differently. And more importantly, since the drifts in the IMUs are un-systematic, they influence the performance of the compensation method. For instance, at times, the occlusion is too long and hence the compensation is

not enough and not in-time when the Vicon system is recovered from the occlusion or flipping issue. In general, the tracking error after compensation is one magnitude smaller than the error with IMU only. In conclusion, the above results indicate that the proposed compensation method can improve the human motion tracking performance especially flipping and occlusions for some time. The longer the occlusion, the more estimation bias can present due to IMU drifts and environmental changes. For the applications used in this study, the occlusion time (1~2s) is not the predominant factors, therefore, futher discussion of the influence of the occlusion time will be addressed in the future work.

Table 3-7 Evaluation of the proposed compensation method for the hand IMU

a)	PiH Hand IMU					
	IMU only			With compensation		
	Yaw	Pitch	Roll	Yaw	Pitch	Roll
Subject A	2.40˚± 0.75˚	3.09˚± 0.60˚	5.25˚± 1.13˚	0.26˚± 0.07˚	0.35˚± 0.15˚	0.89˚± 0.33˚
Subject B	2.10˚± 1.97˚	5.18˚± 6.55˚	6.97˚± 7.17˚	0.14˚± 0.09˚	0.32˚± 0.16˚	0.50˚± 0.28˚
Subject C	0.92˚± 0.22˚	2.59˚± 1.35˚	2.62˚± 0.46˚	0.16˚± 0.05˚	0.37˚± 0.11˚	0.32˚± 0.20˚
Subject D	1.65˚± 0.55˚	5.82˚± 3.89˚	4.46˚± 1.28˚	0.19˚± 0.08˚	0.41˚± 0.35˚	0.58˚± 0.19˚

b)	PnP Hand IMU					
	IMU only			With compensation		
	Yaw	Pitch	Roll	Yaw	Pitch	Roll
Subject A	6.87˚±13.83˚	3.07˚± 2.04˚	13.77˚±23.80˚	0.07˚± 0.02˚	0.16˚± 0.06˚	0.23˚± 0.13˚
Subject B	0.63˚± 0.09˚	2.16˚± 1.49˚	3.52˚± 2.39˚	0.10˚± 0.07˚	0.17˚± 0.13˚	0.21˚± 0.16˚
Subject C	1.05˚± 0.58˚	2.88˚± 0.84˚	5.01˚± 2.18˚	0.28˚± 0.20˚	0.49˚± 0.73˚	0.61˚± 0.59˚
Subject D	2.19˚± 3.19˚	4.32˚± 6.61˚	9.14˚± 11.45˚	0.12˚± 0.07˚	0.14˚± 0.14˚	0.22˚± 0.07˚

Table 3-8 Evaluation of the proposed compensation method for the lower arm IMU

a)	PiH Lower arm IMU					
	IMU only			With compensation		
	Yaw	Pitch	Roll	Yaw	Pitch	Roll
Subject A	4.47˚± 1.32˚	5.06˚± 1.88˚	2.84˚± 0.60˚	1.30˚± 0.47˚	0.62˚± 0.21˚	0.39˚± 0.18˚
Subject B	3.80˚± 1.13˚	4.64˚± 1.60˚	2.44˚± 0.93˚	1.16˚± 0.48˚	0.44˚± 0.18˚	0.46˚± 0.19˚
Subject C	7.32˚± 3.30˚	8.49˚± 2.29˚	3.81˚± 0.94˚	1.89˚± 0.68˚	0.57˚± 0.21˚	1.69˚± 1.05˚
Subject D	5.85˚± 1.50˚	4.53˚± 2.02˚	4.05˚± 1.05˚	1.56˚± 0.51˚	0.89˚± 0.32˚	1.00˚± 0.20˚

b)	PnP Lower arm IMU					
	IMU only			With compensation		
	Yaw	Pitch	Roll	Yaw	Pitch	Roll
Subject A	3.46˚±1.07˚	1.67˚± 0.47˚	1.78˚±0.52˚	0.69˚± 0.22˚	0.47˚± 0.30˚	0.39˚± 0.19˚
Subject B	2.80˚± 1.12˚	1.07˚± 0.41˚	1.10˚± 0.35˚	0.37˚± 0.09˚	0.23˚± 0.08˚	0.34˚± 0.10˚
Subject C	2.71˚± 0.92˚	1.26˚± 0.55˚	1.24˚± 0.31˚	0.45˚± 0.13˚	0.39˚± 0.09˚	0.55˚± 0.30˚
Subject D	1.79˚± 0.47˚	1.94˚± 0.62˚	2.42˚± 1.23˚	0.45˚± 0.24˚	0.48˚± 0.23˚	0.47˚± 0.08˚

Table 3-9 Evaluation of the proposed compensation method for the upper arm IMU

a)	PiH Upper arm IMU					
	IMU only			With compensation		
	Yaw	Pitch	Roll	Yaw	Pitch	Roll
Subject A	2.07˚± 1.19˚	3.77˚± 1.17˚	1.37˚± 0.30˚	0.40˚± 0.33˚	0.03˚± 0.01˚	0.12˚± 0.10˚
Subject B	2.63˚± 0.82˚	6.29˚± 0.60˚	1.75˚± 0.24˚	0.13˚± 0.05˚	0.06˚± 0.02˚	0.05˚± 0.03˚
Subject C	3.81˚± 0.96˚	8.09˚± 0.95˚	1.47˚± 0.42˚	0.06˚± 0.03˚	0.01˚± 0.01˚	0.06˚± 0.03˚
Subject D	6.93˚± 1.50˚	9.52˚± 3.26˚	3.21˚± 1.08˚	0.08˚± 0.06˚	0.04˚± 0.03˚	0.12˚± 0.10˚

b)	PnP Upper arm IMU					
	IMU only			With compensation		
	Yaw	Pitch	Roll	Yaw	Pitch	Roll
Subject A	9.37˚±2.90˚	5.92˚± 1.49˚	5.29˚±7.96˚	0.14˚± 0.07˚	0.05˚± 0.02˚	0.11˚± 0.03˚
Subject B	9.25˚± 1.53˚	5.27˚± 1.23˚	5.76˚± 9.06˚	0.38˚± 0.20˚	0.15˚± 0.06˚	0.19˚± 0.09˚
Subject C	8.74˚± 2.62˚	6.46˚± 1.36˚	5.25˚± 7.58˚	0.25˚± 0.08˚	0.12˚± 0.08˚	0.16˚± 0.12˚
Subject D	11.21˚±3.66˚	3.43˚± 0.31˚	5.06˚±11.64˚	0.33˚± 0.07˚	0.13˚± 0.02˚	0.23˚± 0.08˚

3.4 Conclusion

In this chapter, the objective is to evaluate the state of the art motion tracking system and improve their reliability to be used within an LfD framework to better transfer skills from human experts to machines. To achieve this, a combined 2-camera Vicon system with IMU inertia tracking has been successfully evaluated. A new data fusion method has been created that compensates for the known limitations of the both systems and improves their combined accuracy (±2˚ achieved) and robustness for tracking human tasks with minimum drifts. It has been demonstrated that the proposed compensation method generates continuous and more reliable joint transformation measurements. This system was implemented for tracking both a PiH as well as a Pick-and-Place task. This work is important since it provides a more reliable tracking system for human body segments. Also, the proposed approach only requires a small number of cameras. In this case a 2-camera setup combined with commercially available IMUs was used to successfully track the two tasks. The cost of such a system is very much lower than the normally recommended higher number of camera setups. More importantly, the proposed approach can cope with occlusions which entirely camera based system cannot.

This chapter has made an important contribution in providing a more reliable means to dynamically track human motions within the workspace. Together with the haptic feedback, the motion data will provide the important information required to enable human skills capture in fine manipulation tasks. The system will be used to capture motion data in the next chapter, where a method will be developed to encode the recorded measurements as probabilistic models that can be translated for robot executions.

Chapter 4. Encoding and Reproductions of the Human Skills from Demonstrations - A Peg in Hole Case Study

4.1 Introduction

Most of the time, tasks contain both gross and fine motions. Gross motions can be captured using the proposed motion tracking techniques in chapter 3, whereas fine motions are often required in close or actual contact manipulation situations. Contact between components introduces uncertainties, and robots require Force/Torque (F/T) feedback to detect changes in contact conditions. Geometrical uncertainties are one of the primary sources of the uncertainties. It is difficult for a robot to become adaptive to these because the contact bodies might deform which would invalidate the existing model [213]. Also, it is almost impossible to create a sufficiently accurate geometry model for every single contact configurations; therefore, a readily generalisable method is needed to cope with a wider range of situations.

Compared to robots, humans can adapt to these uncertainties easily. Close contact manipulation tasks can be classified as skill-based tasks, which are harder to extract and generalise so that a robot can understand and use it to inform its control strategy. An important reason why humans are good at performing manipulation tasks is that we have strong haptic feedback. This is a key indicator that capturing the force interactions during manual contact manipulation will be instrumental for transferring some of the tacit knowledge from human experts to a robot.

In this chapter, the aim is to address the research objective two, which is to build state-action policy models from human demonstrations that relate to industrial manual manipulations. To achieve this, the force information is measured by fixing the F/T sensor on the work piece. The human operator wore the IMU-Vicon system introduced in chapter 3 and the hand motions were recorded accordingly. The state-action policy was derived based on the demonstrated subject's skills. A PiH task which requires tacit skills to align the peg with the hole and gradually sliding the peg into the hole task is used as a case study. This task is chosen because the subject can be easily trained to be a skilful operator and the motion and force signals are relatively simple which contain repetitive patterns. Since individual subject might show different skills in completing the task, the generalisation capabilities of the model were evaluated against

different demonstrations and various subjects. The methodology is described in section 4.2. Results analysis is presented in section 4.3 and conclusions are presented in section 4.4.

4.2 Methodology

In order to encode the skills during fine motions, a methodology is proposed in this section. The operators were asked to demonstrate the PiH task according to the experiment setup in 4.2.1. The sample skills were encoded by a novel probabilistic encoding method in 4.2.2. The method was mainly built based on the Gaussian Mixture Model (GMM) and Hidden Markov Model (HMM) which are detailed in section 4.2.2d) and 4.2.2f) respectively. The reproduction of the skills was addressed by Gaussian Regression Model (GMR) in 4.2.3. The related uncertainties in this chapter are listed as follow: i) different data filtering strategy. ii) selection of reference signal for demonstration misalignment. iii) data dimension reduction (c). iv) number of Gaussians in the Gaussian Mixture Model. v) the number of demonstrations. vi) generalisation across different subjects. vii) motion reproductions. The influence of these uncertainties are discussed in section 4.3.

4.2.1 Experiment setup

The subjects involved in the experiment are postgraduate students in Intelligent Automation research lab with mechanical engineering training and are familiar with practical assembly tasks. The experiment setup is shown in Figure 4-1 where each subject performs the experiment in turn. The subject was asked to stand in front of the test rig, wearing the Vicon-IMU system as described in chapter 3. The PiH task contains relatively simple motions, and the hand pose introduces the dominant movement. Therefore, the operator was briefed to restrict the elbow and shoulder joint movements while he/she was doing the task and only use the wrist joint instead. Also, the subject was asked to hold the peg in the way shown in Figure 4-1 to reduce the variations introduced by different holding strategies. The 6 axis F/T sensor used in the experiment is ATi Gamma series. It is rigidly installed underneath the workpiece. The data acquisition rate is 200 Hz and synchronised by using ROS approximation timing synchronizer. The peg and hole diameters used in this case study were 16.0 mm and 16.2 mm respectively. Six subjects (at the average age of 27) were asked to demonstrate the PiH skills and 10 trials were performed by each of them during the

experiment. They were asked to repeat the experiment a few times to familiarise themselves with the approach for consistency. Each trial contained approaching, insertion, extraction and resting phase lasting between 3-5 seconds. At the beginning, the initial approaching strategy for individual subject is not constrained by the same pose. Then, an alignment procedure is applied (as discussed in section 4.3.2) and the subjects are asked to repeat the experiments again.

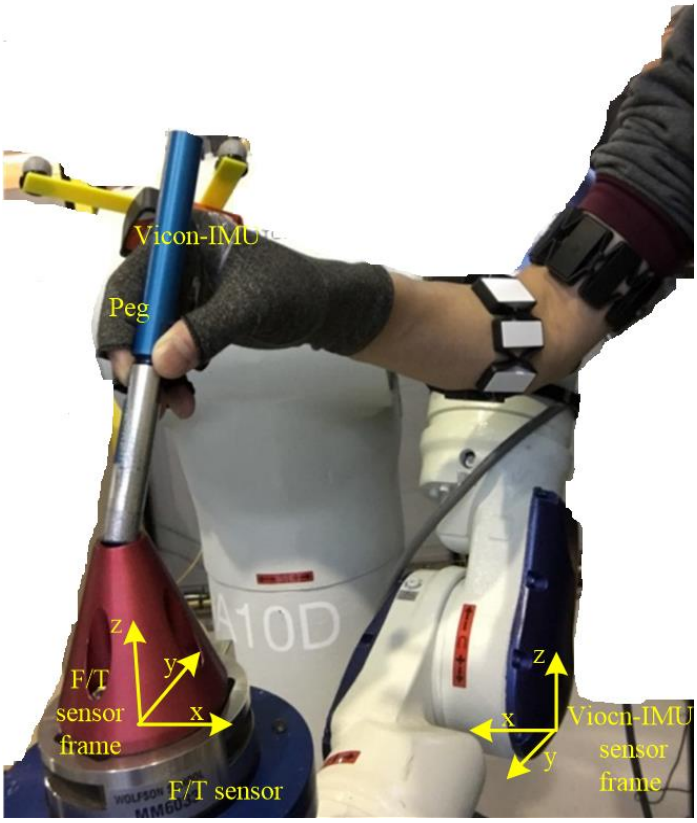


Figure 4-1 Experiment setup of the PiH process.

4.2.2 Novel probabilistic skills encoding method

The proposed methodology for skill encoding is shown in Figure 4-2. The input signals used in this chapter are the 1st PCA of the force signal and the hand orientations $\{f_{pca}^{1}, roll, pitch, yaw\}$. As shown in Figure 4-2, the methodology contains two levels. In the first level, models are built on individual subjects across various demonstrations. The GMR method is used to generate one single smooth curve which represents his/her model. In the second level, a generalisation model is further built based on the regression models from different subjects. This model represents skills across all the subjects.

The skill encoding method is a key component of the LfD framework, and details are explained in this section. The input signals were firstly pre-processed, which includes filtering, Dynamic Time Warping (DTW) alignment, and PCA as described in subsection a), b) and c). Then, a skill extraction pipeline was implemented for encoding and decoding the pre-processed signals. A number of hidden states (each one accounts for a specific distribution of the samples) were learned and recognised. In this pipeline, a *K-means* algorithm was used to initialise the GMM. The GMM was then trained to estimate observation density for each state. A Bayesian Information Criterion (BIC) was utilised for the model selection where the candidate number of components was determined. The HMM then took the output from the GMM and further encoded the states into transition probabilities. Finally, the Viterbi algorithm [214] was used for the state recognition (decoding) with a given set of model parameters determined from the training model. After recognition, the operator interpreted the state during the process, and some of the parameters such as the transition matrix (a matrix that describes the probabilities of a state to transition to another state) could be refined after interpretation by the experts.

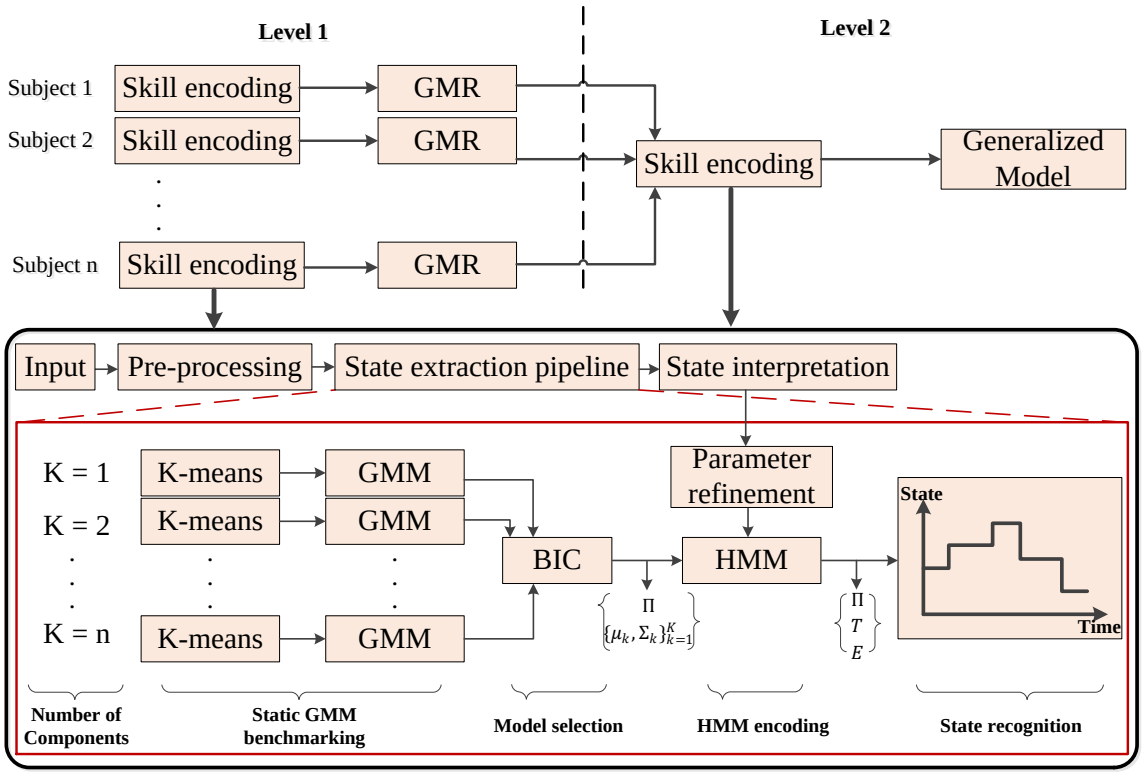


Figure 4-2 An overview of the methodology for skill encoding.

The HMM has been chosen as a general encoding structure to limit the assumptions made on the spatial-temporal nature of the data set. The time variables are implicitly represented by the state transition model using HMM and the Markov assumption assumes the current state contains all the information from the previous states. This reduces the model complexities. Here, a continuous HMM, with a full covariance matrix describing the distributions of the output variables was used. For implementation purpose, it was desirable that the operator should not have to demonstrate the task more than a few times (5~10). This compromise has led to more parameters to be estimated compared to the amount of training data required.

The Baum-Welch (BW) algorithm [215], an HMM extension of Expectation-Maximization (EM) optimization algorithm, was used for the parameter estimation purpose. However, the algorithm cannot guarantee a global maximum and may become trapped in a local maximum of the likelihood function. Thus, the initialisations highly influence the model performance. Consequently, it is recommended to run *K-means* algorithm multiple times to guarantee a good convergence.

a) Data filtering strategy

The data from the F/T sensor, as shown in Figure 4-3, can be manually labeled into states (i.e. PiH phases namely approaching, insertion and extraction etc). As shown in Figure 4-3, the raw signal from the sensor contains noise. In this work, a Discrete Wavelet Transformation (DWT) is implemented as a filtering strategy as shown in equation (4-1),

$$X(a,b) = \frac{1}{\sqrt{a}}\int_{-\infty}^{\infty} \overline{\Phi\left(\frac{t-b}{a}\right)}\, x(t)dt \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots .. (4\text{-}1)$$

where a is a scaling factor, b is a time scaling factor, Φ is basis function, $x(t)$ is time-series input data. DWT is a powerful tool for extracting time-frequency domain features from time series data. Compared with other feature extraction strategies in time and frequency domains, DWT has a good time localization property, which suits on-line applications [45]. DWT splits the signal into an approximation and detail coefficients by passing it through complementary low- and high-pass filters. The approximation coefficients are further divided into a second-level approximation and detail coefficients. By repeating the process, one signal is broken down into many lower resolution components. Therefore, DWT can be used in the multi-resolution analysis where a different level of details will be reserved [216][217].

The selection of the elemental orthonormal basis wavelet function depends on the application. Typically, a ‘haar’ wave is widely used due to its simplicity, and it is closely related to the Piece-Wise Aggregation (PWA) algorithm. PWA is a symbolic representation of time-series data [218]. It reduces the dimensionality by re-representing the data by its average. However, DWT can be more useful than dimensionality reduction. By only allowing the basis function to change in time extension, but not the shape, both approximation and detail information of the original signal are preserved. Therefore, the signal can be de-noised by DWT. ‘Daubechies 10’ (*db10*) basis wavelet and the 6th level approximation of the original signals are chosen heuristically as the data filtering strategies. The shape of the force signal is smoothed after filtering. The reason behind this is to simplify the prediction because the wearable sEMG sensors cannot capture subtle force change caused by the muscle group. As shown in Figure 4-3 (the force along the z-axis contains the largest variations during the insertion phase). The spikes in the force signal were shown in the raw data are due to the initial contact and the large internal frictions when the operator is trying to react

and adjust his insertion angle. The features on the F_z profile are quite clear, but their shapes vary across subjects and demonstrations. The results will be shown in section 4.3. Due to the clearance (see 4.2.1), the jamming occurred when the operator was trying to align the peg with the hole, but as long as the peg has inside the hole it would slide in. Therefore, some of the frictions happened after the peg alignments could be handled easily by the robot by attempting insertion in vertical axis without the human knowledge.

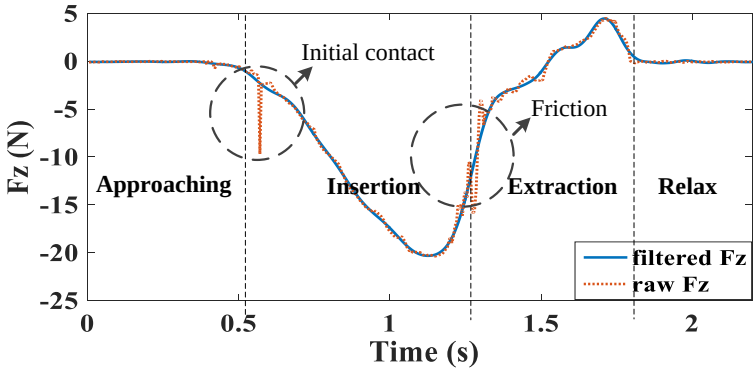


Figure 4-3 Fz signal from F/T sensor. The solid line is the filtered signal. The dotted line is the raw signal. The signal has been labeled into approaching, insertion, extraction and relax states.

b) Dynamic time warping (DTW)

It is unlikely that every demonstration was performed with the same time duration/length (data samples/sampling points). Even if two trials happen to have the same length, the underlying states may not align in the time domain. These misaligned signals can influence the results from GMM because it will have difficulties in producing a generalised representation of the data set. HMM can be also used for time series alignment, but DTW is used here due to its simplicity. It is a distance based approach to pre-process the data and then feed the output to the consequence steps. DTW was used in section 4.3.1 as a pre-alignment step. The algorithm is shown in Table 4-1.

Table 4-1 Algorithm: DTW

```
Input: S: Sequence of length n, Q: Sequence of length m.
Output: DTW distance.
1. Initialise D (i, 1) = iδ for each i
2. Initialise D (j, 1) = iδ for each j
3. For all i such that 2 < i < n do
4.     For all j such that 2 < j < m do
5.         $D(i,j) = d(i,j) + \min\begin{cases} D(i-1,j) \\ D(i-1,j-1) \\ D(i,j-1) \end{cases}$
6.     End for
7. End for
8. Return D (n, m)
9. [p,q] = Backtracing ( D ), where p, q are the indices of aligned signals
10. S_aligned = S (p)
11. Q_aligned = Q (q)
```

c) *Principle component analysis (PCA)*

PCA is a method to un-correlate the variables of a given signal and reorder them regarding their importance. PCA can also be used for data dimensionality reduction. Considering an M-dimensional dataset $X = X_1, X_2 \ldots, X_n$, where n is number of samples, the PCA will project all the observations based on their Eigen vectors and Eigen values to full or lower dimensions. The equation is shown in equation (4-2), where x is the observations, h is the latent states, Λ is a mixing matrix with dimension $M \times N$ $(N \leq M)$ that defines the rules of projection, and ϵ is a M dimensional zero-mean multivariate Gaussian noise vector with diagonal covariance matrix $\Psi = \sigma^2 I$, where I is a $M \times M$ matrix. Defining the model parameters to be $\theta = (\Psi, \Lambda)$, equation (4-3) can be derived by integrating out the factors. In PCA, only the co-variance between the variables are considered and re-ordered from the most important component to the least important component. As a consequence, the data dimensionality is reduced and new dataset is re-ordered based on their importance. The solution of the PCA can be found by Eigen decomposition of the covariance. Taking the limit of $\sigma \rightarrow 0$ of $p(h|x, \Lambda, \sigma)$, it is a delta-function at $h = \Lambda^T x$, which is the projection from x to principle components h. In this chapter, the PCA has been applied to the 6 axis force signals and the 1st PCA of the force vector (f_{pca}^1) which contains 98% percent of the information is used to represent the contact force.

$$x = \Lambda h + \epsilon \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4\text{-}2)$$

$$p(x|\theta) = \int p(h|\theta) p(x|h, \theta) dh = N(0, \Lambda\Lambda^T + \Psi) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4\text{-}3)$$

d) *Gaussian mixture model (GMM)*

After DTW and PCA, a dataset of N data points of dimensionality D, $X = \{\vec{x}_t\}_{t=1}^{N}$ with $x_t \in R^D$ was modelled by a multivariate Gaussian mixture of K components as described in equation (4-4).

$$p(\vec{x}_t) = \sum_{k=1}^{K} \pi_k \mathcal{N}(\vec{x}_t; \vec{\mu}_k, \Sigma_k)$$

$$= \frac{1}{\sqrt{(2\pi)^D |\Sigma_k|}} e^{-\frac{1}{2}(x_n - \vec{\mu}_k)^T \Sigma_k{}^{-1}(x_n - \vec{\mu}_k)} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (4\text{-}4)$$

where $\pi_k \in \Pi$ is the prior probability of the Gaussian component k, and $\mathcal{N}(\vec{x}_t; \vec{\mu}_k, \Sigma_k)$ is the D-dimensional Gaussian density of component k. $\vec{\mu}_k, \Sigma_k$ are the mean and covariance matrices of the multivariate Gaussian k. $\pi_k, \vec{\mu}_k, \Sigma_k$ were estimated using the Expectation Maximization (EM) algorithm. GMM could be considered as a model with latent variables, where K is the total number of component to represent the data set. The EM algorithm estimates the Maximization Likelihood parameters of a model with latent variables. Consider a model with variables x, latent variables h and parameters θ. The idea of EM is to repeat the Expectation and Maximization step in each timestamp, where the Expectation step predicts the distribution over the hidden states $q(h) = p(h|x, \theta_{t-1})$ where $h = \{h_k\}_{k=1}^{K}$ by fixing the model parameters and the Maximization step optimizes the parameter by fixing $q(h)$.

e) *Model parameters selection*

The optimal number of components K in the GMM model is not known beforehand. A method that evaluates the trade-off between optimizing the model's likelihood (a metric of how well the fitting is) and minimizing the number of parameters to estimate is needed. Even though this number can be learned heuristically, a formalized approach is preferred.

In order to select the optimal number of components K, a Bayesian Information Criterion (BIC) [219] is used after GMM in the benchmarking stage using equation (4-5).

$$S_{BIC} = -L + \frac{n}{2} log(N) \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (4\text{-}5)$$

Where L is the log-likelihood of the model, n is the number of free parameters required for a mixture of K components with a full covariance matrix, i.e. $n =$

$(K-1)+K\left(D+\frac{1}{2}D(D+1)\right)$. *N* is the number of D-dimension data points. The first term of the equation measures how well the model fits the data, while the second term has two parts: the number of parameters to estimate in the transition matrix and the observation densities.

f) Hidden markov model

A Hidden Markov Model [214] uses a mixture of multivariate Gaussians to describe the distributions of the observations. The temporal variations are also encapsulated in the transition probabilities. Let *x* denote the hidden state of an HMM at time *t*; the model can be written in equation (4-6).

$$p(x_{1:T}, h_{1:T}|\theta) = \prod_{t=1}^{T} p(h_t|h_{t-1},\theta)p(x_t|h_t,\theta) \quad \text{(4-6)}$$

Let $\{\Pi = \mathrm{p}(x_0),\ T = p(h_t|h_{t-1},\theta),\ E = p(x_t|h_t,\theta)\}$ be the initial guess of the state's distribution, the transition probabilities between the states or components and the multivariate data distribution respectively. $\{\vec{\mu}_k, \Sigma_k\}_{k=1}^{K}$is returned from the GMM and is directly used for initializing the HMM. The prior state distribution *Π* can also be suggested by the GMM. Therefore, the HMM only needs to estimate the state transition probabilities *T* and refine the parameters of *Π* and *E*. The Baum-Welch algorithm is used to estimate those parameters [215].

4.2.3 Motion reproduction

After GMM/HMM encoding, a Gaussian Mixture Regression was applied to reproduce a smooth trajectory, which is inferred from the generalized model. For a D-dimension variable $\vec{x} \in R^D$, the means and covariance matrices given by GMM/HMM encoding for component k are given by $\vec{\mu}_{kX}^{H}$, and Σ_{kX}^{H}. The regression is done along the time index. The means and covariance matrices of the set of observations $\{t, \vec{x}_t\}$ with dimension (*D+1*) were computed. Here, only the time-indexed means and covariance matrices were estimated because the rest of the means and covariance matrices $\{\vec{\mu}_{kX}^{R},\ \Sigma_{kX}^{R}\}$ have already been estimated using:

$$\vec{\mu}_k^R = \{\vec{\mu}_{kt}^R, \vec{\mu}_{kx_1}^H, \vec{\mu}_{kx_2}^H, \dots, \vec{\mu}_{kx_D}^H\}$$

$$\Sigma_k^R = \begin{pmatrix} \Sigma_{kt}^R & \Sigma_{ktX}^R \\ \Sigma_{kXt}^R & \Sigma_{kX}^H \end{pmatrix}$$

The GMR then estimates the generalised trajectory by computing $E[p(x_t|t)]$ and the covariance is calculated by $E[cov(p(x_t|t))]$ as shown in equations (4-7-4-10):

$$\vec{x}^d(t) = \sum_{k=1}^{K} \beta_k(t)\vec{x}_k^d(t) \quad \text{(4-7)}$$

$$\beta_k(t) = \frac{\pi_k \mathcal{N}(t;\vec{\mu}_{kt}^R, \Sigma_{kt}^R)}{\sum_{i=1}^{K} \pi_i \mathcal{N}(t;\vec{\mu}_{it}^R, \Sigma_{it}^R)} \quad \text{(4-8)}$$

$$\vec{x}_k^d(t) = \vec{\mu}_{kX} + \Sigma_{kXt}^R \Sigma_{kt}^{R\,-1}(t - \mu_{kt}) \quad \text{(4-9)}$$

$$\Sigma_{xx,k}^d = \Sigma_{xx,k} - \Sigma_{xt,k}\left(\Sigma_{tt,k}\right)^{-1}\Sigma_{tx,k} \quad \text{(4-10)}$$

To better explain the GMR, the following symbol representation is used:

$\begin{bmatrix} \xi^I \\ \xi^O \end{bmatrix}$ represents the input variables and output variables. $\begin{bmatrix} \mu^I \\ \mu^O \end{bmatrix}$ represents the input and output mean values. $\begin{bmatrix} \Sigma^I & \Sigma^{IO} \\ \Sigma^{OI} & \Sigma^O \end{bmatrix}$ represents the covariance matrix for input and output variables. The target orientations ξ^d and $\Sigma_{\xi\xi}^d$ are then estimated by equation (4-11) and (4-12).

$$\xi^d = \sum_{k=1}^{K} \beta_k(\xi)(\mu_k^O + \Sigma_k^{OI}(\Sigma_k^I)^{-1}(\xi^I - \mu_k^I)) \quad \text{(4-11)}$$

$$\Sigma_{\xi\xi}^d = \sum_{k=1}^{K} \beta_k{}^2(\xi)(\Sigma_{\xi\xi,k} - \Sigma_{\xi t,k}\left(\Sigma_{tt,k}\right)^{-1}\Sigma_{t\xi,k}) \quad \text{(4-12)}$$

4.3 Result analysis

The skills encoding results are shown and discussed in this section. As shown in Figure 4-4, the force and hand motion signals shared similar patterns. However, even though the operator was trained to deliver such repetitive skill pattern, there are variations in the hand pose and force profiles in the different states (i.e. approaching, insertion. etc). Misalignments in the time line also occurred in the demonstrations. The proposed skills encoding methods need to address these variations therefore 5 out of 10 trials were selected as training samples, the rest of the trials were used as testing samples. After selection of the trials, the skills encoding results from individual subject are discussed in sections 4.3.1a)-b). Then the recognised states are explained in section 4.3.1.c). The evaluations of the results across different subjects are discussed in section 4.3.2. Finally, the motion reproduction results using the GMR are discussed in section 4.3.3.

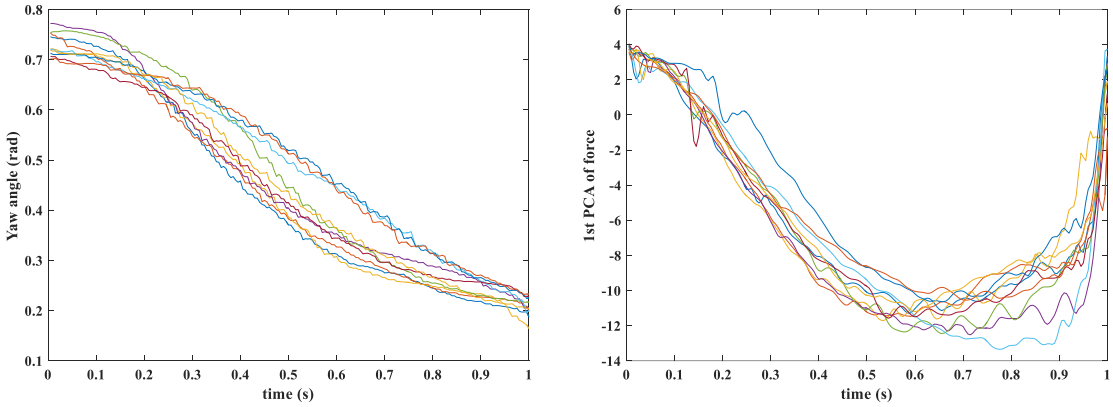


Figure 4-4 10 trials of demonstrations from subject B. The yaw angles (left) and the 1st PCA of force (right) are plotted for visualisation.

4.3.1 Skills encoding for individual subject --- level 1 encoding

By using the proposed skills encoding method, each person's skill was encoded, and the results are shown in Figure 4-6 - Figure 4-11. Figure 4-5 shows the BIC plot against the number of hidden states (N_h) from a demonstration by Subject A. As shown in the plot, when the number of the states increased beyond four, the overall performance did not improve by much. A model with four states were selected because it was very close to the optimal, therefore represents a good compromise to reduce the number of parameters to estimate. The same method was applied to the rest of the subjects, which may have a different optimal number of states.

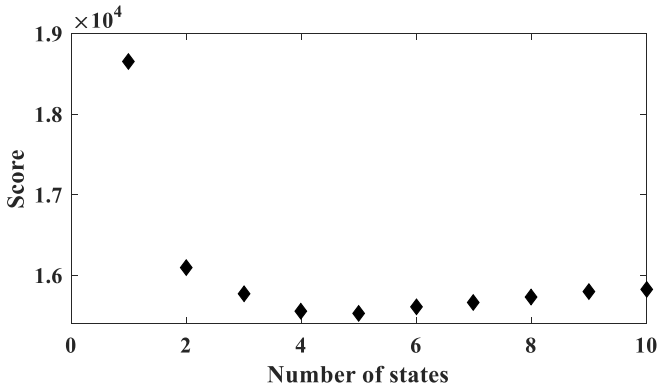


Figure 4-5 BIC plot for subject A.

The generalisation results across different trials for each subject are shown in Figure 4-6 - Figure 4-11 and Table 4.2 - Table 4.7. The solid curves are the trajectories reproduced by GMR. The ellipse is the visualisation of the 2×2 covariance matrix representing the covariance in between the time and the variables. Here, the variables are [f_{pca}^1, *roll*, *pitch*, *yaw*]. From the results, a number of findings can be summarised below:

i. The force variations in each state were different. For instance, in the 1st PCA of force signal by subject A, the Gaussian ellipse for state 1 (*0.113*) was narrower than the other states. This shape indicated that subject A was applying more consistent force with fewer variations in state 1. But, later as the subject left state 1, the variations in the force applied increased, which indicated that subject A was working in a consistent manner. This means the skills in the later states were complicated and harder to generalise for subject A. However, the situation changed in case of subject B where the variations in state 3 (*1.196*) were smaller for the 1st PCA of the force signal. This indicated that state 3, in this case, are more certain/controlled than the other states. Subject B was working more repetitive at state 3. For subject C, the variations started with a bigger value (*7.2698*) then a smaller value in the middle and a bigger value in the end. For subject D, the variations started with a small value (*0.209*) then a bigger value at state 2, a smaller value at state 3 and 4 and a bigger value in state 5. For subject E, the variation started with a bigger value (*27.154*) then a smaller value in the middle and a bigger value in the end. For subject F, the variation started with a smaller value (*1.723*), then a bigger value in the middle and the end. In general, all subjects seem to demonstrate different levels of control or consistency cross the different states.

ii. The hand pose variations in each state were different. The hand pose was the composition of the yaw, pitch and roll values. The overall variations in the hand poses were small which indicate that the operator was performing the task in a very consistent manner. This indicated that in the fine manipulations, hand positions should not change much and the force based control is more useful. The first state encoded the uncertainties of the initial orientation of the hand from the time the peg initially contacted the hole and started sliding. The uncertainties in each state were different depending on the subjects. However, the overall trend of the hand pose variations was small at the beginning, bigger in the middle state and smaller in the last state. This indicated that the subject demonstrates more sophisticated movement in the middle of the state where the frictions were overcome by adjusting the peg angles.

iii. A small variation in f_{pca}^{1} did not mean a small variation in hand pose. This variance can be confirmed by the result in subject A. In subject A, the operator

was trying to align the peg with the hole (transition from state 2 to state 3) and encountered small resistence force. Therefore, the force signal contained little variation in state 3 (*3.755*), but its yaw variance (*0.0082*) was larger than the yaw variance in state 2 (*0.005*).

In conclusion, the individual subject demonstrated a consistent pattern in the PiH task but with variations in the various state of subject's performance. Different operators showed some consistency in the force and motion patterns but the variations in the same state (i.e. insertion) are different. The less variations in state 1 (except subject D) indicated that people were performing a similar initial strategy in the chamfer crossing step. More uncertainties were generated and controlled by the operators in the rest of the states. Those uncertainties are necessary for the robot to learn the skills and reproduce reliable control for complex tasks.

Table 4-2 Skills encoding for subject A. State variances result

	State 1	State 2	State 3	State 4
1st PCA of force signal(N)	0.113	10.770	3.755	14.4934
Yaw (degree)	0.11	0.29	0.47	0.04
Pitch (degree)	0.01	0.05	0.28	0.09
Roll (degree)	0.02	0.07	0.06	0.04

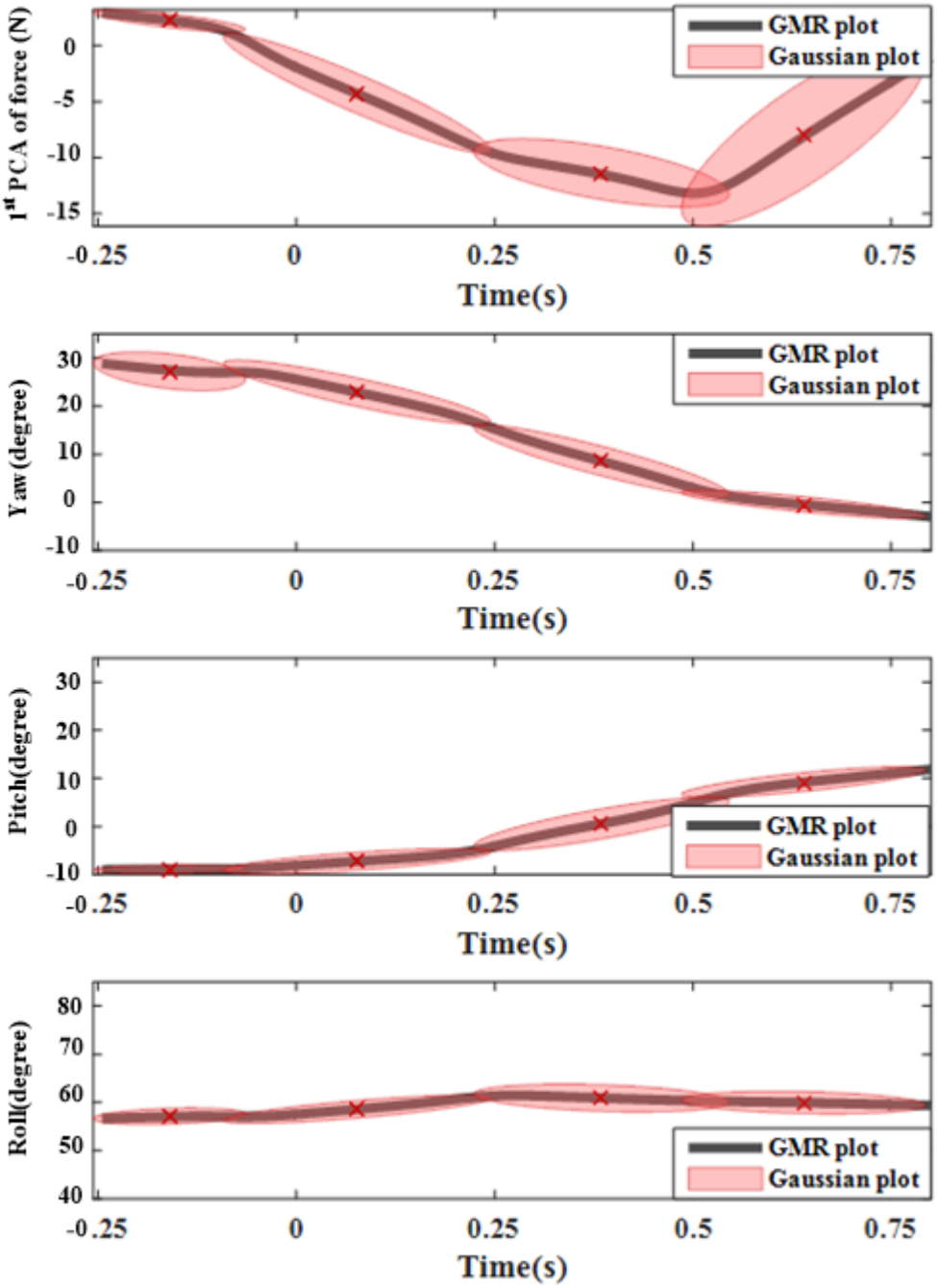


Figure 4-6 Skills encoding for subject A. Gaussian and GMR plots.

Table 4-3 Skills encoding for subject B. State variances result

	State 1	State 2	State 3	State 4
1st PCA of force signal(N)	3.450	12.34	1.1967	13.963
Yaw(degree)	0.08	0.59	0.12	0.05
Pitch(degree)	0.06	0.2	0.09	0.06
Roll(degree)	0.40	0.2	0.06	0.02

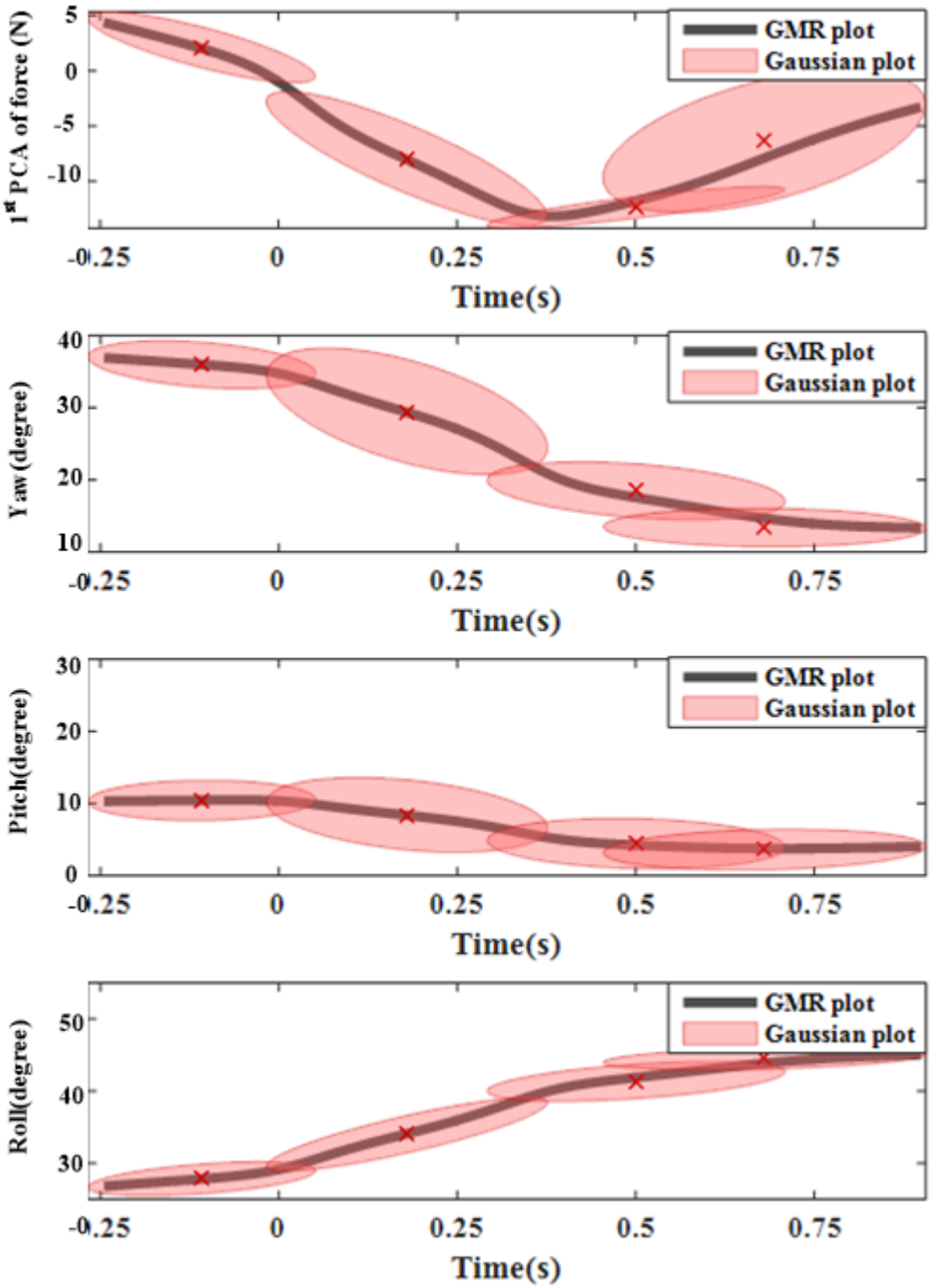


Figure 4-7 Skills encoding for subject B. Gaussian and GMR plots.

Table 4-4 Skills encoding for subject C. State variances result

	State 1	State 2	State 3	State 4
1st PCA of force signal(N)	7.2698	5.6936	5.951	11.6
Yaw(degree)	0.06	0.30	0.14	0.17
Pitch(degree)	0.06	0.14	0.05	0.06
Roll(degree)	0.04	0.33	0.09	0.11

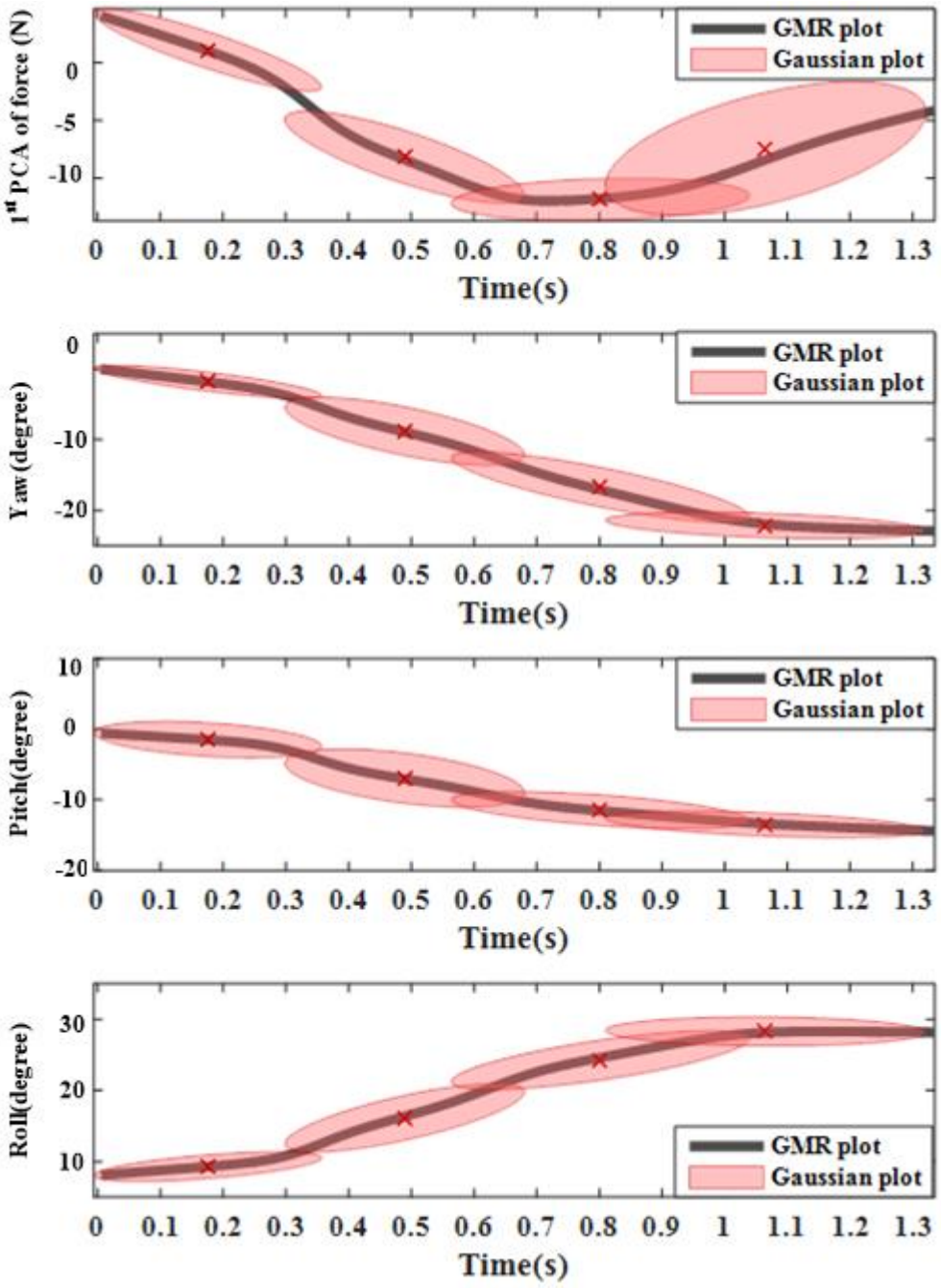


Figure 4-8 Skills encoding for subject C. Gaussian and GMR plots.

Table 4-5 Skills encoding for subject D. State variances result

	State 1	State 2	State 3	State 4	State 5
1st PCA of force signal(N)	0.209	7.622	1.822	1.533	12.604
Yaw(degree)	0.01	0.09	0.11	0.19	0.04
Pitch(degree)	0.29	0.09	0.10	0.09	0.07
Roll(degree)	0.03	0.05	0.08	0.11	0.01

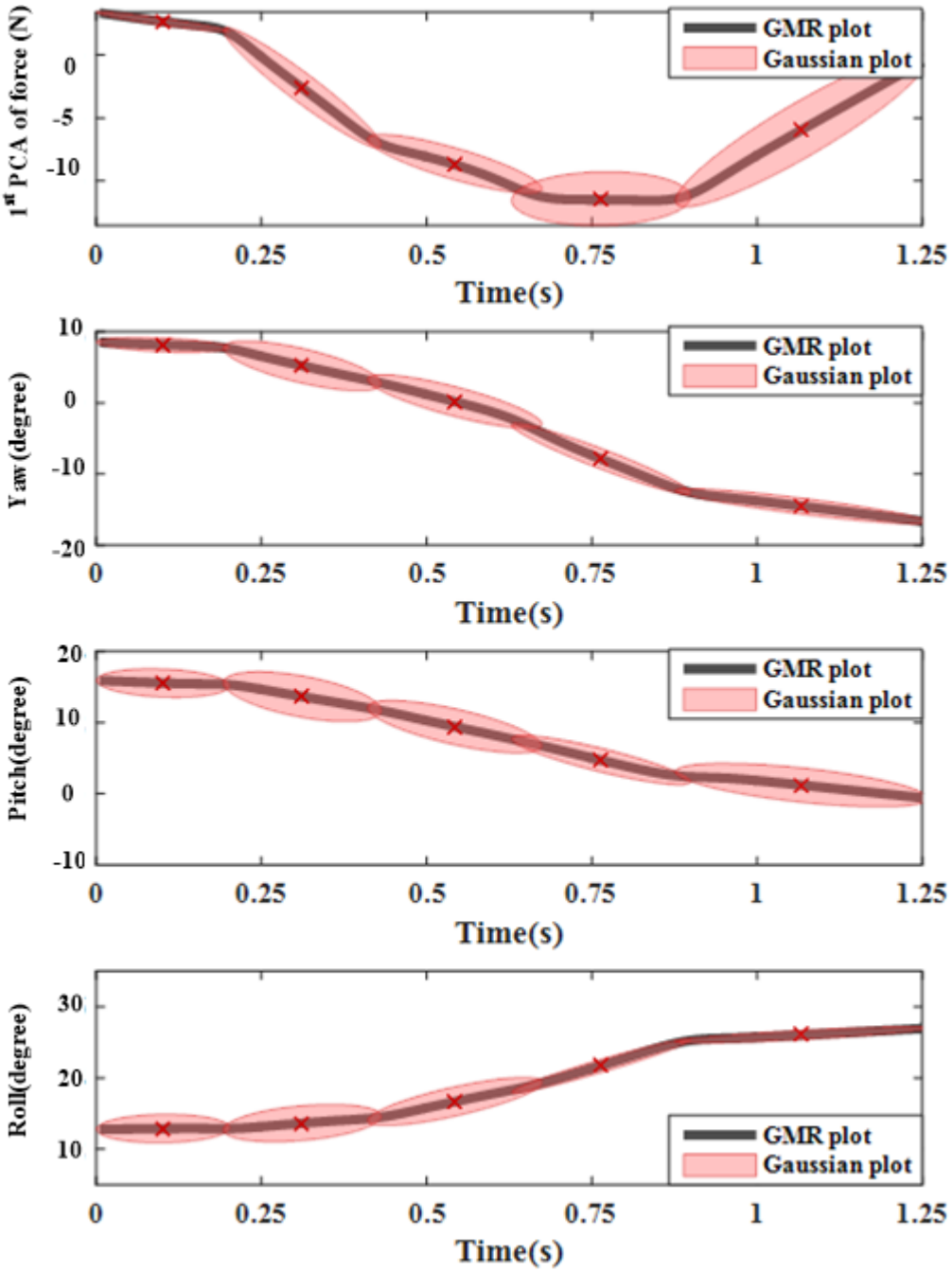


Figure 4-9 Skills encoding for subject D. Gaussian and GMR plots.

Table 4-6 Skills encoding for subject E. State variances result

	State 1	State 2	State 3
1st PCA of force signal(N)	27.154	3.9	8.866
Yaw(degree)	0.07	0.09	0.02
Pitch(degree)	0.31	0.35	0.05
Roll(degree)	0.34	0.08	0.01

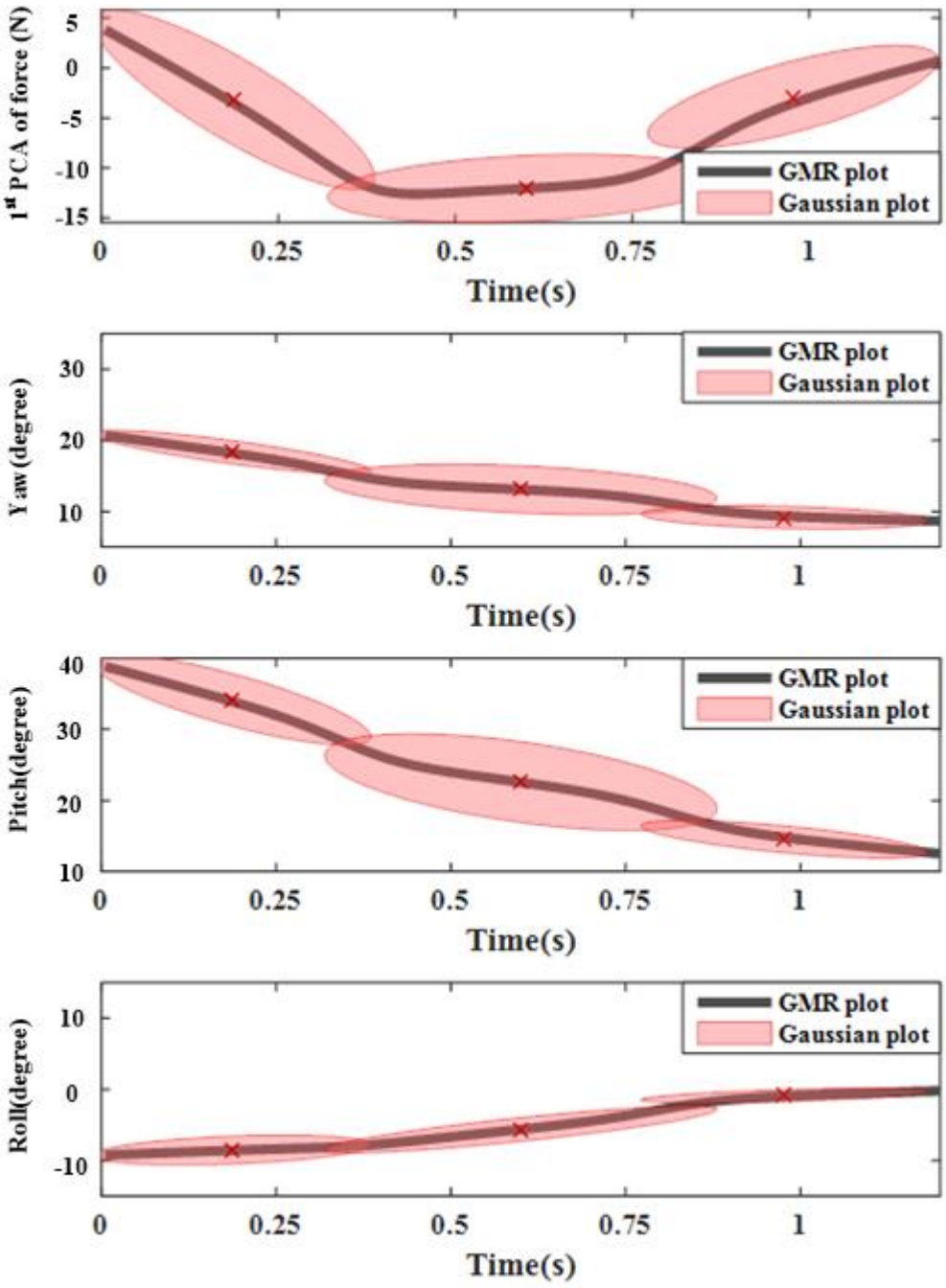


Figure 4-10 Skills encoding for subject E. Gaussian and GMR plots.

Table 4-7 Skills encoding for subject F. State variances result

	State 1	State 2	State 3	State 4
1st PCA of force signal(N)	1.723	14.103	12.737	17.643
Yaw(degree)	0.01	0.29	0.21	0.04
Pitch(degree)	0.01	0.05	0.03	0.01
Roll(degree)	0.02	0.07	0.12	0.017

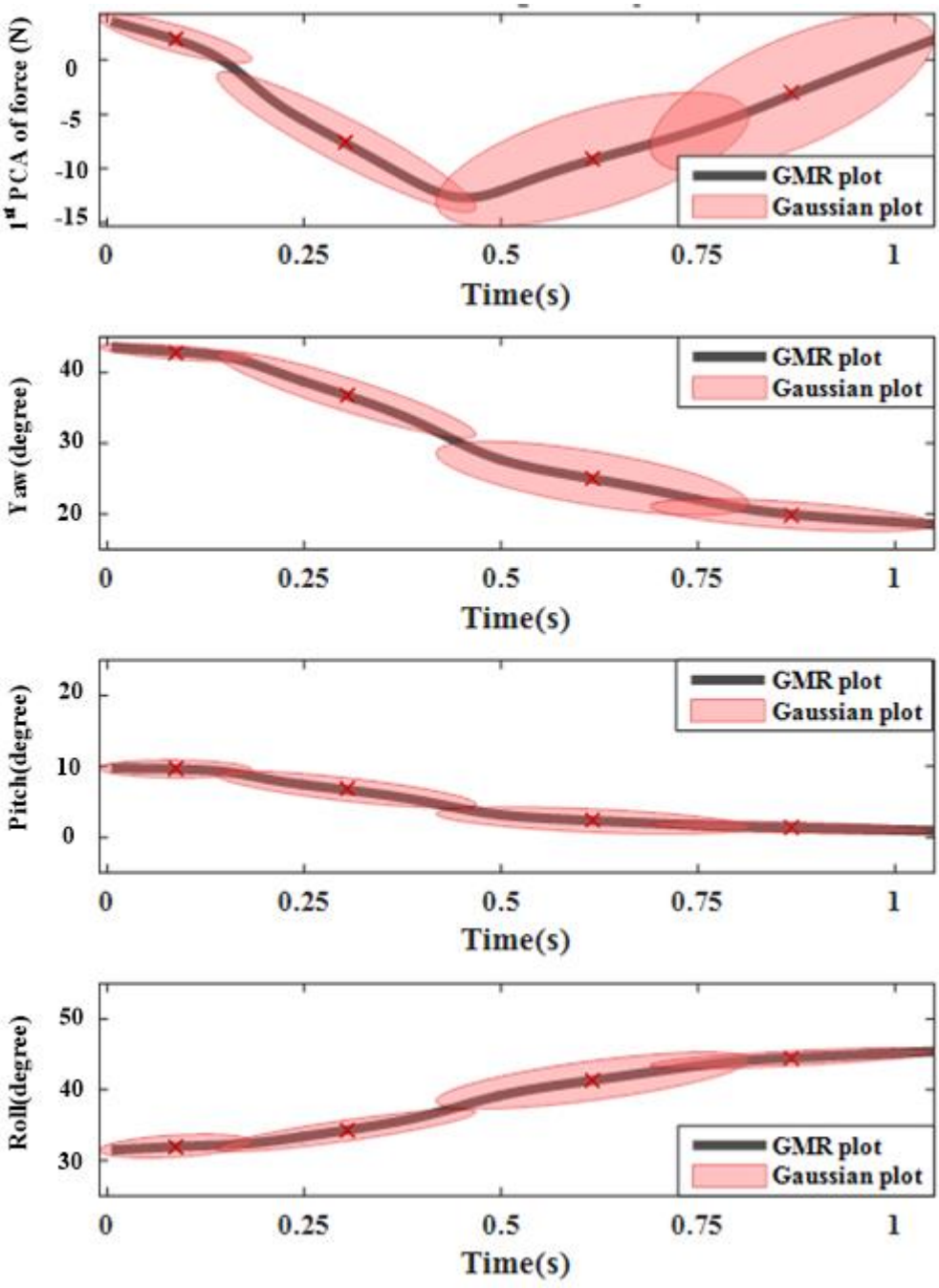


Figure 4-11 Skills encoding for subject F. Gaussian and GMR plots.

a) Changing of the reference signal

In order to align the timeline across different demonstrations, one trial had to be selected as a reference. The selection of the reference signal was currently heuristically decided. In this section, results from subject A were used for discussion purpose. Results in Table 4-8 (Model 1–5) showed the variations by choosing different demonstration from the 5 trials as the reference signal. The results indicated that the reference signal had an impact on the statistics in each state. The 1st PCA of force signal in Model 1, 4 and 5 started with small variances in state 1 (*0.113 N, 0.19 N, 0.24 N*). Model 2 and 3, on the other hand, had larger variations in state 1 which is not preferred because from practical point of view the initial state of the robot can be well defined with small variation. These results indicated that by selecting different reference signal, the skills may change even for the same subject. When these skills are transferred to the robot, the robot would behave closer to the reference model. There are ways to select the reference signal. One can simply use the first trial as the reference signal and assume the rest of the trials come in sequence. Also, the new reference signal can be generated by applying the GMR to the learned sequences. Therefore, the new tests will be aligned to this new reference. As alternative, the reference signal that generates a model, which contains fewer uncertainties in the initial state, can be selected. Therefore, model 1 is chosen for subject A.

In this section, skills encoding results by selecting different reference trials were discussed. Subject A was used as an example. The discussion provided a way to justify the trial that needed to be selected. The reference signal should produce smaller variations in the initial state.

Table 4-8 skills encoding result by choosing different reference signal from subject A. Values are standard deviations in each state.

	State 1	State 2	State 3	State 4
Model 1				
1st PCA of force signal(N)	**0.113**	**10.770**	**3.755**	**14.4934**
Yaw (degree)	**0.11**	**0.29**	**0.47**	**0.04**
Pitch (degree)	**0.01**	**0.05**	**0.28**	**0.09**
Roll (degree)	**0.03**	**0.07**	**0.06**	**0.04**
Model 2				
1st PCA of force signal(N)	**11.524**	**3.806**	**1.256**	**21.139**
Yaw (degree)	**0.26**	**0.52**	**0.05**	**0.05**
Pitch (degree)	**0.03**	**0.25**	**0.04**	**0.11**
Roll (degree)	**0.04**	**0.05**	**0.06**	**0.03**
Model 3				
1st PCA of force signal(N)	**8.962**	**3.62**	**2.11**	**14.747**
Yaw (degree)	**0.22**	**0.39**	**0.23**	**0.03**
Pitch (degree)	**0.01**	**0.09**	**0.16**	**0.09**
Roll (degree)	**0.03**	**0.04**	**0.08**	**0.03**
Model 4				
1st PCA of force signal(N)	**0.19**	**10.321**	**3.922**	**16.344**
Yaw (degree)	**0.13**	**0.36**	**0.46**	**0.03**
Pitch (degree)	**0.01**	**0.06**	**0.29**	**0.10**
Roll (degree)	**0.02**	**0.06**	**0.07**	**0.04**
Model 5				
1st PCA of force signal(N)	**0.24**	**10.678**	**2.996**	**22.11**
Yaw (degree)	**0.12**	**0.35**	**0.49**	**0.06**
Pitch (degree)	**0.01**	**0.05**	**0.27**	**0.08**
Roll (degree)	**0.02**	**0.07**	**0.06**	**0.04**

b) Changing the number of demonstrations

The model generalisation performance is discussed in this section by varying the number of demonstrations involved. The results, as shown in Figure 4-12, from subject B and C were used for discussion purpose. Log-likelihood is used for evaluating the model performance. From the results, the generalization performance of the model improves with the number of demonstrations. Especially when only one demonstration was used for training, the model could not fit the data from the rest of the demonstrations (it only fits the data to some degree with low log likelihood). On the other hand, for subject B, when the number of demonstrations increased to 4, the model showed good generalisation capability on the 5^{th} demonstration. But the same result was not shown in subject C. Therefore, the learned model could gradually increase its generalisation capability as long as the demonstrations were reasonably consistent. After identification of the demonstrations with good consistency, the learned model can

be used to select the trials with similar consistency from the new demonstrations. This method can also be used to select the reference signals as discussed in section 4.3.1a).

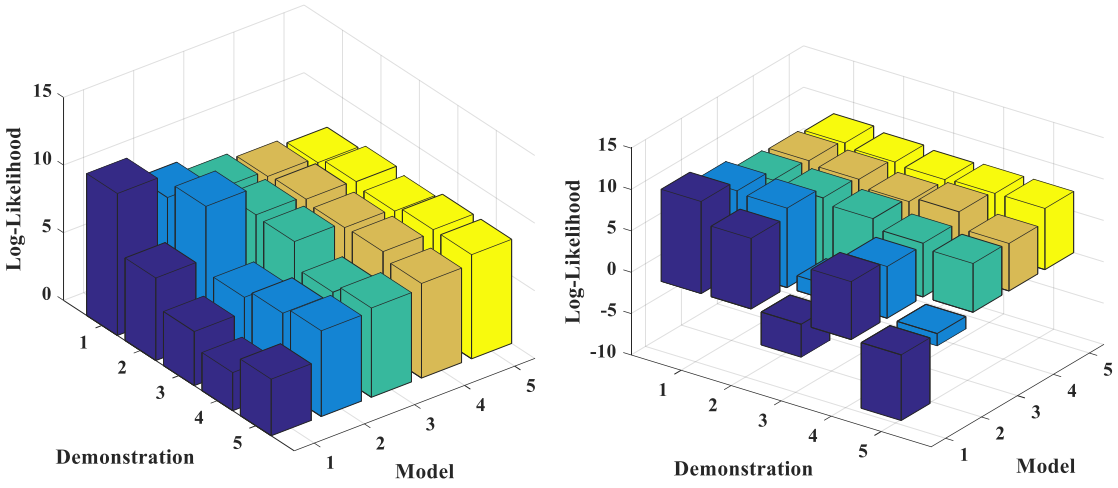


Figure 4-12 Model generalization across different demonstrations for subject B (left) and C (right)

c) Interpretation of the individual skills

The previous results showed skills encoding results for individual subject. However, what is happening in each state was unclear. In this section, the results from individual subject was first compared, and then the states were interpreted.

The results in Table 4-9 stored the mean variance of the states for each variable for individual subject. Subject D had the least variation in the 1st PCA of force signal, subject E had the least variation in the yaw angle, subject C had the least variation in the pitch angle, and subject E had the least variation in the roll angle. In general, subject D was showing the best performance in applying force with $\sigma_{mean} = 4.758\,N$. This result indicated that he was performing the task in a more repetable way. Subject A, B and C were showing similar σ_{mean} in applying force. Subject E and F had larger variations in the force applied with $\sigma_{mean} = 13.311\,N$ and $\sigma_{mean} = 11.551\,N$ respectively. The σ_{mean} in all hand pose was small which indicated that the subjects were performing very repetitive movement while inserting the peg.

Although different subject shows differences in performing the task, there are common features across all the subjects as shown in Figure 4-6 - Figure 4-11. The PiH can be classified into three major phases wherein the first phase the operator attempted to insert the peg from the hole chamfer and sliding the peg into the hole. This motion led to the dominant pressing down force (Fz) and the slope curvature in the plot. In the second phase, the peg was already inside the hole, but the lateral angles still needed to

be adjusted (wobbling movement) so that the peg was well aligned with the hole. In phase three, as the peg had already been aligned with the hole, the operator reduced the force and guides the peg into the hole. The same pattern could be observed in all demonstrations. The difference was some phases contain two states. For example, states 1 and 2 were accounting for the phase 1 in subject A, B, and C.

In conclusion, the generalised trajectories from all the subjects contained the same phase. However, each phase might have a different number of states. The human operator can interpret the phase; however, it is difficult to explain the meaning of the hidden states further. More in-depth human factor analysis could be performed to interrogate the hidden information behind each state. For LfD, the probabilistic models contain the knowledge to transfer human skills to robot control, even without fully interpreting the meaning of each state.

Table 4-9 Skill encoding comparison results between subjects. Values are average standard deviations.

	Subject A	**Subject B**	**Subject C**	**Subject D**	**Subject E**	**Subject F**
1st PCA of force signal(N)	**7.283**	**7.737**	**7.63**	**4.758**	**13.311**	**11.551**
Yaw (degree)	**0.24**	**0.21**	**0.13**	**0.09**	**0.06**	**0.14**
Pitch (degree)	**0.11**	**0.10**	**0.07**	**0.08**	**0.24**	**0.02**
Roll (degree)	**0.04**	**0.08**	**0.12**	**0.05**	**0.04**	**0.06**

4.3.2 Model generalisation across subjects --- level 2 encoding

In this section, the results from model generalisation across all the subjects are shown in Figure 4-14-Figure 4-17. The BIC result in Figure 4-14 indicated more states to encode the task. $k = 5$ was selected, and the results are shown in Figure 4-16 and Figure 4-17. When $k = 5$, the state contained considerable overlap which indicated more confusion at the cross sections of the two states. Therefore, the situation when $k > 5$ was no longer considered. The explanation of the overlapping of the Gaussian is that the different skills are encoded into similar Gaussians. Also, since the IMU-Vicon system needs to reinstall from one subject to another, it is likely that the sensor position will vary between subjects. Therefore, the larger variations in the hand pose are anticipated (as shown in Table 4.10 and Figure 4-16). This will lead to the difficulties for the robot to learn because the behaviour contained more uncertainties. The reconfiguration of the IMU-Vicon system could be resolved by asking the subject to start with a pose where the peg is placed vertically on the flat surface and held firmly

by fingers as shown in Figure 4-13 with the hand assumed as a rigid body. Then the orientation measurements needed to use this pose as an origin. After this process, the variations left were only different movement strategies by the subjects. The results, as shown in Table 4.11 and Figure 4-17, indicated that after the alignment procedure, the variations in the force profiles did not change, however, the variations in the pose trajectories were smaller than before. Therefore, the alignment procedure had improved the model generalisation process. The model generalisation performances across different subjects by gradually increasing the number of the participants are shown in the results in Figure 4-15. As shown in the figure, when the model was built from demonstrations by one subject, it may fail to model the demonstrations by other subject. When more subjects were involved, the model tended to fit better.

Figure 4-13 Initial hand pose by placing the peg vertical on the flat table.

The capabilities to generalise the PiH task across all the subjects are discussed in this section. The result contained more states because individuals tended to have different ways of completing the task. Since initially the subject was not asked to start from the same pose as suggested, the results showed larger variations in the hand pose. A method to compensate this variation was proposed, where the results showed a better performance. The results also showed the generalisation capability of the model by increasing the number of subjects used for training the model. The level 2 encoding proposed in this section provided the possibilities to expand the skills knowledge base. However, as shown from the results, the model contained more complexity compared with the individual model. The advantage is the model could adapt to situations that are

more complex. The disadvantage is the model might lead to a failure to complete the task because of uncertainties. The next section will evaluate the ability of the model to reproduce motions for the robot.

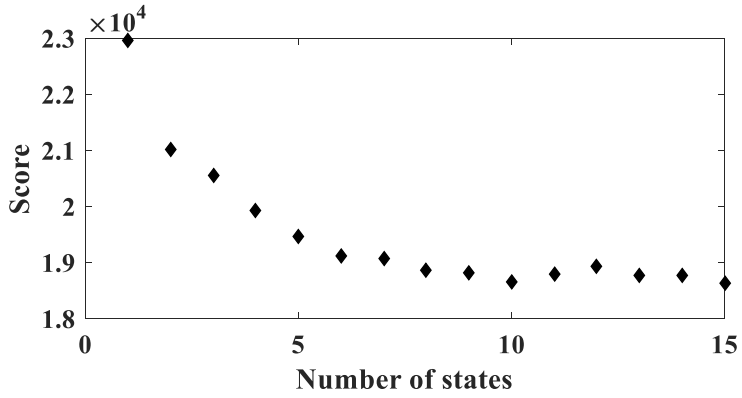


Figure 4-14 BIC plot for all subjects.

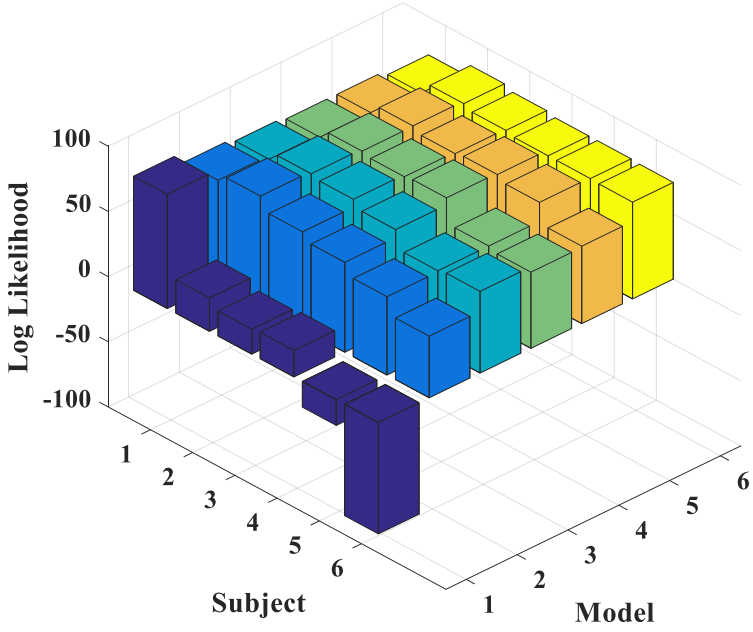


Figure 4-15 Model generalisation performances across different subjects

Table 4-10 Skills encoding from all subjects (k = 5). State variances result

	State 1	State 2	State 3	State 4	State 5
1st PCA of force signal(N)	5.596	13.443	0.433	17.301	12.273
Yaw (degree)	7.03	6.43	7.6	4.28	6.83
Pitch (degree)	1.09	4.15	3.25	2.48	0.01
Roll (degree)	1.72	6.43	6.59	9.40	1.95

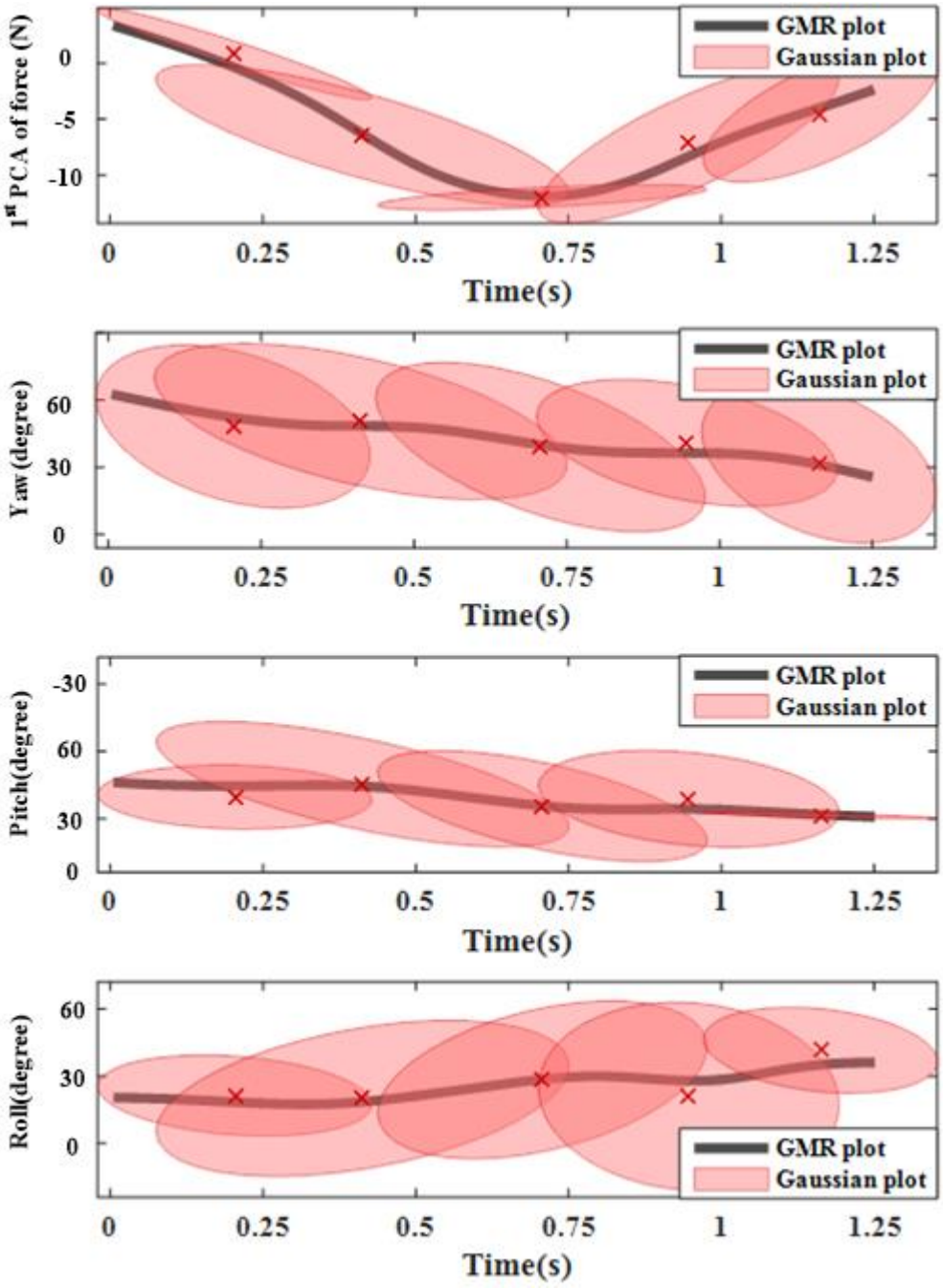


Figure 4-16 Skills encoding from all subjects (k = 5). Gaussian and GMR plots.

Table 4-11 Skills encoding from all subjects after alignment (k = 5). State variances result

	State 1	State 2	State 3	State 4	State 5
1st PCA of force signal(N)	22.4559	15.2192	5.7816	9.3314	20.7683
Yaw (degree)	0.14	8.59	0.25	0.11	0.54
Pitch (degree)	0.18	0.17	0.25	0.10	0.17
Roll (degree)	0.25	1.66	0.32	0.22	1.07

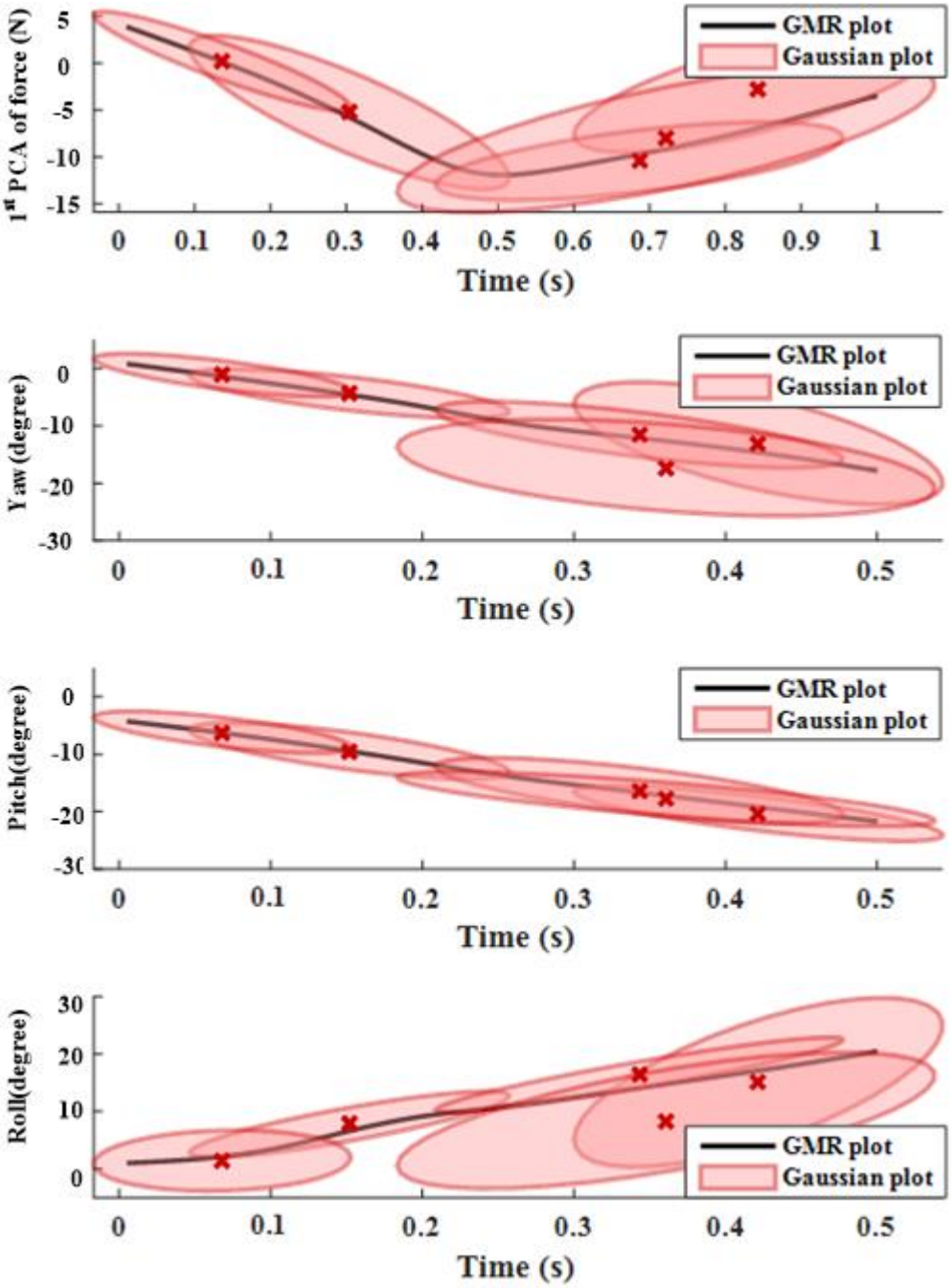


Figure 4-17 Skills encoding from all subjects after alignment (k = 5). Gaussian and GMR plots.

4.3.3 Motion reproduction for robot

a) Explicit time sequence based motion reproduction

From the previous sections, the probabilistic models were learned from the individual subject and extended to all the subjects. This section illustrates how the robot can use the learned probabilistic models to reproduce the motion command by using explicit time sequence. The model had not been applied to the real robot because the human teacher and robot learner have different kinematics. For evaluation purpose, an identical physical mapping from human to the robot is assumed so that human demonstrations data could be used to evaluate the model ability to generate the motion path based on the force signals. Five test data sets from the same subject were used as simulation signals. The model produced from subject B is used for discussion in this section (The summarised results are shown in Appendix A). The input signal $\xi^I = \{t, f_{pca}^1\}$ are the time sequence and the 1st PCA of the force signal. The output signals are $\xi^O = \{roll, pitch, yaw\}$. The influence of t and f_{pca}^1 on the desired ξ^O trajectories are discussed in this section. The simulated hand motion was generated from ξ^I and compared with the true ξ^O signal. The mean square error (MSE) and regression coefficient (*R*) values were used for evaluating the estimation performance.

From the results in Table 4-12, the following findings can be made:

i. Using time sequence only approach, the model could reproduce smooth motion trajectories. The *MSE* were *1.19±0.66 degree* for yaw, 16.40±*11.59 degree* for pitch and *2.71±1.49 degree* for roll. The *R* are *0.9945±0.0071* for yaw, *0.9869±0.0059* for pitch and *0.0473±0.0261* for roll. Since the time variable is fixed (due to DTW alignment) for all cases, the reproduced results were actually the mean values of the hand motions from the training model. The small *MSE* and large *R* indicated that the generalised trajectory from the learned model was close to the new test samples.

ii. The time and force sequence could reproduce smooth trajectories. The predicted ξ^O trajectories for each trial were different from time sequence only model because of the influence of the force signal (see Figure 4-18-Figure 4-22, the Blue solid line represents the target signal. The green dotted line represents the estimation signal generated from *t* and *f*. The red dashed line represents the predicted signal generated from *t*). The *MSE* were *1.10±0.72*

degree for yaw, *23.54±23.44 degree* for pitch and *1.76±0.69 degree* for roll. The *R* were *0.9935±0.0038* for yaw, *0.9735±0.0283* for pitch and *0.9923±0.0047* for roll. The small *MSE* and large *R* indicated that the reproduced trajectory based on the time and force sequences were close to the new test samples.

iii. The Gaussians have defined the boundaries for the hand motions. This can be confirmed in trial 1, 3 and 4. The pitch angles from the test demonstrations were outside the scope of the training model. But the reproduced motions were within the Gaussian boundaries. The advantage of this boundary is that the robot will not move to unexpected area that had not previously been demonstrated by a human. Similarly, the robot cannot deal with uncertainties that have not been captured during training (outside of the Gaussian plots). This is the reason why a larger error was found on pitch angle estimations in Figure 4-20 suggesting the model cannot accurately predict the motion trajectories.

Table 4-12 Motion reproduction results for subject B. 5 trials were used. The influences of the input variables are compared by using *MSE* and *R* values.

Trial	Input Variable	Yaw (degree)		Pitch (degree)		Roll (degree)	
		MSE(degree)	R	MSE(degree)	R	MSE(degree)	R
1	[*t*]	2.32	0.9818	17.25	0.9824	2.99	0.9834
	[t,f]	2.16	0.994	10.28	0.987	1.16	0.992
2	[*t*]	1.07	0.9979	6.93	0.9967	1.66	0.9959
	[t,f]	0.40	0.9976	8.39	0.9943	2.19	0.9927
3	[*t*]	0.54	0.9978	2.58	0.9824	5.14	0.9887
	[t,f]	0.58	0.9963	1.15	0.9911	2.68	0.9969
4	[*t*]	1.01	0.9982	25.74	0.9865	1.40	0.9959
	[t,f]	1.49	0.9881	50.77	0.9259	1.71	0.9848
5	[*t*]	1.00	0.9966	29.44	0.9863	2.34	0.9978
	[t,f]	0.87	0.9917	47.12	0.969	1.05	0.9953
All	[*t*]	1.19±0.63	0.9945±0.007	16.40±11.58	0.9869±0.005	2.69±1.49	0.9923±0.006
	[t,f]	1.10±0.69	0.9935±0.003	23.54±23.44	0.9735±0.028	1.76±0.69	0.9923±0.004

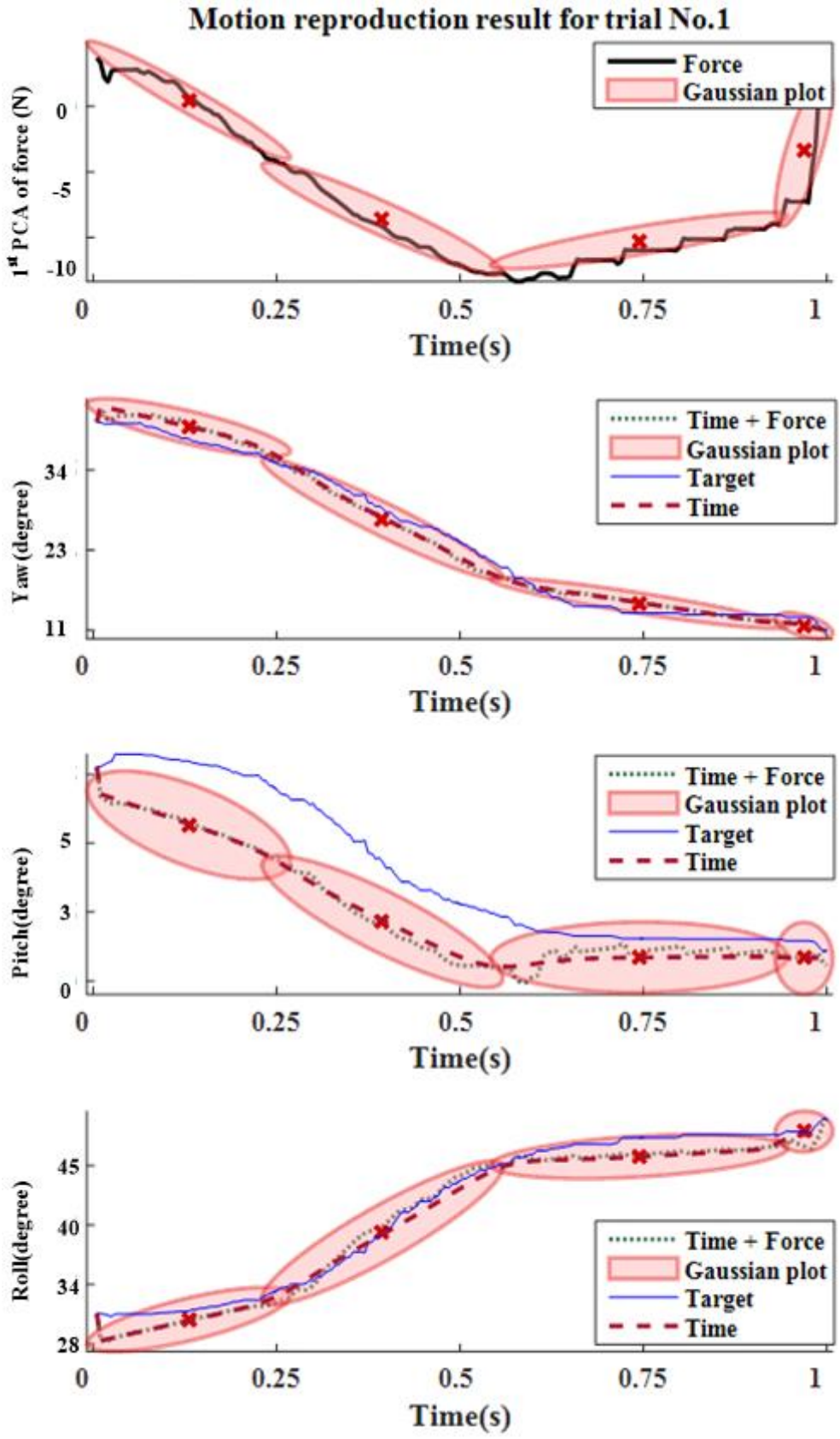


Figure 4-18 Motion reproduction result for trial 1.

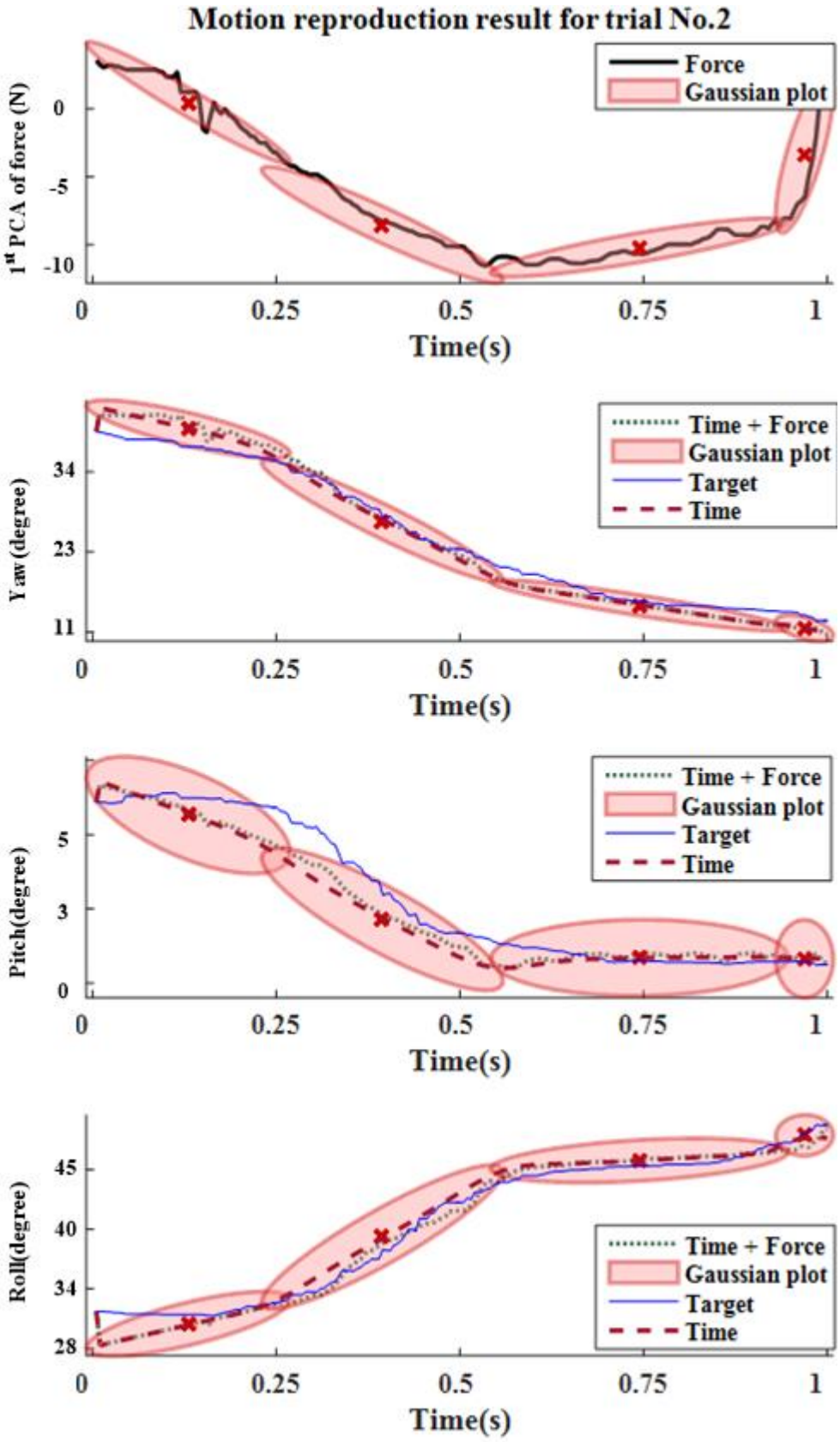


Figure 4-19 Motion reproduction result for trial 2.

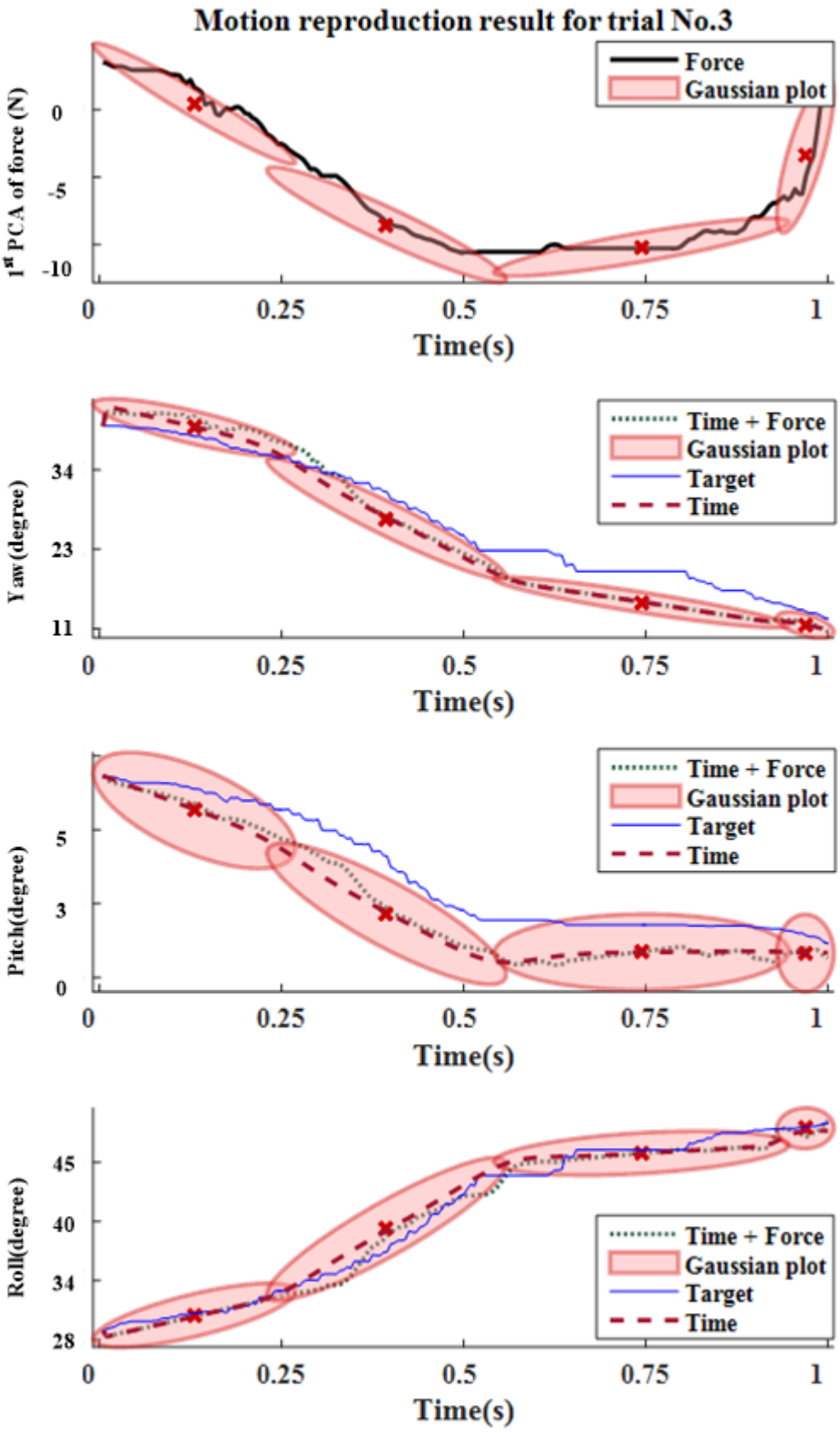


Figure 4-20 Motion reproduction result for trial 3.

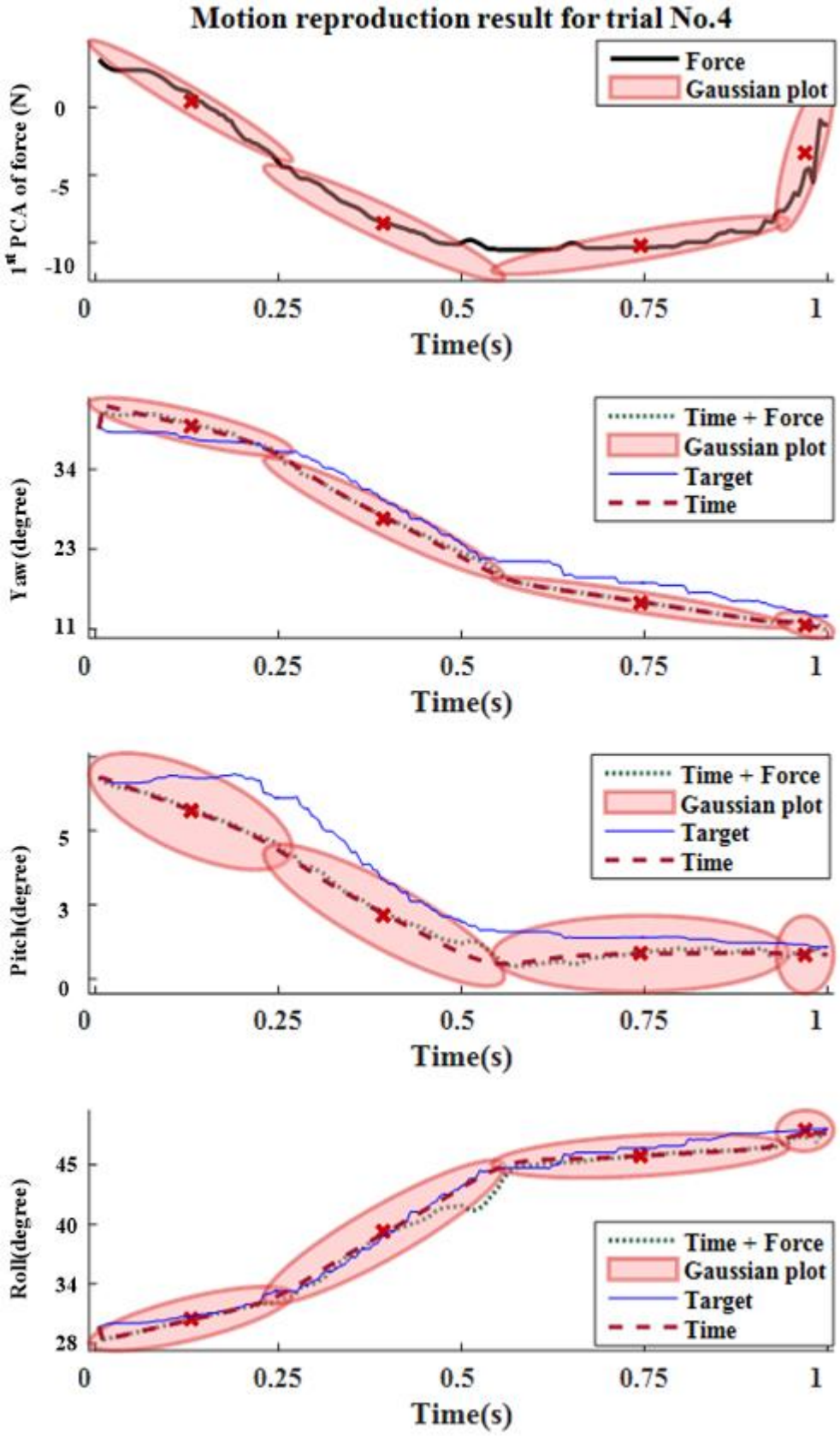


Figure 4-21. Motion reproduction result for trial 4.

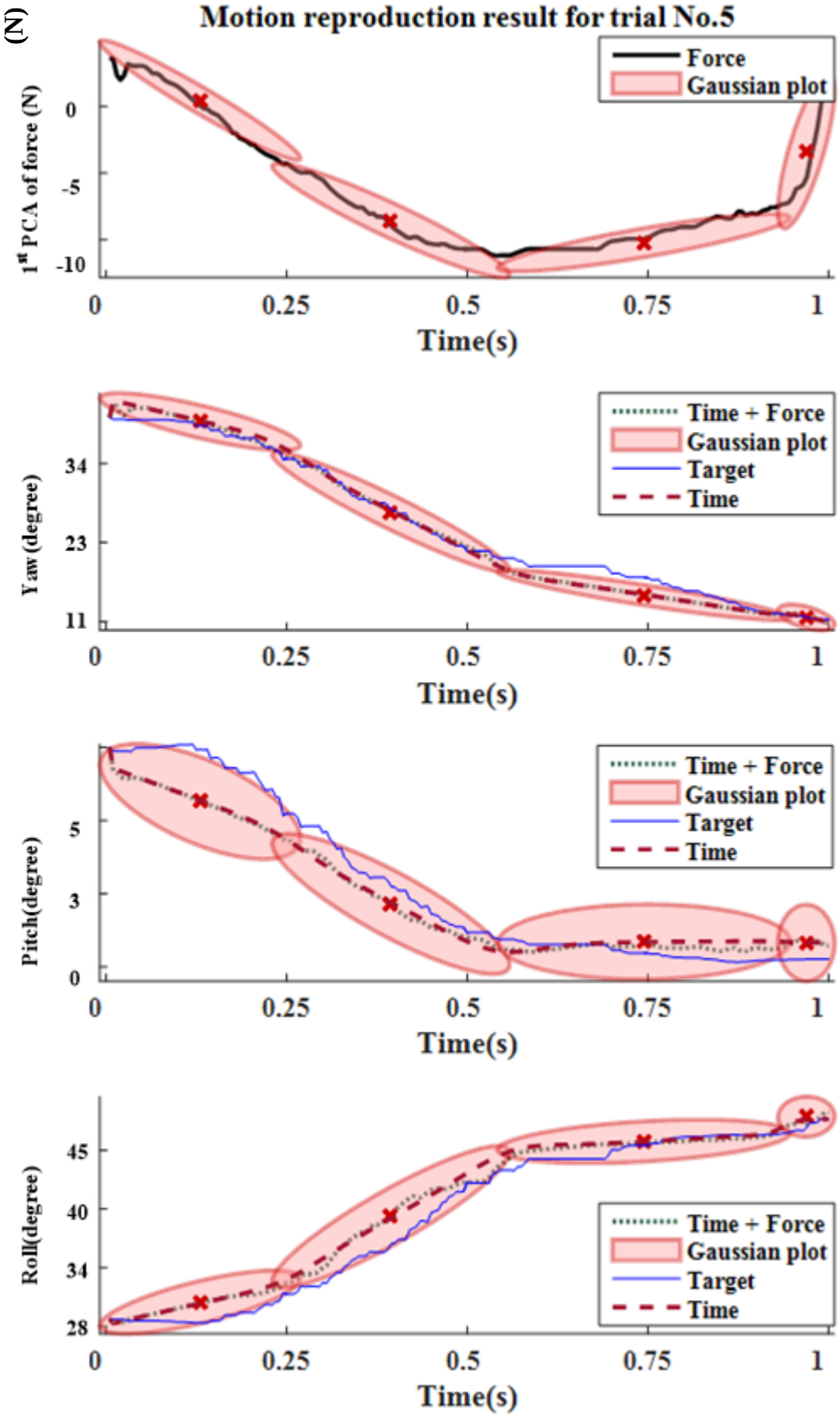


Figure 4-22. Motion reproduction result for trial 5.

b) *Motion reproduction by using extended GMR*

The discussion in the last section assumed the demonstrations were pre-aligned by the DTW, and the weights were computed based on $\{t, f_{pca}^{1}\}$ only. The advantage of this assumption is skills model can be learned regardless of the time variations, but it requires the trials are pre-aligned in time domain. To release this constraint, the time is implicitly encoded in this section. An extended version of GMR (GMRa) is proposed by modifying the weights computation, which are defined in equation (4-11).

$$\beta_i(\xi_t) = \frac{\alpha_{i,t}}{\sum_{k=1}^{K} \alpha_{k,t}} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots . \dots . . \text{(4-11)}$$

with $\alpha_{i,t} = \left(\sum_{k=1}^{K} \alpha_{k,t-1}\, a_{k,i}\right) N(\xi_t; \mu_i, \Sigma_i)$, $a_{k,i}$ is the transition probability being in the state *K*. $\alpha_{i,t}$ is the forward variable (defined recursively through the HMM representation) corresponding to the probability of partially observing the sequence $\{\xi_1, \xi_2, \dots, \xi_t\}$ of length *t* and of being in the state *i* at time *t*, where $\xi_t = \{f_{pca}^{1},\ q,\ \dot{q}\}_t$. This method had the advantage of encapsulating robustly the sequential nature of the data. Also, since angular velocity information was also available during the reproduction, from the current f, q and $\dot{q}$, a task-level proportional-derivative controller similar to a mass-damper system could be computed to reach the desired q and $\dot{q}$. The angular acceleration $\ddot{q}$ command in task space was determined by:

$$\ddot{q} = \overbrace{(\dot{q}_d - \dot{q})k_v}^{\ddot{q}^v} + \overbrace{(q_d - q)k_p}^{\ddot{q}^p} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots . \text{(4-12)}$$

where $\dot{q}_d$ is the desired angular velocity derived from the GMRa, and q_d is the desired angle derived from the GMRa. In general, the idea is to allow the robot to follow the model dynamics but not moving away from the learned model. The $\ddot{q}^v$ term follows the learned motion but tends to move away from the demonstrations after a few iterations or in some new unexplored situations. The $\ddot{q}^p$ term allows the robot to follow the learned model by commanding the robot to move to the closest point of the generalized trajectory. Therefore, the explicit time sequence is excluded and the joint distribution to be learned is $P\left(f_{pca}^{1},\ q,\ \dot{q}\right)$.

In this section, the results from subject B are used for discussion purpose. The results are showed the 1st PCA component of the force profiles from the 5 testing data trials in section 4.3.3a). The force profiles used in this section contains detail about the velocity and acceleration, which were not used in the previous sections. As shown in

the Figure 4-23, the force profiles contained time variations. The model generalization capability across different trials is shown in Figure 4-24. The results indicated that the training models gradually generalised across the new test datasets. Figure 4-25 - Figure 4-29 showed the reproduction results by benchmarking the *kv* and *kp* parameters of the spring-damper controller. Note the results in the figure cannot directly be compared with the figures in the last section because the time variable is not explicitly learned in the GMM. Therefore, the corresponding statistical results are summarised in Table 4-13 for comparison. From the results:

i. The reproduced trajectories followed the target trajectories as demonstrated by subject B. This result indicates that though the time variations exist, the GMRa model could sufficiently generate corresponding motion commands for the robot to reproduce.
ii. The reproduced motions tended to follow the learned model captured by the Gaussian distributions. As shown in trials 1 and 4, although the target motions deviated from the learned model, the reproductions mainly stayed in the Gaussians. This behaviour is due to the $\ddot{q}^v$ and $\ddot{q}^p$ terms.
iii. The small *MSE* and large *R* are shown in Table 4-13which indicated good model reproduction performance. In general, the pitch and roll results in this section are better than section 4.3.3.4.3.3 where explicit time variable was used. The yaw angles are slightly worse.
iv. The selections of the *kv* and *kp* parameters vary across trials. The values used here are constant and selected by exhaustive search. In reality, these control gains need to be adjusted online.

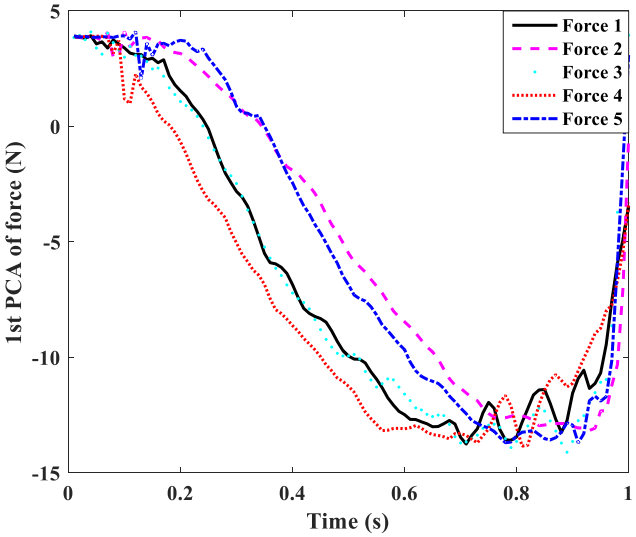


Figure 4-23 Five testing force profiles.

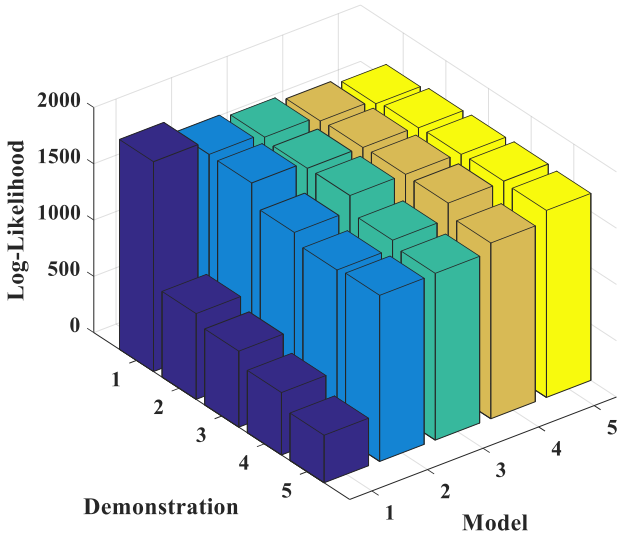


Figure 4-24 Model generalization across trials for subject B.

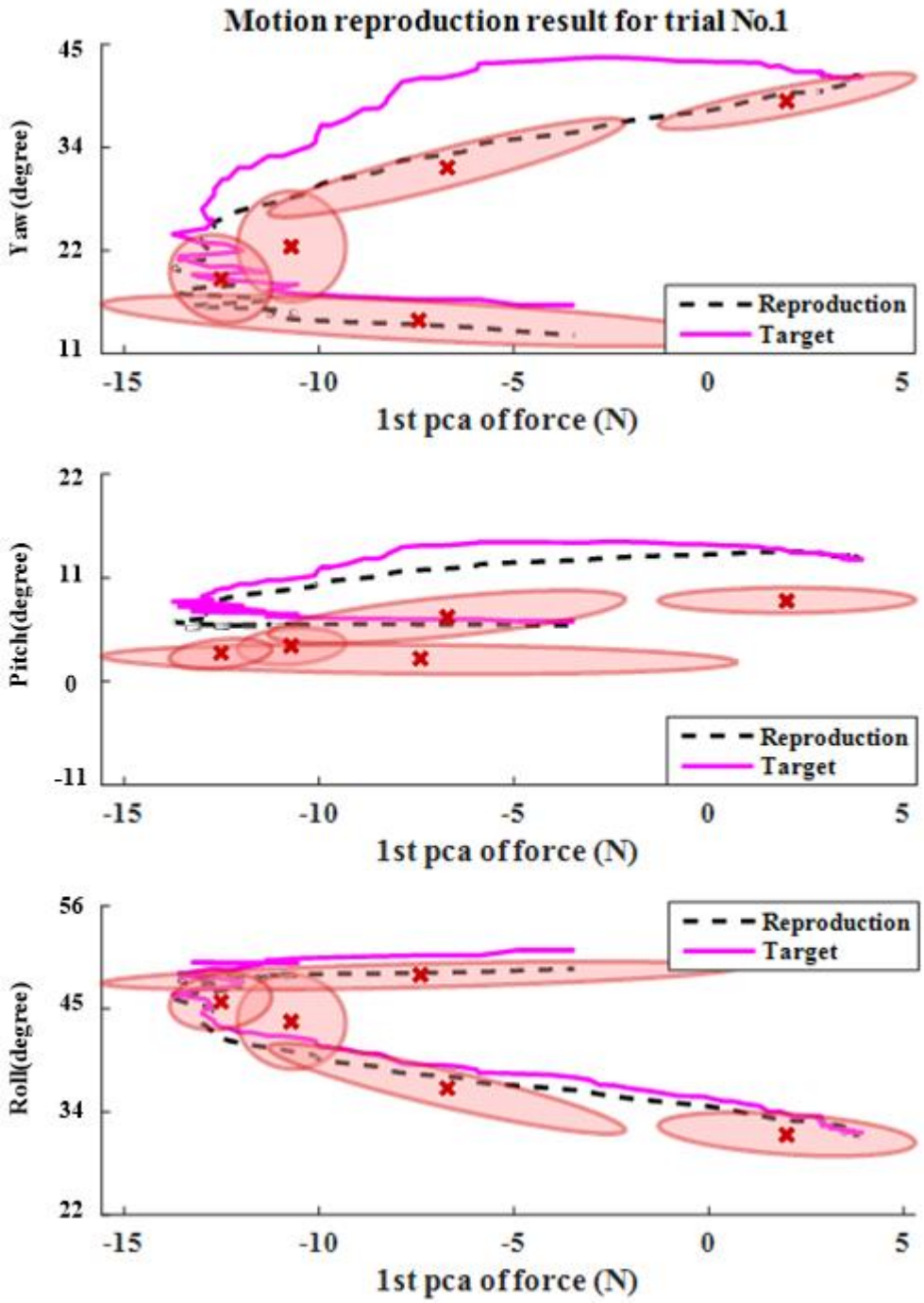


Figure 4-25 Motion reproduction for trial No.1

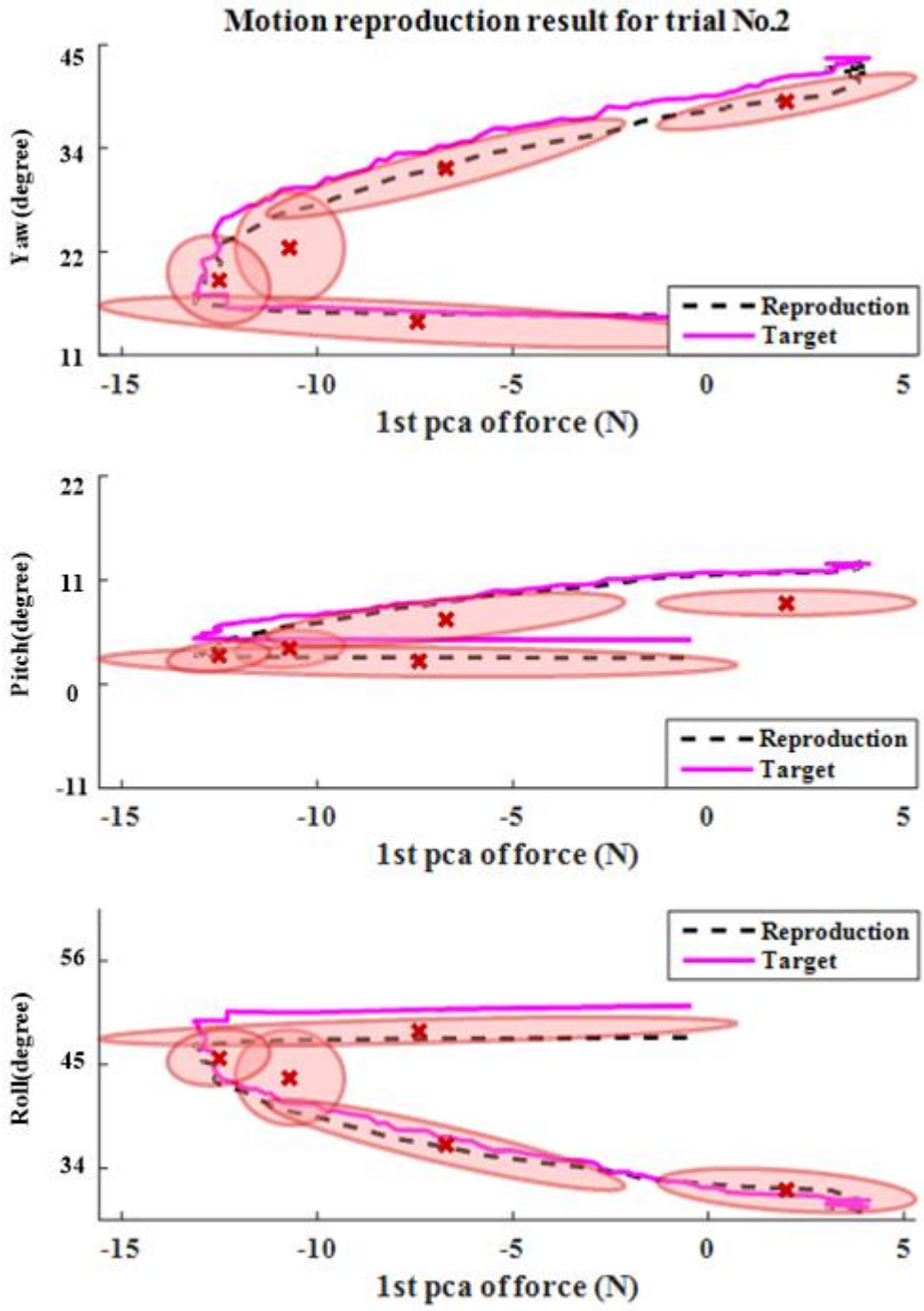


Figure 4-26 Motion reproduction for trial No.2

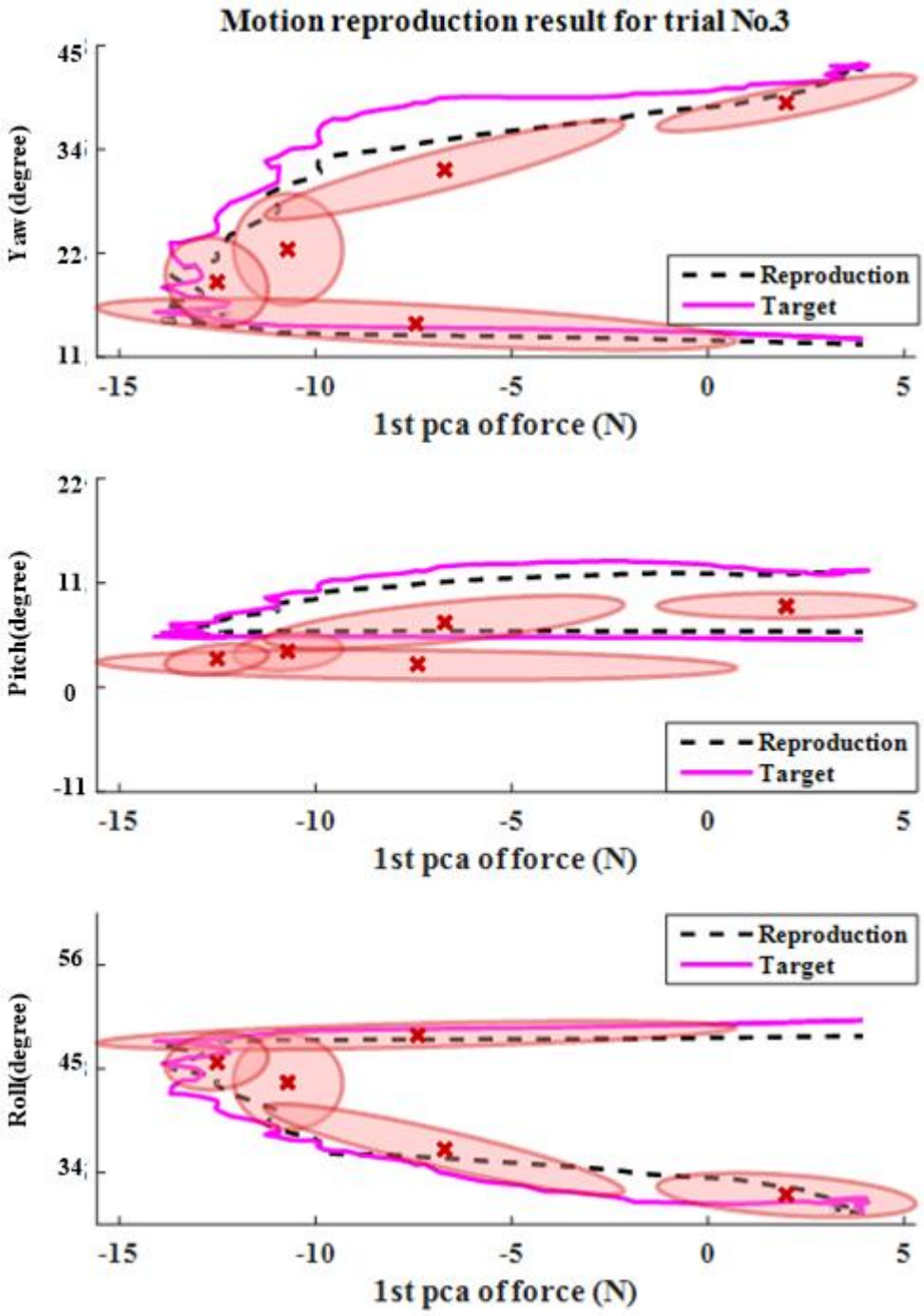


Figure 4-27 Motion reproduction for trial No.3

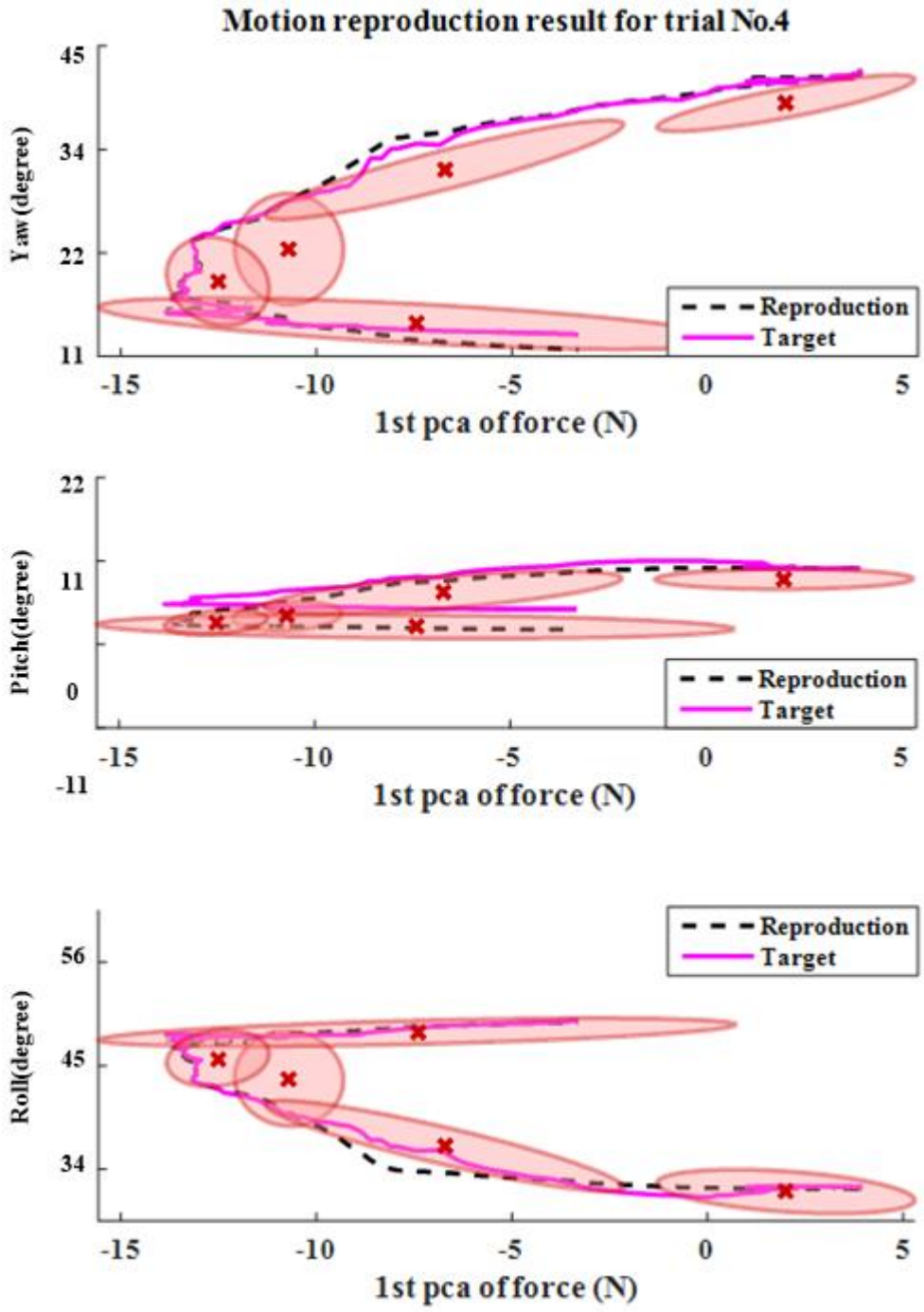


Figure 4-28 Motion reproduction for trial No.4

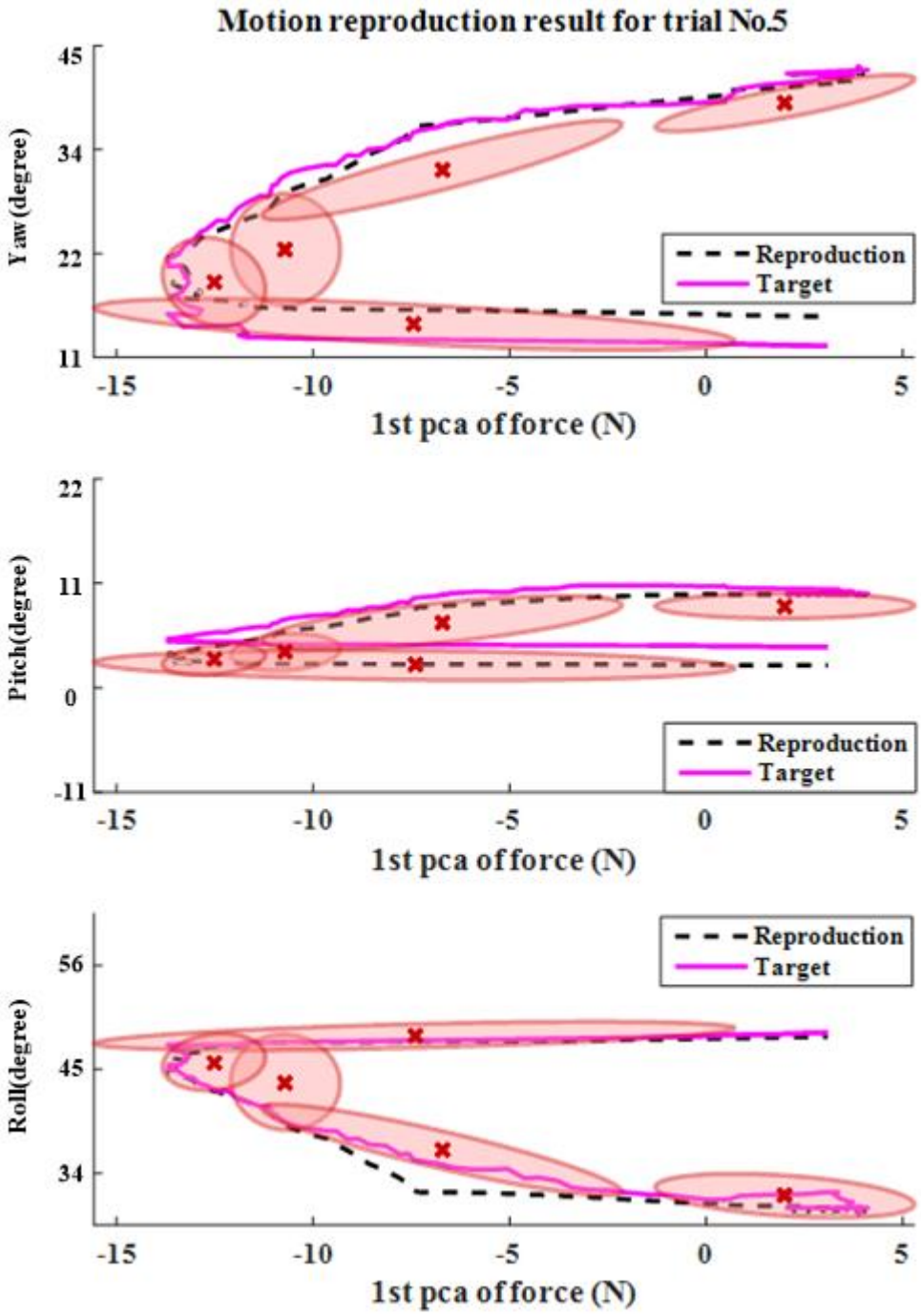


Figure 4-29 Motion reproduction for trial No.5

Table 4-13 Motion reproduction results for subject B. a) with implicit time encoding b) with explicit time encoding from. 5 trials were used. The influences of the input variables are compared by using *MSE* and *R* values.

a)								
Trial	**Kv**	**Kp**	**Yaw (degree)**		**Pitch (degree)**		**Roll (degree)**	
			MSE(degree)	**R**	**MSE(degree)**	**R**	**MSE(degree)**	**R**
1	**0.5**	**0.01**	**4.87**	**0.9687**	**1.55**	**0.9648**	**1.18**	**0.9971**
2	**0.5**	**0.01**	**2.01**	**0.9967**	**1.06**	**0.9971**	**1.26**	**0.9968**
3	**0.3**	**0.01**	**2.58**	**0.9904**	**0.82**	**0.9833**	**1.77**	**0.9802**
4	**1.0**	**0.03**	**0.60**	**0.9986**	**1.94**	**0.9913**	**0.86**	**0.9936**
5	**0.7**	**0.01**	**1.05**	**0.9942**	**1.32**	**0.9912**	**1.21**	**0.9934**
All	**0.6**	**0.01**	**2.28±1.60**	**0.9897±0.012**	**1.32±0.4**	**0.9855±0.012**	**1.26±0.28**	**0.9922±0.007**

b)							
Trial	**Input Variable**	**Yaw (degree)**		**Pitch (degree)**		**Roll (degree)**	
		MSE(degree)	**R**	**MSE(degree)**	**R**	**MSE(degree)**	**R**
1	**[t,f]**	**2.12**	**0.994**	**10.26**	**0.987**	**1.16**	**0.992**
2	**[t,f]**	**0.40**	**0.9976**	**8.37**	**0.9943**	**2.19**	**0.9927**
3	**[t,f]**	**0.57**	**0.9963**	**1.15**	**0.9911**	**2.64**	**0.9969**
4	**[t,f]**	**1.49**	**0.9881**	**50.77**	**0.9259**	**1.71**	**0.9848**
5	**[t,f]**	**0.87**	**0.9917**	**47.12**	**0.969**	**1.05**	**0.9953**
All	**[t,f]**	**1.10±0.57**	**0.9935±0.003**	**23.55±23.44**	**0.9735±0.028**	**1.76±0.47**	**0.9923±0.004**

4.4 Conclusion

In this chapter, the research objective three (*to build state-action policy models from human demonstrations that relate to industrial manual manipulations*) has been addressed. The human skills for PiH process were captured by using the IMU-Vicon system (chapter 3) and F/T sensor. A novel probabilistic encoding method was proposed to encode the skills from the PiH task. The data were first encoded by using the proposed GMM-HMM based method into probabilistic models. The recognized states were then interpreted statistically and descriptively. The results showed that different subject adopted a different approach to complete the task. This difference was reflected on the timeline and variations in each state. The proposed method showed its capability to encapsulate all these uncertainties into a probabilistic models and a generalized trajectory can be reproduced based on GMR. Furthermore, the method provided the ability to generalize the model across different subjects by the level two encoding which is another encoding layer on the individual models.

After the skills had been encoded and generalised, the motion reproduction using the models were evaluated using further test from the same subjects. To simplify the validation process, robot was not used because the inverse kinematic solver and the

transformations from the human hand to the robot end effector were not available. The performance of the model to reproduce motions on unseen testing scenarios was evaluated from the same subject by assuming the robot generates these additional trials. Two schemes were compared for reproductions. In section 4.3.3a), GMR with explicit time was used. In section 4.3.3b), a modified GMRa using forward variable from HMM was applied. The results showed that the reproduced motion trajectories were within the learned Gaussian models and the estimations were close to the target trajectories with some exceptions when the uncertainties were not previously demonstrated. Also, the reproduced trajectories generated from the GMRa have smaller variations.

In this chapter, the influence of the force signal has been discussed, and the model results from motion trajectories have been validated. However, for practical applications such as polishing or bolt assembly, it is unlikely to have a place to install a F/T sensor as shown in Figure 4-1. In addition, holding an F/T sensor to accomplish a task might skew the underlying skills, therefore a more flexible method to collect the haptic feedbacks is important to generalise the force based skills encoding method in this work. To achieve this, in the next chapter, a sEMG-driven model is developed to estimate the contact forces in the PiH task using the wearable sEMG sensors.

Chapter 5. Muscle-Force Modeling using Wearable sEMG sensors - A PiH Case Study

5.1 Introduction

The skills in the peg insertion task has been learned and encoded in chapter 4. In manufacturing, the Force and Torque (F/T) sensor attached to the robot end effector generates force feedback to the robots. However, it would be impractical to mount an F/T sensor on the human operator whilst they are carrying out skilled tasks.

Despite the mentioned difficulties, surface Electromyography (sEMG) can provide valuable information about the muscle activation, which can be closely related to the generated force [91]. The advantages of using wearable sEMG sensors are that it has minor interference of the operator and the signal is closely related to the actual contact forces. However, the sEMG-force model needs to deal with variations from the noisy signals, joint positions and different subjects etc. Therefore, this chapter will investigate how well a sEMG signal collected from a human operator's forearm can be mapped to the force response at the point of assembly. If the force information can be reliably predicted from the level of the muscle activations, this would have significant benefits as the human operator can wear the sensors easily and perform assembly tasks without the need for process specific instrumentation or restrictions caused by such instrumentation.

The main contribution of this chapter is the method to build the muscle-force model with the aim to address research objective three, which is to develop a muscle-force model to predict the forces generated from the forearm muscle activations using wearable sEMG devices. A Peg-in-Hole (PiH) assembly task was used as in chapter 4. It contained relatively simple motions so that the non-linear mapping from sEMG to force can be reliably modelled within the context. To avoid direct F/T sensor installation during the actual demonstration, a generic test rig was designed to decompose the hand poses in the PiH task prior to model training. Then, the data from the actual PiH were used to evaluate the model generalisation performance. The proposed method of mapping sEMG to force signal is described in section 5.2. The PiH experiment setup and protocol are explained in section 5.2.1, and the results are analysed in section 5.3.

5.2 Methodology

In order to measure the contact forces using sEMG sensors for Learning from Demonstration, a methodology is proposed in this section. To simplify the model, motion primitives were defined and a test rig has been designed. The model built from simple motion is assumed close to the actual PiH task and verified. Since the predominant force in the PiH is in vertical axis, the three primitive motions (A, B and C) capture the pressing down motion by activating different group of muscles. In section 5.2.1, the operators were firstly asked to perform the defined primitive motions then demonstrate the actual PiH in the same workspace defined in the experiment setup. The sEMG-force model is implicitly learned by time delayed neural network (TDNN) as discussed in section 5.2.2. The related uncertainties in this chapter are listed as follow: i) different data filtering strategy; ii) data dimension reduction; iii) different muscle groups (two sEMG armbands are used for measuring upper and lower arm muscle group activations); iv) changing of environment conditions; v) different calibration primitive pose; vi) different assembly clearance (loose, middle and tight). The variations in building primitive motion models are addressed in section 5.3.1. The actual PiH samples were then used to evaluate the model built from primitive motions and to select the most accurate primitive motion in section 5.3.2.

5.2.1 Experiment setup

In this section, the experiment setup of this chapter is explained. The input data were collected from the two sEMG armbands (one for upper arm muscle group and the other for lower arm muscle group) with 8 channel sensors each to record the muscle activations and the Vicon-IMU system to track the hand motions. The target data were collected from the F/T sensor. The approximate synchronisation algorithm in ROS platform synchronised the sEMG signal and the F/T signal. The sampling rate was 200 Hz. Since the hand motion was relatively slow and almost static in this case study, it was not necessary to up sample and synchronise with the F/T signal. It was simply updated in its thread, and the most recent value was used in the synchronizer thread.

Four subjects (from age 25 ~ 27) were invited to the experiment. All of them were male researchers from Intelligent Automation lab and trained to be familiar with the experimental protocols. To begin with, the wearable Vicon-IMU system needed to start from approximately the same pose to minimise the uncertainty of the initial pose.

This could be any natural pose. The sEMG sensors needed to warm up to allow the electrode to attach firmly onto the skin. The operator was asked to wear them and wait for 2 minutes before starting the experiments. The F/T sensor was rigidly installed on the MOTOMAN end effector. The test rig and the hole frame were also rigidly attached to the F/T sensor. The position of the setup, as shown in Figure 5-1, allowed the operator to accomplish the task in a comfortable way. The experiments procedure is detailed as following:

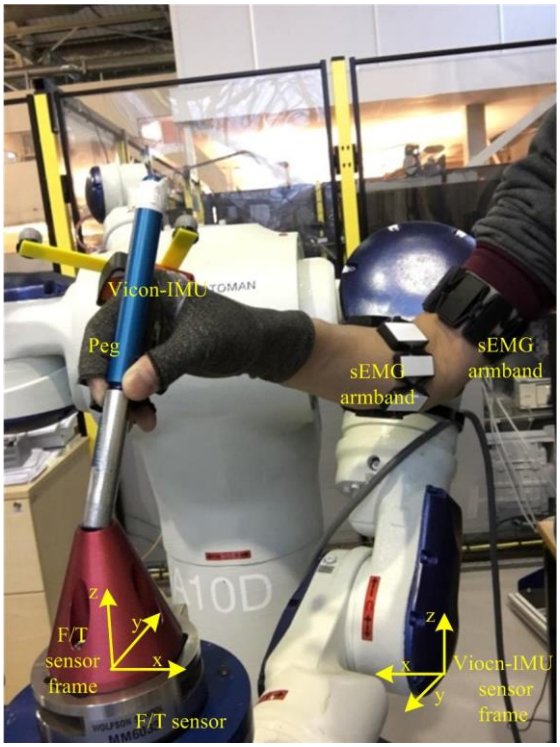


Figure 5-1 Experiment setup for PiH. The operator is wearing two sEMG armbands to record the muscle activations. A Vicon-IMU system is installed on the hand for tracking hand motions. An F/T sensor is statically installed to record the reference contact force signal.

1. In order to calibrate the muscle activations against the F/T sensors. The operators were asked to wear the sEMG sensors and stood in front of the test rig in a fixed position as shown in Figure 5-2A). After the warming-up stage, the operator started pressing on using pose A with their thumb and index fingers in the contraction and relaxing pattern for 25 seconds. The completion time of this pattern can be different. The force generation level should not exceed the maximum actual PiH execution (20N). The value can be selected by finding the peak values in the force signals on z axis. This experiment data were recorded in the morning for all four subjects. Two trials with at least ten contractions and relaxing were collected for training and a further trial was used for testing.

A) B) C)

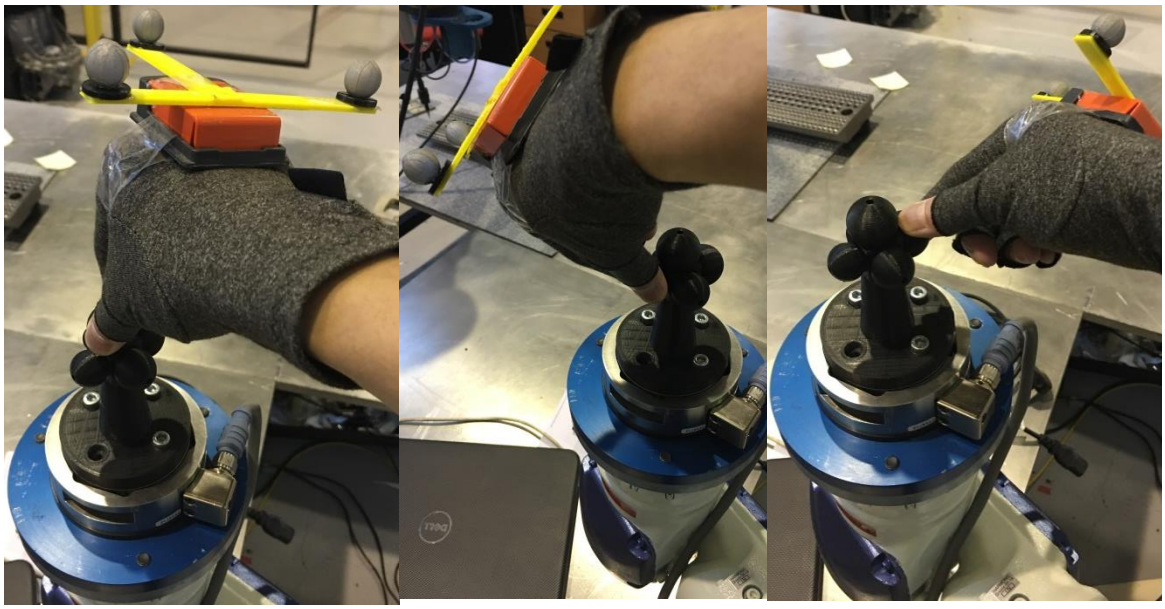


Figure 5-2 A test rig for muscle-force calibration.

2. The operators were asked to repeat the experiment one on pose B and C. This experiment aimed to evaluate the methodology on a different hand pose as shown in Figure 5-2(B, C). In this case, pose B was pressing on the left node and pose C was pressing on the right node.
3. In order to evaluate the generalisation of the model by changing the time of operation, the operators were asked to wear the same setup and repeated experiment 2 times in the afternoon (6 hours later). The model built from experiment one will be used to evaluate against the data from experiment three.
4. The operators were asked to perform the actual PiH skills by starting from the hole chamfer. Full insertion was not necessary because most of the haptic feedbacks were related to the corrective motions to align peg and hole. Two completions of the PiH contained insertion and extraction phase is shown in Figure 5-3. Each completion time can be different. Each trial included at least ten PiH completions and lasted for about 25 seconds. Two trials were recorded for testing the model built from the test rig (from steps 1-2).

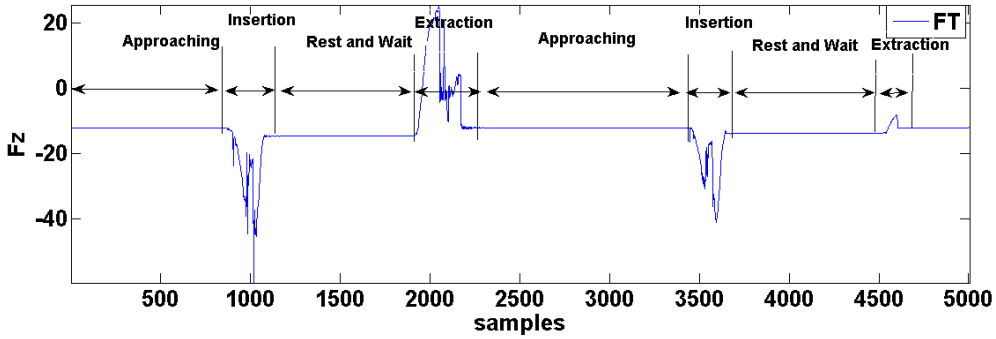


Figure 5-3 Illustration of the insertion and extraction phase in PiH.

5. In order to evaluate the generalisation capability of the approach on different peg clearance and fit. The operators were asked to repeat experiment 4 for additional two different pegs. The nominal diameter of the hole was 16.2 mm. In total, three pegs that form loose, middle and tight tolerances with the hole were used, as shown in Figure 5-4. The tighter tolerance means more resilient force on the feedback which potentially produces more strong correlation of the sEMG and force signals.

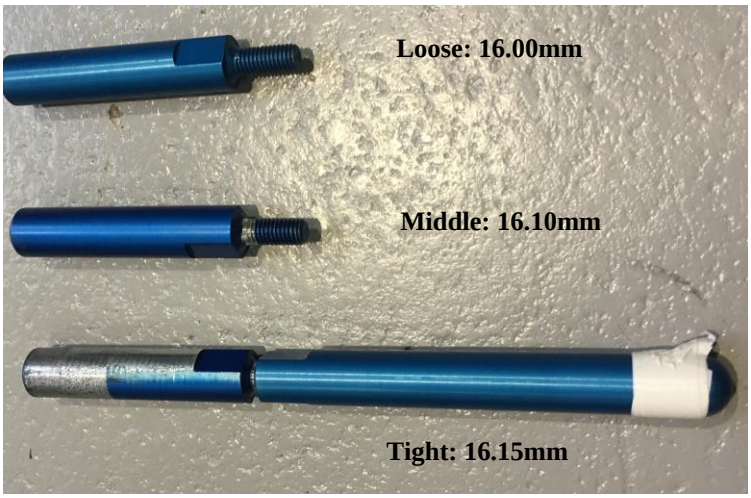


Figure 5-4 Pegs of different dimensions. The loose, middle and tight clearances are 0.2mm, 0.1mm and 0.05mm respectively.

5.2.2 Modelling method

The method of mapping from sEMG to force signals is explained in this section. In this research the insertion phase was emphasised as it provided the most valuable information for the PiH task. The variability introduced in this task was mainly from the noisy sEMG signals, electrode placements, different subjects and the training parameters.

To address these varitions, a method is proposed as shown in Figure 5-5. Firstly, a rectification strategy that converts the negative sEMG signals to positive was applied. Secondly, data filtering strategies for both sEMG and F/T sensor data were developed. Then, the signals were re-represented in the time domain; data was split based on the states of interest, i.e. the insertion, and extraction phases of the assembly. Thirdly, normalisation was applied to make the data-set scale invariant followed by a Principal Component Analysis (PCA) which was used to further reduce the dimensionality of the data set. After the previous pre-processes, a Time Delay Neural Network (TDNN) was implemented for regression (mapping) purpose. The training inputs and outputs were the raw sEMG (*emg*) and F/T ($f^{F/T}$) data, which have been pre-processed before the training session. The stages of this methodology are elaborated in the following sub-sections.

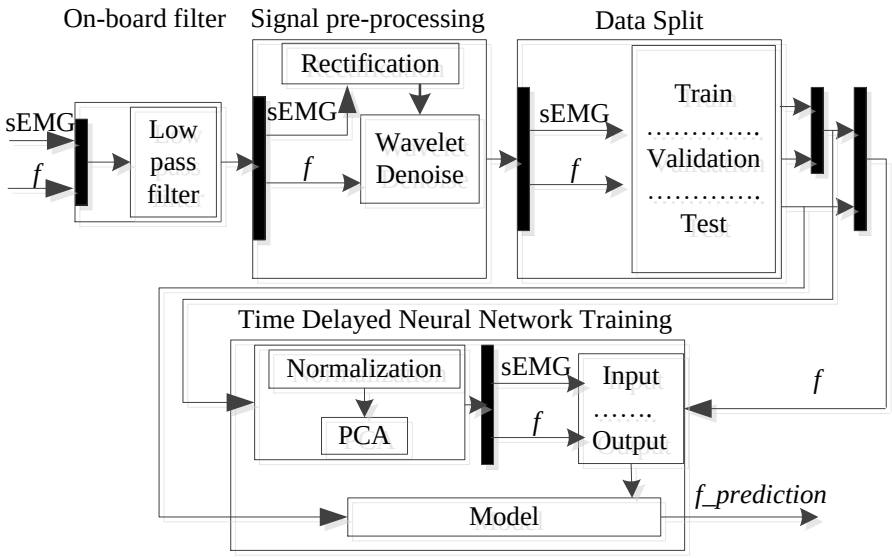


Figure 5-5 An overview of the proposed sEMG-force modelling method.

a) Data rectification and filtering

The data from the F/T sensor, as shown in Figure 5-6, can be labeled into states (e.g. PiH phase). However, from the sEMG data, it was not a trivial task to identify when the insertion phase happened due to noise. Therefore, the sEMG and force signals were sampled at the same sampling rate (200 Hz) so that every state could be labelled by referring to the force signal.

As shown in Figure 5-6, the raw signals from both of the sensors contained noise, but the sEMG signals were noisier (bottom). This was due to the stochastic behaviour of the sEMG signals. They contained information from the actual muscle

activations, the power line interference and the influence from the different group of muscles etc. The Discrete Wavelet Transformation (DWT) was implemented as a filtering strategy for the F/T signal in chapter 4. In this chapter, DWT was also used to de-noise the sEMG signals from various sources. 'Daubechies 10' (*db10*) basis wavelet and the 6th level approximation of the original signals were chosen heuristically as the data filtering strategies. The results are shown in Figure 5-6; the sEMG envelope data (solid line) is positive. This is because the negative component of the signal was changed to positive in the rectification process. The mean, integration and wavelet transform can then be calculated based on the rectified signal.

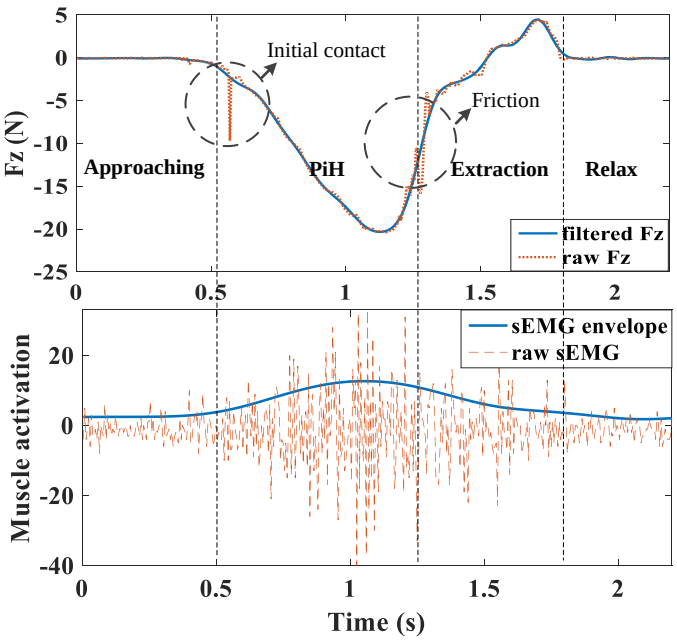


Figure 5-6 Filtered Fz data (top) and sEMG data with its envelope from one of the electrodes (bottom).

b) Data normalisation and PCA

Normalization is a critical pre-processing stage for the sEMG signals for comparison between different muscles or individuals. Instead of comparing the absolute muscle activities levels, a percentage of changes to a reference signal is used. There are a few approaches to normalise the sEMG signals. In [220], the most popular method was illustrated. In this chapter, the peak or mean activation normalisation was used and implemented by the standard zero mean normalization algorithm. The reason behind this was that the muscle activities levels are not directly comparable, but the activation patterns and their corresponding force torque datum are of interest. The equation is

shown in equation (5-1), where z is the original data after normalisation, μ is the mean, and σ is the standard deviation of the signal.

$$z = \frac{(x-\mu)}{\sigma} \quad \text{(5-1)}$$

Then, a PCA is implemented [221][222][5]. In this chapter, PCA is used to un-correlate the variables of a given signal and reorder them regarding their importance, i.e.to reduce the data dimensionality. In PCA, only the co-variance between the variables (8 channels of sEMG data due to the MYO armbands used, see section 5.3.1.b) were considered and re-ordered from the most important component to the least important component. Consequently, the data dimensionality was reduced and the new dataset was re-ordered based on their importance.

c) Time delayed neural network (TDNN)

From the literature, explicit models can be built by investigating the non-linear relationship of Musculotendon Dynamics [22]. However, there is lack of research of using a data-driven approach to predict the force response during an assembly process from sEMG signals. The data-driven method is considered in this chapter due to its simplicity and generalization capability across applications. The techniques to find the mapping from the sEMG signals to the force signals are complex for unconstrained motions because the model parameters vary when the joint positions change. It has been reported in [22] that the accuracy of mapping or predicted force signals from sEMG varies across different tasks with different degrees of freedom. This means a task-specific model should be generated for different applications. Therefore, by constraining the motion into a fixed workspace, it should be easier to obtain a reliable model.

Neural Network (NN) is a widely used function approximator with the aim of minimising the error of the network compared to a validation set. Each artificial neuron, as a computational unit, will calculate the weighted sum $\sum_j w_j x_j$, where x is individual input, w is weight. If the summation is greater than a threshold value, it will make a decision such as true or false. Therefore, an NN is essentially a decision making tool that weighs up the evidence. The detail explanation of the NN is not of the main topic here, see [10-11] for details. A typical structure of NN may vary according to the applications. The conventional feed-forward network with back propagation optimization is widely used [147]. However, in order to model the dynamics of time

series data, network structures allow feedback (time delay or recurrent) with multiple layers are considered more advanced (accurate) but more complex to train.

To achieve this, in this chapter, we used a TDNN with two hidden layers as shown in Figure 5-7. *i* stands for the *i* th element of the sEMG signal. *t*, *t-1*, and *t-2* annotate the time step of the data. The activation functions used for hidden layers are a sigmoid and linear line for the output layer. The architecture contained a hidden layer with a non-linear sigmoid kernel and a layer with a simple linear regression based on previous hidden layer's outputs. Training a neural network is not a trivial task. If not properly trained, there is a risk of fitting a model without any real meaning. Over-fitting is a known problem when the model becomes too complex and loses its generalization capabilities. However, some strategies can be used to overcome this issue such as cross-validation, Bayesian regularization [56] and early stopping. The selected training strategies are shown in section 5.3.1a).

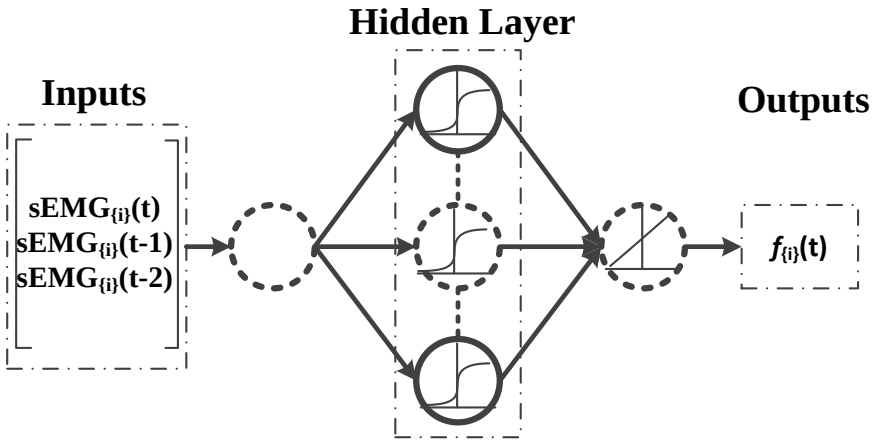


Figure 5-7 TDNN architecture.

5.3 Result analysis

The sEMG-force model was firstly built for pose A, and then it was built on pose B and C. Then the best model for the individual subject was applied on the actual PiH. The input variables are sEMG, 1st PCA of the force signal and the hand motions $\{emg, f_{pca}^{1}, and\ q\}$. The signals from the F/T sensor were used to evaluate against the estimated signals. The 1st principle component of the tri-axial force signals from PCA contained 98% of the force information and was used as the reference to simplify the modeling process.

For model validations on the various calibration poses, the selection of the filtering and TDNN training strategies are discussed in section 5.3.1. The influence of PCA is discussed in section 5.3.1.b, followed by the model evaluations on the different muscle groups and environment conditions in section 5.3.1c) and section 5.3.1d) respectively. Finally, model validation for the actual PiH executions is discussed in section 5.3.2 where the effects of the calibration pose and peg clearances are discussed.

5.3.1 Model selection for calibration pose

The pressing down movement on the muscle-force calibration test rig was considered as the primary element in the PiH task. Therefore, the model was trained for this movement by mapping the sEMG signals to the contact force (pose A). The two other poses (pressing down in pose B and C) on the same test rig were used to verify the model. The effects of the filtering strategy, network training parameters, PCA, muscle groups and experiment time on the model are discussed.

a) Selection of the filtering and TDNN training parameters

The sEMG signals from the armbands were noisy, but they contain useful information to build the muscle-force model. Therefore, a filtering strategy was important to effectively use the sEMG signals. The model performance could be degraded by either remaining or removing too many details in the signals. On the other hand, the training parameters including the number of time delays (N_d) and hidden units (N_h) also had an effect on the model predictions. Therefore, a benchmark was needed to address these parameters.

As shown in Figure 5-8, the benchmark results from subject A were used as an example. Three trials from the pressing down experiment on node A in the morning were recorded. Two of them were used for training, and one of them was for testing. The *x*-axis is N_h, the *y*-axis is N_d, and the *z*-axis is the *MSE* performance. D_l is the wavelet decomposition level. From the results:

- The regression coefficients indicate a good model prediction performance (0.970, 0.974 and 0.968 in *a*, *b* and *c*). But the *MSE* has larger variations (10.70 N, 9.63 N and 15.74 N in *a*, *e* and *i*).
- The best result for this particular data set was using '*db5*' as wavelet basis function with $D_l = 5$. $N_h = 2$ and $N_d = 10$. The *MSE* = 9.64 N and the R = 0.974.

- Most of the plots indicated a clear trend that the increasing the hidden units do not lead to a better model. For instance, in *a*) and *b*), the number of hidden units increased from 1 to 3, but the *MSE* increased. This indicated that a simple Neural Network structure was a suitable model.
- Most of the plots indicated a clear trend that the increasing the number of time delays improved the model performance. However, starting from $N_d > 5$, the improvement tended to be smaller. This result indicated that the standard Neural Network without time delays is not suitable. But there is an appropriate sliding window size and $N_d = 10$ seemed to be a common choice.

A summary of the parameter benchmark for all the data collected from experiment one and two are shown in Table 5-1-

Table 5-3. The results indicated that it was necessary to address these parameters first since the model performances depended on the dataset. However, the results in Figure 5-8 showed that the performances were close to each other. Therefore the optimal parameter set might not be the most important considerations. From the summary results, a simple network structure with N_h equals to 1 or 2 with N_d equals to 10 or 15 was found to be optimal. This result confirmed the conclusion based on the performance plots shown in Figure 5-8. More importantly, the simple network structure indicated that the predictions were superposition of the linear functions with time delays by multiplying their corresponding weights. The selected model parameters were used in the subsequent discussions.

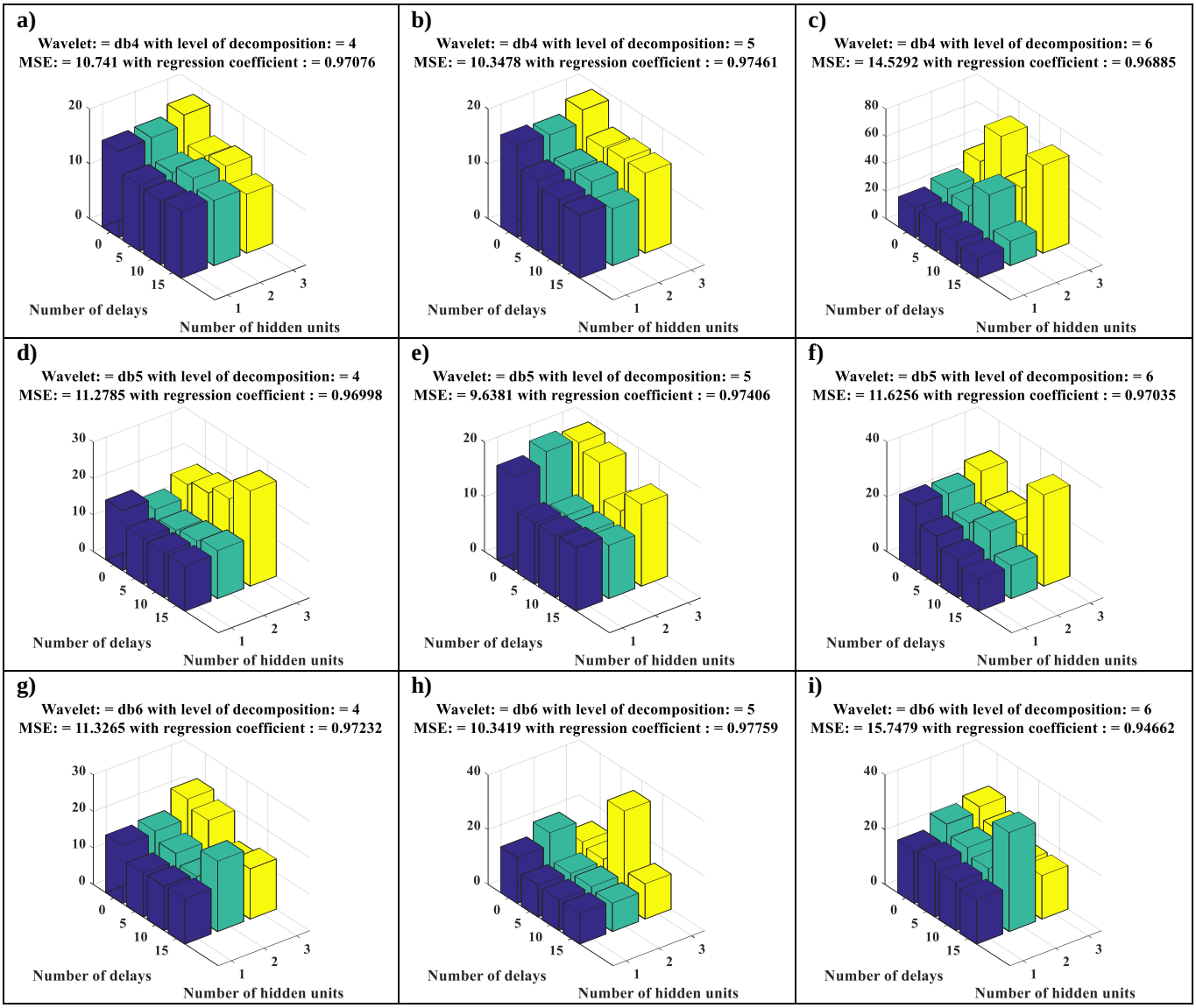


Figure 5-8 Benchmark of the parameter selections for the data from subject A.

Table 5-1 The benchmark results for pose A.

	Wavelet	Decomposition level	Number of time delay	Number of hidden units
Subject A	**db5**	**5**	**15**	**2**
Subject B	**db5**	**5**	**15**	**1**
Subject C	**db6**	**4**	**10**	**1**
Subject D	**db6**	**5**	**10**	**2**

Table 5-2 The benchmark results for pose B.

	Wavelet	Decomposition level	Number of time delay	Number of hidden units
Subject A	**db5**	**4**	**15**	**2**
Subject B	**db6**	**5**	**10**	**2**
Subject C	**db5**	**5**	**10**	**2**
Subject D	**db6**	**4**	**15**	**1**

Table 5-3 The benchmark results for pose C.

	Wavelet	Decomposition level	Number of time delay	Number of hidden units
Subject A	**db6**	**5**	**10**	**1**
Subject B	**db5**	**5**	**15**	**2**
Subject C	**db6**	**4**	**15**	**2**
Subject D	**db5**	**5**	**10**	**1**

b) Influence of PCA

The purpose of using PCA in this chapter was to reduce the dimension of the dataset, especially the F/T data and the 16 channels of the sEMG signals from the two armbands. As explained in the methodology, the dominant force component for the PiH task was the pressing down force as the z-axis force dominated the signal. Using PCA, the dominance of the z-axis force together with the information from the other axes of forces were considered. The dimension of the sEMG signal was 16, but not all the muscle activations were necessary for the PiH task. Therefore, potentially the PCA can help to reduce the dimension of the input variable while training the model. The results discussed in this section are the dataset truncated with and without 2% of PCA reduction from experiments one and two.

The following results are summarised in Table 5-4:

- The results from model predictions for subject A improved when using PCA for pose A and C. The *MSE* with PCA and without PCA in pose A were 9.49 N and 12.40 N respectively. The *MSE* with PCA and without PCA in pose C were 3.58 N and 11.16 N respectively. This indicates that for pose A and C, the PCA reduced some of the unnecessary details in the input variables so that the overall performance was better than without PCA. The results in pose B indicated that the *MSE* (12.87 N) after PCA was slightly worse than without PCA (11.36 N). However, the R was larger which meant the predictions were closer to the target signal.
- The results from subject B did not have the same performance as subject A. All the *MSE* and *R* showed that the model built with PCA was not as good as the model built without PCA. For instance, the *MSE* and *R* were 6.72 N and 0.972 respectively with PCA, and the *MSE* and *R* were 3.42 N and 0.982 respectively without PCA.

These results indicated that the PCA was not the necessary step for the dataset from subject B.

- The results from subject C had a better performance in pose B after PCA. The *MSE* was smaller, but the *R* value was almost the same. The results for pose A and C showed the opposite in the *MSE* performance.
- The *MSE* in the model performance built from subject D after PCA were all smaller than the model built without PCA. But the *R* was almost the same.
- In general, the low *MSE* and high *R* indicated that the prediction accuracy was acceptable and reasonably good across all the subjects. The best *MSE* and *R* was 2.19 N and 0.985 respectively for subject C in pose C.

In general, the model showed the capability to predict the force signal accurately for different postures and subjects. The PCA can improve the results for most of the dataset, but the improvements were not significant. However, even though not shown in the table, the training time with PCA was shorter than without PCA. Therefore, PCA was an optional step in the training process if the training efficiency is important.

Table 5-4 A summary of the result with and without PCA.

	Pose A				Pose B				Pose C			
	With PCA		Without PCA		With PCA		Without PCA		With PCA		Without PCA	
	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	9.49	0.975	12.40	0.974	12.87	0.973	11.36	0.958	3.58	0.982	11.16	0.979
Subject B	6.72	0.972	3.42	0.982	7.58	0.973	5.66	0.978	3.43	0.978	2.44	0.986
Subject C	6.88	0.978	5.86	0.972	4.62	0.98	7.88	0.982	3.34	0.978	2.19	0.985
Subject D	3.05	0.980	3.27	0.985	4.76	0.980	6.67	0.982	4.03	0.974	5.52	0.981

c) Influence of the muscle groups

The activations of two groups of muscles were recorded during the experiments: the biceps and triceps in the upper arm, and extensors and flexors in the lower arm. The focus of this section is on the contributions of the muscle groups on the contact force. This was because different subject might use slightly different muscles to execute the PiH task. Also, the placement of the sEMG sensor might vary when a different person reinstalled it on his/her arm. Thus the recorded muscle group activations might not be the same. In this section, the dataset from three poses was used to discuss the differences in activations of the muscle groups. The results from Table 5-5-Table 5-7 summarised the model performances for the different muscle groups. The whole

dataset ($emg, f_{pca}^{1}, and\ q$) contains all the input variables. Here five different models were analysed and discussed here: lower arm with motion data (L_M), upper arm with motion data (U_M), motion data only (M), the whole arm with motion data (W_M) and the whole arm only (W). The purpose of adding motion only data is to investigate whether a model can be built only based on the hand movements (no muscle activations information). The findings are discussed below:

Table 5-5 A summary result of the influence of the muscle group in pose A.

Pose A	Lower arm(motion)		Upper arm(motion)		Motion only		Whole arm(motion)		Whole arm	
	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	14.29	0.959	14.06	0.976	248.20	0.143	12.40	0.974	8.89	0.975
Subject B	5.74	0.971	3.37	0.983	10.5	0.965	3.42	0.982	5.89	0.973
Subject C	14.88	0.951	6.28	0.971	24.46	0.894	5.86	0.972	9.12	0.971
Subject D	5.93	0.968	3.77	0.981	14.17	0.929	3.27	0.985	3.16	0.983

- In pose A, the best model performance was from the whole arm model for subject D. The *MSE* was 3.16, and the *R* was 0.983. This result indicated that the model built from the muscle outperformed the model built from the muscle with hand motions. However, the improvement was not significant for subject D. The following are the detail discussions for individual muscle groups:
    - The L_M model for all the subjects was not as good as the model built from the W_M model. For instance, the *MSE* and *R* were 14.29 N and 0.959 respectively for subject A's L_M model. However the *MSE* and *R* were 12.40 N and 0.974 respectively for subjects A's W_M model. Within the L_M, subject B had the best performance with *MSE* and *R* equal to 5.74 N and 0.971 respectively.
    - The U_M models for subjects A, C, and D were not as good as the model built from the W_M. Subject B had a slightly better performance on his U_M model. The previous had *MSE* and *R* equal to 3.37 N and 0.983 respectively, and the latter had *MSE* and *R* equal to 3.42 N and 0.982.
    - The hand motion data cannot produce an acceptable model in all subjects. The best result was in subject B where the *MSE* was 10.5 N, and the *R* was 0.965.
    - The whole arm models had varying performances when they compared with the W_M model. Subject A and D had better performance on their whole arm only model. Subject B and C's models had better performance

with W_M model. Since the operators were asked to press down on calibration node A, the hand motion contained small rotation movements. The results indicated that the dataset from subjects A and D contained fewer hand rotations than the dataset from subject B and C. In another word, when subject A and D were performed the calibration tasks, their hands were almost static, so that the hand motions contributed a little on the model performance.

Table 5-6 A summary result of the influence of the muscle group in pose B.

Pose B	Lower arm(motion)		Upper arm(motion)		Motion only		Whole arm(motion)		Whole arm	
	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	21.6	0.930	18.36	0.962	106.94	0.696	11.36	0.958	26.20	0.940
Subject B	11.68	0.965	6.06	0.980	101.00	0.782	5.66	0.978	8.20	0.971
Subject C	14.64	0.960	12.88	0.978	133.46	0.542	7.88	0.982	4.76	0.979
Subject D	17.53	0.955	7.13	0.980	41.18	0.877	6.67	0.982	4.69	0.982

- For pose B, the best model performance was from the whole arm model for subject D. The *MSE* was 4.69 N, and the *R* was 0.982. This result indicated that the model built from the muscle outperformed the model built from the muscle with hand motions. The *MSE* was smaller, but the R-value remained the same. The following is the detailed discussions for individual muscle groups:
    - The L_M model for all the subjects was not as good as the model built from the W_M model. For instance, the *MSE* and *R* were 21.6 N and 0.930 respectively for subject A's L_M model. However the *MSE* and *R* were 11.36 N and 0.958 respectively for subjects A's W_M model. In the L_M models, subject B had the best performance with *MSE* and *R* equal to 11.68 N and 0.965 respectively.
    - The U_M models for all subjects were not as good as the model built from the W_M models. Subject B had a slightly better R on his U_M model. The previous has *MSE* and *R* equal to 6.06 N and 0.980 respectively, and the later had *MSE* and *R* equal to 5.66 N and 0.978.
    - The hand motion could not produce sufficient information to build a reasonable model. The best result was from subject D with *MSE* equal to 41.18 N and *R* equal to 0.877.
    - The model built from the whole arm only model outperformed the W_M models for subject C and D. This indicated that the features from muscle

activations were sufficient to build an accurate muscle-force model. The hand motion does not help to improve the model performance. However, the results from subject A and B were contradictory and suggested that the hand motions are useful for building the model in pose B

Table 5-7 A summary result of the influence of the muscle group in pose C.

Pose C	Lower arm(motion)		Upper arm(motion)		Motion only		Whole arm(motion)		Whole arm	
	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	15.71	0.964	14.44	0.972	84.3	0.02	11.16	0.979	3.18	0.983
Subject B	3.62	0.976	2.94	0.981	18.59	0.876	2.44	0.986	2.75	0.980
Subject C	3.50	0.975	3.84	0.984	92.88	0.164	2.19	0.985	2.77	0.982
Subject D	5.48	0.961	8.14	0.976	45.55	0.599	5.52	0.981	2.97	0.982

- In pose C, the best model performance was from the W_M model for subject C. The *MSE* was 2.19 N, and the *R* was 0.985. This result indicated that the model built from the muscle with hand motions outperformed the model built from the muscle only. The following were the detail discussions for individual muscle groups:
    - The L_M model for all the subjects was not as good as the model built from the W_M model. For instance, the *MSE* and *R* were 15.71 N and 0.964 respectively for subject A's L_M model. However the *MSE* and *R* were 11.16 N and 0.979 respectively for subjects A's W_M model. In the L_M models, subject C had the best performance with *MSE* and *R* equal to 3.50 N and 0.975 respectively. The *MSE* in subject D was almost the same with the W_M model, but worse than the whole arm only model. This indicated that the hand motion degraded the lower arm model performance.
    - The U_M models for all subjects were not as good as the model built from the W_M. Subject B had the best result on his upper arm model. The *MSE* and *R* equal to 2.94 N and 0.981 respectively.
    - The hand motion could not produce sufficient information to build a reasonable model in all case. The best result was from subject B with *MSE* equal to 18.59 N and *R* equal to 0.876.
    - The model built from the muscle activation only outperformed the model with hand motion for subject A and D. The improvement indicated that the sEMG signals were the predominant components for building muscle-force model. The hand motions were not necessary for these two datasets. On the other hand, subject B and C had similar performance with and

without using motion data. This indicated that the hand motions almost had no effect on the model training. The reason of this indication was that the operators used static hand pose, which contained little variations.

In conclusion, the training results from the different muscle groups indicated that all the subjects were using both groups of muscles to execute the pressing down movements for three different poses studied. This is because the model built from either lower arm and upper model cannot outperform the whole arm model. Even though reasonable model accuracy was achievable in both models, the whole arm model always provides improvement. The hand motions were negligible if the operator did not have sufficient movements in the training samples. This happens in the experimental dataset for pose C. However, in pose A and B, the hand motions contributed to model improvements due to the movements in the hand motion samples. Therefore, the muscle activations had strong correlation with the force generations, and if the calibration primitives (e.g. pose A, B, and C) contained hand movements such as angle tilting; the training model should consider the hand motions as additional important features.

d) Influence of environment conditions

The test data from the previous discussions were from two trials in the morning. In this section, to test the repeatability of the experimental setup, the experiments were performed in the afternoon. The purpose of this discussion is to show the generalisation capability of the training model by changing when the experiment was performed. The sEMG sensors were fixed in the same positions by marking on the arm. However, some misalignment might occur due to the reinstallations. The results are shown in Table 5-8-Table 5-10 and explained as following:

- In pose A, the model performances on the testing calibration trials in the PM were close to the model in the AM. For instance, the model performances for subject B in PM (trial2) had *MSE* and *R* equal to 6.55 N and 0.979 respectively. It was slightly worse than the results in the AM. On the other hand, the model performances for subject D in PM (trial2) had *MSE* and *R* equal to 2.96 N and 0.982 respectively. It was slightly better than the results in the AM.

Table 5-8 A summary of the results for different experiment time in pose A.

Pose A	AM		PM (Trial 1)		PM (Trial 2)	
	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	8.89	0.975	17.62	0.961	10.74	0.973
Subject B	5.89	0.973	4.76	0.985	6.55	0.979
Subject C	9.12	0.971	6.41	0.969	4.38	0.977
Subject D	3.16	0.983	4.18	0.981	2.96	0.982

- In pose B, the model performances on the testing calibration trials in the PM showed a good prediction accuracy on subject C and D. For instance, the *MSE* and *R* for subject C in PM (trial 1) were 4.62 N and 0.981 respectively, which were better than the evaluation results from AM. However, the model prediction performances from PM for A and B were worse than the performances from AM.

Table 5-9 A summary of the results for different experiment time in pose B.

Pose B	AM		PM (Trial 1)		PM (Trial 2)	
	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	11.36	0.958	8.37	0.965	14.34	0.959
Subject B	5.66	0.978	10.23	0.972	9.05	0.965
Subject C	7.88	0.982	4.62	0.981	4.27	0.982
Subject D	6.67	0.982	3.42	0.984	4.31	0.985

- In pose C, the model predictions for the PM were similar to the predictions from AM. For instance, the *MSE* and *R* for subject C in PM (trial 1) were 2.60 N and 0.981 respectively while the *MSE* and *R* for subject C in AM were 2.77 N and 0.982 respectively.

Table 5-10 A summary of the results for different experiment time in pose C.

Pose C	AM		PM (Trial 1)		PM (Trial 2)	
	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	3.18	0.983	3.46	0.976	3.05	0.978
Subject B	2.75	0.980	2.64	0.961	2.48	0.972
Subject C	2.77	0.982	2.60	0.981	2.86	0.974
Subject D	2.97	0.982	2.06	0.981	2.67	0.974

The results discussed here showed the model repeatability capabilities at different experiment times. The similar prediction accuracy in the PM compared with

the performance in the AM indicated that the methodology was repeatable. However, in some of the cases, the performances had degraded. This was because the sensors were retrofitted and might have misalignment with the original positions. Also even though the operator was asked to execute the calibration experiments by standing in the same position with the same hand pose, he/she might still introduce some misalignments in the postures. In general, the models built from different poses and subjects were transferable to a different execution time and were able to reproduce similar prediction performances.

In conclusion, this section has systematically evaluated the proposed model selection strategy for building the muscle-force model. Firstly, the filtering strategy and TDNN training parameters addressed the appropriate level of smoothing in the sEMG signals and the network structures. The results suggested a simple network structure is sufficient. Secondly, the PCA was used to reduce the dimensions of the input signals. The results indicated that the improvement of the model using PCA was not significant. But if the training efficiency was concerned, PCA could help to accelerate the training process for a noticeable time (at least 30s). Thirdly, the contributions of the different muscle groups were discussed. The results indicated that in order to build a comprehensive model for all the postures used in the experiments, both the upper arm and lower arm muscle groups should be used. The results also indicated that the motion data in the dataset were not as important as the muscle activations, but if the operator's hand motion contained noticeable variations, it would help to improve the model performance otherwise, it might degrade the model performance. Lastly, the experiments had been executed at different times of the day. The results indicated that the model was capable of generalising to these new datasets in PM with similar accuracy in AM. An example of the model-force predictions is shown in Figure 5-9; the testing calibration data was from subject D in PM. The prediction performance is representative of the other results henceforth it was used for visualization purpose. From all the discussions above, the model is transferable and produces reasonable accurate prediction results (with *MSE* and *R* range from 4.69 N– 11.36 N and 0.979 – 0.982 respectively).

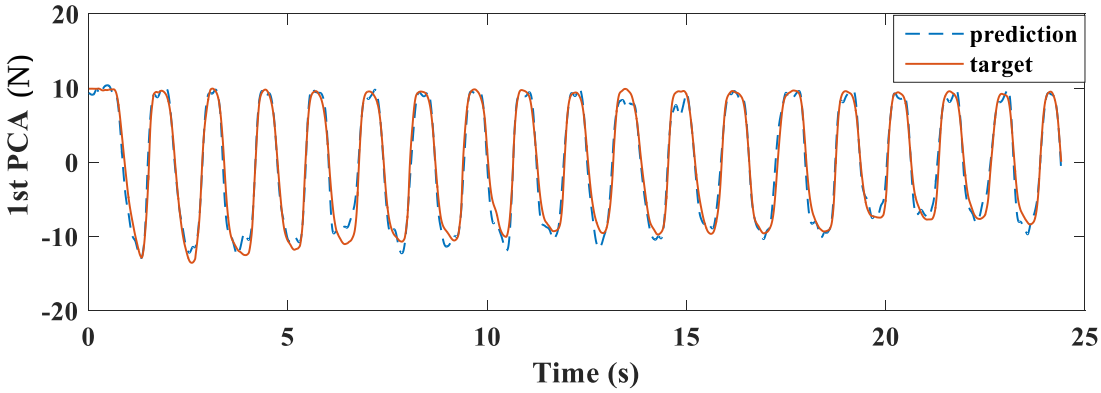


Figure 5-9 An example of the muscle-force predictions results.

5.3.2 Model validation on the actual Peg-in-Hole demonstrations

In section 5.3.1, the proposed muscle-force modelling approach had been tested and evaluated on the test rig in which the muscle activations had been mapped to the corresponding force for three calibration poses. The test rig was used as a general purpose calibration setup because different hand pose variations could be applied on the nodes. Three calibration poses had been defined as a close form of the PiH primitives (section 5.2.1). The model generalisation and prediction accuracy are evaluated against the actual PiH dataset in this section. The input variables were the sEMG signals (*emg*) from the two armbands and hand motion (q). The output signal generated from the model built from calibration poses and the target signal is from the 1st PCA of the F/T sensor (f_{pca}^{1}).

The main focus of this discussion is the influence of the calibration poses and the peg clearances on the model performances. They are discussed in section 5.3.2a), and section 5.3.2b) respectively.

a) Influence of the calibration pose

The model built from the three pressing down poses (A, B and C) were evaluated against the PiH samples for all the subjects, in this section. *MSE* and *R* were the performance metrics. From Table 5-11-Table 5-13, the following results are found:

- For the assembly with loose clearance, the *MSE* from model with pose B contained larger *MSE* error compared with the other poses. Also the *R* values in pose B were smaller. This indicated that the estimations from the pose B's model were not appropriate for PiH task. In the worst case, the *MSE* was 134.54 N and *R* was 0.817. Pose C always has better performances. The best *MSE* was 22.81 N, and the best *R*

was 0.909. This indicates that the force estimations from pose C were closely related to the contact forces in PiH executions. Pose A had the best *MSE* and *R* (17.54 and 0.932 respectively). Also, though the *MSE* were no better than pose C, the R was better. This indicated that pose A was more suitable than pose C since the shape of the prediction curves is closer to the target.

- For the assembly with middle clearance, the pose B had the worst model prediction performance with *MSE* equal to 159.33 N and *R* equal to 0.840. This indicated that the pose B was not suitable for contact force estimations. Pose C had better performance in MSE, but the R were better in pose A. This result indicated that the shape of the predictions in pose A was closer to the targets, but the pose C had better prediction accuracy.
- For the assembly with tight clearance, the results were similar to the discussions in the loose and middle clearances. The pose B was not suitable for PiH task. Pose A had better *R* performance and pose C had better *MSE* performance.

From the results, it was clear that both poses A and C could generate reasonable results for the PiH task. The model predictions for tight clearance in subject A and pose A, and the model predictions for tight clearance in subject A and pose C are shown in Figure 5-10-Figure 5-11. The target signal contained an insertion, extraction, and approaching phases. The focus was only the insertion phase because the approaching phase did not have force measurement and the extraction phase did not require force based control. Therefore, the phases with larger error were approaching and extraction phase in which the estimations could not follow the target well. This caused the degraded *MSE* and *R*. The actual predictions from the insertions contained some error in the peak. This might be caused by the missing detail of the sEMG signals. In general, the prediction results were close to the actual forces in the insertion phase with R ranges from 0.838 to 0.951.

Table 5-11 Result summary for loose clearance assembly. Models were built from different poses A, B and C.

	Pose A		**Pose B**		**Pose C**	
	MSE(N)	**R**	**MSE(N)**	**R**	**MSE(N)**	**R**
Subject A	**17.54**	**0.875**	**31.05**	**0.838**	**27.06**	**0.909**
Subject B	**66.90**	**0.916**	**40.81**	**0.861**	**22.81**	**0.876**
Subject C	**37.99**	**0.926**	**84.55**	**0.856**	**33.57**	**0.893**
Subject D	**56.00**	**0.932**	**134.54**	**0.817**	**35.57**	**0.869**

Table 5-12 Result summary for middle clearance assembly.

	Pose A		Pose B		Pose C	
	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	20.88	0.881	63.04	0.840	24.89	0.879
Subject B	53.19	0.938	159.33	0.845	25.91	0.904
Subject C	49.37	0.951	57.25	0.886	31.13	0.915
Subject D	35.99	0.933	125.0	0.859	34.01	0.898

Table 5-13 Result summary for tight clearance assembly.

	Pose A		Pose B		Pose C	
	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	52.50	0.935	166.46	0.872	28.90	0.855
Subject B	58.26	0.950	102.30	0.882	29.91	0.898
Subject C	66.90	0.916	104.02	0.886	31.28	0.922
Subject D	42.40	0.953	126.55	0.865	30.13	0.911

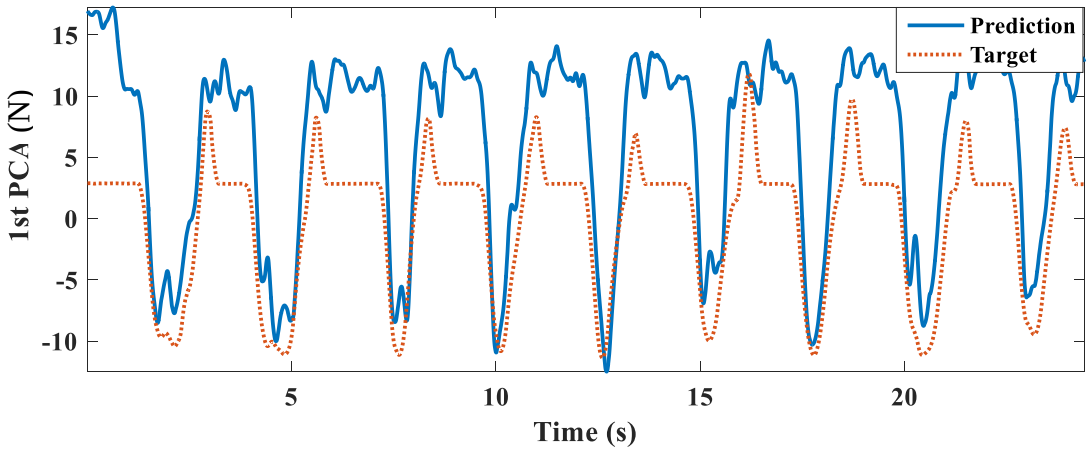


Figure 5-10 Prediction result using pose A performed by subject A for tight clearance.

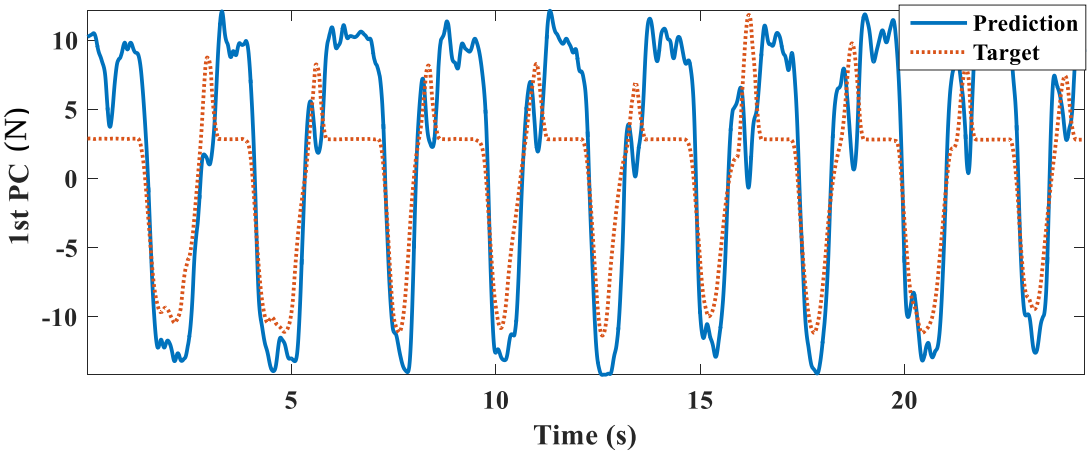


Figure 5-11 Prediction result using poses C by subject A with assembly tight clearance.

b) Influence of the assembly clearance

Three scenarios with different assembly clearances were used to evaluate the model performances. From the results, there was no clear trend that the changing of the assembly clearance would significantly change the model performances. For instance, in the estimations from subject A in pose A, the *MSE* were 17.54 N, 20.88 N and 52.50 N respectively when the peg diameters increased. However, the same trend did not repeat for subject B. In general, the *R* for model built using pose A was better in different clearances, which indicated that the model built from this pose was producing close enough contact forces. Pose C was also a good candidate for predicting the contact force due to its lower *MSE*.

In summary, in Section 5.3.2, the model built from section 5.3.1 had been evaluated against the actual PiH process. The results indicated that the insertion phase of the PiH process could be reliably and accurately predicted by using the simplified calibration poses. If accuracy was the main focus, pose C was the most appropriate posture for muscle-force calibration. The model performance had been verified on the different subjects and the different clearances. They showed that the proposed methodology was capable of generating the specific muscle-force model by using generic calibration test rig.

5.4 Conclusion

In this chapter, research objective two *(to develop a muscle-force model to predict the forces generated from the forearm muscle activations using wearable sEMG devices*) has been addressed. The model will be used to replace the force sensor to avoid the need to measure the force signal in-situ, which may be difficult to achieve. A methodology has been proposed to map the muscle activations from the arm to the contact forces in the peg insertion process. This process contained the skills in which the operator needed to provide compliant peg motions by sensing the contact force and controlling the hand motions. It was shown that the hand motions were not the main predominant features to build the model compared with the sEMG signals.

A calibration test rig had been used for general purpose calibration of the muscle-force relations. Three pressing down poses had been tested and evaluated by different subjects and for different muscle groups. The experiments were executed at different time of the day to evaluate the influence of the environment conditions. The

results from the test rigs showed that the sEMG signals could be reliably and accurately mapped to the contact forces (*MSE* = 2.91 N and *R* = 0.985 in the best case in pose C). Further to these results, the model had been evaluated by using the actual PiH process. Different subjects and clearances had been considered. The results indicated that not all the data from the poses were relevant to the PiH task. The model from pose C showed the capability to generalise across different clearances with reasonable accuracy (*MSE* = 2.91 N and *R* = 0.985 in the best case). The testing dataset contained approaching, insertion and extraction phase, and the model could only produce accurate results for the insertion phase. This was because the extraction and the approaching phases were not considered in the test rig models. The small errors in the insertion phase were due to the missing features of the sEMG signals. The features might not be captured during sampling or has been filtered due to wavelet decompositions. The signals were not able to capture all the corresponding features that generated the exact contact forces.

The proposed methodology for sEMG-force modelling removes the need for the installation of the F/T sensors on the actual task demonstration site. However, for the model to be reliable, indicating that the primitive motions need to be carefully selected and the demonstrations it requires building test rig and defining calibration poses to build the sEMG and force signals off-line. The poses should reflect the primitive motions during the task (the press downing motion in the PiH case). So far, the proposed learning from demonstration framework using wearable sensors has been applied on the peg insertion assembly task. In the next chapter, the framework will be verified and evaluated using a beater winding task.

Chapter 6. Indirect Method of Human-Robot Skill Transfer - A Drum Beater Winding Case Study

6.1 Introduction

In Chapter 4 and 5, the proposed methodology has been applied in the Peg in Hole case study. In this chapter, the aim is to address the research objective four, which is to verify and evaluate the robustness of the proposed framework. To achieve this, the beater winding case study is chosen and discussed in this chapter.

Within the chapter, a systematic framework to learn the winding policy from multiple expert demonstrations is presented. This chapter starts with introducing the research methodology and detailed rationales on the Tension Measurement Unit (TMU) design, the sEMG-force model, the HMM based skills encoding method, and the GMR based motion reproduction method, as shown in section 6.2. The detailed description of the experimental setup is presented in section 6.3.1. The evaluation of the TMU, the accuracy of the sEMG-force model, and the motion reproduction capability are discussed in section 6.4. Human demonstrations are used to validate the model built.

6.2 Case study background

Beater construction [226] for percussion musical instruments is mostly procedural; however, within the winding section of construction, a combination of procedural and tacit knowledge is used. This tacit knowledge is seen in utilisation and modulation of tension while winding the yarn around the beater head. The task is shown in Figure 6-1, where the right-hand changes the head pose, with different angles deviating from the vertical line, and the left hand performs a circular motion roughly in one plane. Procedural knowledge can be easily communicated and written down; Tacit knowledge, on the other hand, reflects accumulated experiences, ways of knowing and cannot be easily expressed [227]. This tacit skill is relatively difficult to maintain and replicate by another trainee. The current operator is due to retire with no possibility for replacement. Therefore, an automatic solution of this winding process is being investigated as a potential alternative to ensure the business is sustainable in the near future.

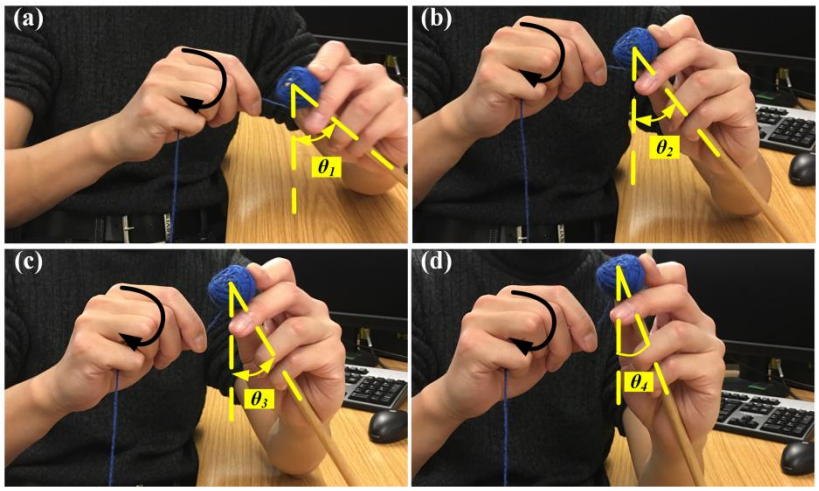


Figure 6-1 Manual drum beater winding process.

The closest automated flexible yarn winding technology in the literature is for cricket ball winding [228]. This technology cannot meet the particular quality standard due to the fundamental geometrical constraints. In order to replicate the flexibility of the current manual process, the automation solution should be delivered in active compliant motion scenario using computer control [229]. Due to the complexity of the process, no passive compliant mechanism will work. Therefore the research from Human Factors (HF) becomes of most interest [230].

In order to understand the process and the skills that required, an HF investigation was performed (in a related research carried out in EPSRC Centre for Innovative Manufacturing in Intelligent Automation) to understand the manual work using a hierarchical task analysis (HTA) and a task decomposition (TD) [38]. HTA is a method for the logical deconstruction of the physical and cognitive components of a task [231]. In [38], the winding task is systematically decomposed into a structure of overall goal, sub-goals, and operations. A TD was then applied to the HTA to extend the data, which breaks the operations from the HTA down further into some categories relevant to the research requirement [231]. These included; the identification of the sensory cues used by the operator, their associated decisions, actions, performance levels (which applies Rasmussen's Skill, Rule, Knowledge (SRK) framework [232][233]), critical values, the cause of process variations, likely errors and error correction. A snapshot of the full TD for "wind appropriate number of vertical winds" is shown in Table 6.1. Full TD is in Appendix C.

Table 6-1 Example Task Decomposition of Beater Construction [38]

HTA Level		Purpose	Beater Variation				Cues	Decision	Action / Response	Performance level	Potential errors	Potential to correct errors
			077 soft hard beater	076 medium beater	071 soft beater	075 hard beater						
2.3.1.2	Wind first three winds securing the thread around the beater then let go of the tail and push it up against the underside of the beater so that it is caught in the next winds	Prevent the end from coming loose and consequently all of the winding					Visual: look to check that the winds overlap to ensure that the tail is caught	Has the end been caught? Or has it slipped out from under the thumb?	While winding make sure that there is some overlap while se	**Rule based**	The tail is not caught in the winds	Catch the tail in subsequent winds
2.3.2	Horizontal winding process											
2.3.2.1	Wind one wind after another shifting the head around using the index finger of the left hand as the thread is wound	Ensure an even coverage					VISUAL: identify any gaps and look at how the winds are lying	Is there an area left uncovered? Are the winds to loose?	Fill as many gaps as possible. Increase tension if winds feel or look to loose.	**Rule based:** act of winding is guided by rules and procedures stored in the operators memory	The coverage is uneven	Rewind ensuring that the winds are more even
							TACTILE: ensure that tension is high but feeling the give in the thread to judge how much more tension it can manage.	Do the winds slip off of the beater head? Does the thread feel like it is straining?	Wind further away from the top of the beater head. Decrease the tension if the thread feels like it is under too much tension and may break.	**Skill Based:** level of tension to apply and where to wind to prevent winds from slipping off of the beater head and to ensure the right appearance	The wrong amount of tension is placed on the thread.	If there is not enough tension then the threads will slip off the beater: tension should be increased. Too much tension and the thread will snap requiring rewinding
2.3.2.2	Hold thumb of left hand against the underside of the beater applying pressure	Prevents the thread from winding down the beater stick and creating a teardrop shaped beater head					VISUAL: identify whether the shape of the beater head is beginning to distort	Is the thread winding down the stick or staying close to the beater head?	Increase the tension of the winds and the pressure applied to the underside of the beater head	**Rule based**	Not enough pressure is applied and shape starts distorting	Start over winding and increase pressure

Although a high proportion of rule and knowledge based operations were found in the HF study [230], the key instances of tacit knowledge identified were the tension maintained on the yarn during winding and the compliant winding motion demonstrated by the human operator. Both hands contributed to maintaining the level of tension. The left hand adaptively changed the angle of the beater head while the right hand accomplished the winding motion. As shown in Figure 6-1, the angle of the beater head is changing continuously and is important to maintain the alignment of the winding around the beater head to achieve a spherical shape. Also it is important to prevent yarn from slipping as well as compactness for sound quality by maintaining tension. To simplify the model, the left hand is assumed fixed in one angle, and only the right-hand accounts for the winding process (generating motions and maintaining yarn tension). Since the winding process is mainly repetitive circular motion in a single plane, the orientation contributed mostly to the movements. In addition to motion data, the sEMG signals were used in this chapter to build the muscle-tension model.

6.3 Methodology

The complete methodology has shown in Figure 6-2, where *emg* is sEMG signal, f_{raw} is the raw output from TMU, f_{raw} is the actual tension, $\theta_{sEMG-force}$ is the model parameter learned off-line, $(\dot{q}, q)$ are the orientation measurements, π is the derived policy, $\hat{q}$ is the evaluation measurements from additional test. First, a TMU is designed to measure the yarn tension during continuous winding without interfering with the hand motion. The sensor readings were calibrated using known force using spring load. An off-line sEMG-force (muscle-tension) model was built to indirectly indicate the tension to sEMG mapping, after calibration with the TMU. The off-line model is built following the methodology in chapter 5 where a calibration test rig is used. Then the kinematic data and the muscle activations were measured on-line for the actual winding process. The samples from multiple demonstrations allow the winding skills to be encoded as probabilistic tension models and the corresponding policy to be derived. In the end, the motions were reproduced for new trials and evaluated against with the reference signal, by assuming the identical physical systems of the human teacher and robot learner.

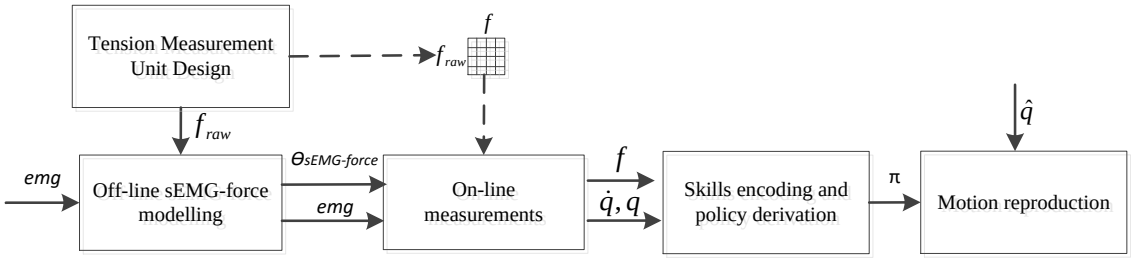


Figure 6-2. An overview of the proposed methodology.

In the off-line model training step, firstly, the selection of the basis function and the level of decompositions (D_l) needed to be evaluated (an example of well-filtered sEMG signals plotted with other signals is shown in Figure 6-3) for filtering the raw signals. Also, the appropriate training parameters including the number of time delay (N_d) and hidden units (N_h) for the TDNN need to be selected. Secondly, the PCA can reduce the dimension of the training data set, thus reducing the overall training time, but too much reduction will degrade the model performance. Therefore, the influence of PCA has to be considered carefully. Thirdly, it is unclear which group of muscle contributes more to the model. Therefore, the influence of the number and position of the sEMG sensors need to be established. Fourthly, the operator might have to work at different times of the day; it was not clear whether the model is sensitive to the environment condition. Lastly, the model built from the test rig required validation on the actual beater winding data. In the on-line validation step, individual subject is asked to perform the winding task several times. The sample trials were encoded for learning the skill models and validated in the reproduction step by using testing trials.

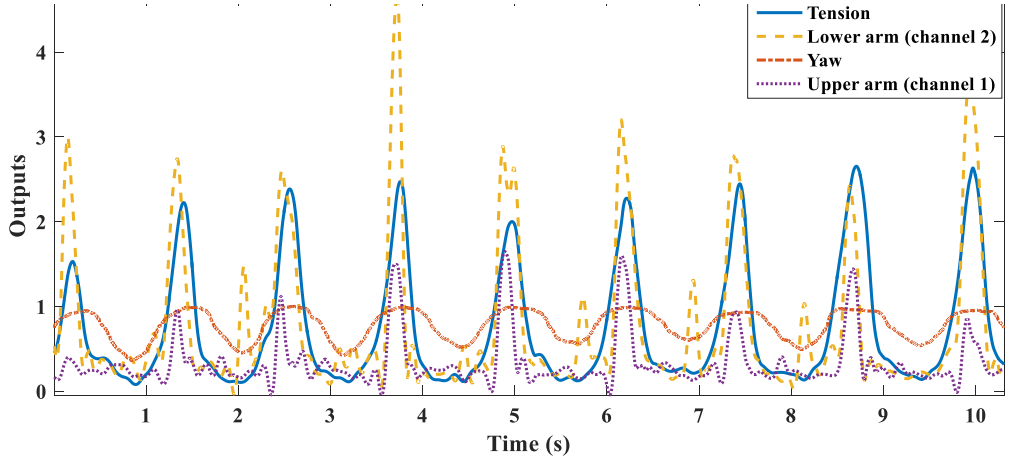


Figure 6-3 One example of filtered signals from sEMG, IMU and $f_{raw}$from tension measurement unit after normalisations. Note, the output is sensor reading and unitless.

6.3.1 Experiment setup

In this section, the experiments were designed to build muscle-tension model and skill transfer model. As shown in Figure 6-4, two sEMG sensors with 8 channels each were used to measure the muscle activation in the human lower arm and upper arm. One Vicon-IMU system was used for hand pose tracking. The tension measurement unit was used to imitate the actual beater winding process and produced tension measurements during simple winding pose around the roller (in one plane). The sampling rate of the sEMG sensors are 200 Hz. The sampling rate of the load cell is 400 Hz. The sampling rate of the Vicon-IMU system is 60 Hz. The sEMG and tension signals were synchronized by using ROS approximate synchronizer algorithm. The sampling rate of the synchronized signals is 200 Hz. The hand motion was slow and does not require high sampling rate. Therefore, the synchronizer did not consider the timestamps from the Vicon-IMU. The pose signals were stored independently from the signals after synchronisation. At each timestamp, the synchronizer grabbed the sEMG and tension measurements, the most up to date pose data were assigned. This method minimises the time shifting of the sEMG and the tension readings. To evaluate the proposed methodology, 4 operators from Intelligent Automation lab have attended the experiments as follows. They were age from 25~27 and had engineering and manufacturing expertise.

- TMU calibration by pulling a spring load in the horizontal direction. The pulling force is gradually increasing by 0.5N until the breaking point (13N) of the yarn achieved.

- Replace the left hand by using a fixture with a dummy roller to minimise the disturbance of the left-hand movements as shown in Figure 6-5. Holding the yarn and keeping it intense with the right hand; then wind 20 cycles on the roller. Three trials were collected for training, and one trial is collected for testing.
- Repeat experiment two 6 hours later in the same day and then evaluate the muscle-tension model build from experiment 2. Two trials with at least 15 windings were collected for testing.
- Install the beater head onto the robot end effector and bring it close to the dummy roller to minimise the variation of the arm poses when performing winding as shown in Figure 6-6. Two trials with at least 15 trials have collected for verification of the model built from experiment two. Since the F/T sensor contains 3 axes of measurements, in order to make the prediction results comparable, the force composition has applied in the following: $f^{F/T} = \sqrt{{f_x}^{F/T^2} + {f_y}^{F/T^2} + {f_z}^{F/T^2}}$. The Mean Square Error (MSE) and Regression Coefficient (R) of the $\hat{f}$ and $f^{F/T}$ are the performance evaluation metric.
- Perform two trials of beater windings for learning skills by policy derivation and evaluating the policy performance. Each one contains 15 winding cycles.

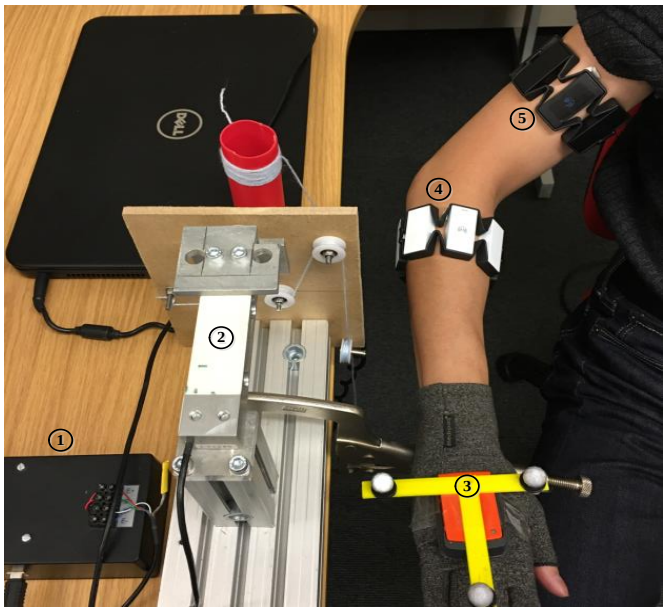


Figure 6-4 Experiment setup. 1) Signal conditioner. 2) Tension measurement unit. 3) Vicon-IMU system. 4) The White sEMG sensor on the lower arm. 5) Black sEMG sensor on the upper arm.

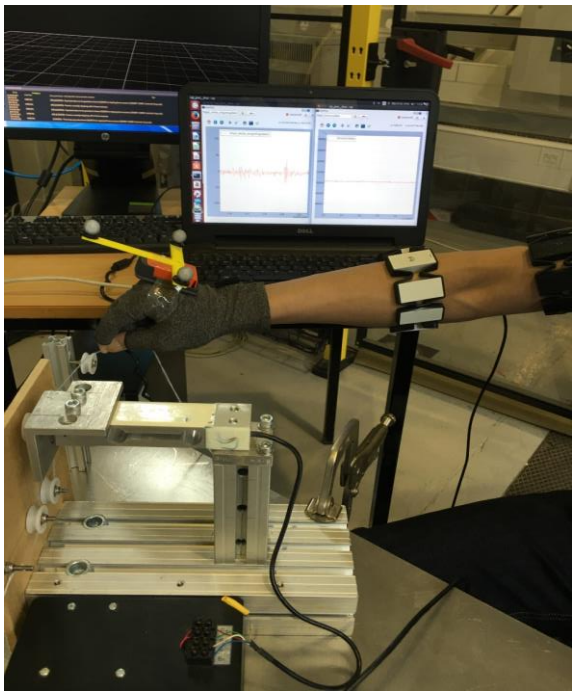


Figure 6-5 Experiment 2. The operator is winding on the tension measurement unit by using a fixed dummy roller.

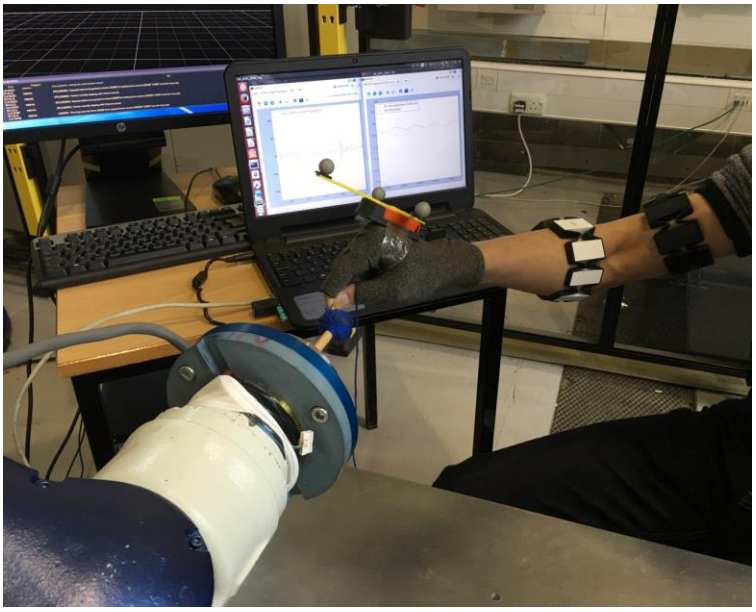


Figure 6-6 Experiment 3. The operator was winding the drum beater head with his left hand fixed (held statically by the robot) as in experiment 1.

6.3.2 Tension measurement unit design

In order to build the off-line model, a tension measurement unit (TMU) was designed. However, measuring tension in a continuous winding process is a challenge. One design has been proposed in our previous work [234], where a handheld sensor was developed by using a miniature load cell and a tweezer-like mechanism. The design aimed to measure the tension on the yarn directly during winding. However, since the sensor body parts are 3D printed and the frictions corrupt the signals, the design has been refined in this chapter. In this section, the development of a tension measurement unit is explained. As shown in Figure 6-7, the design procedures are split into 5 blocks.

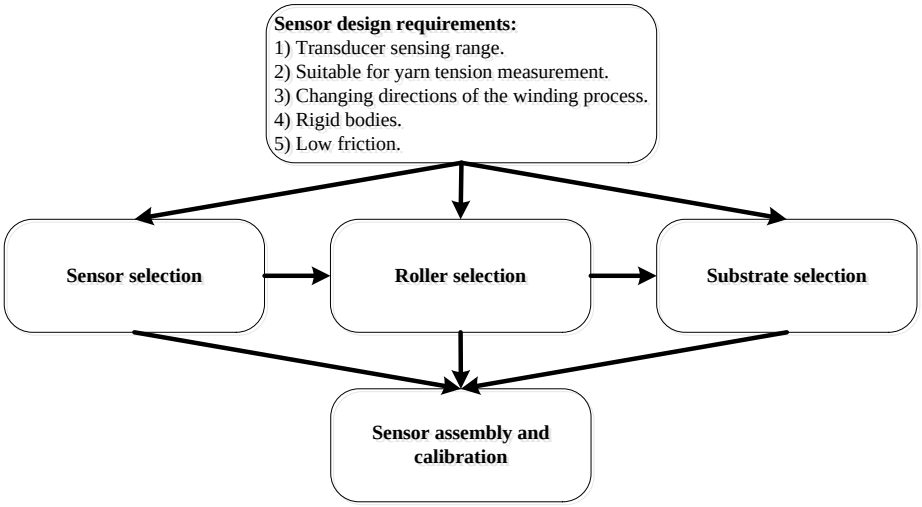


Figure 6-7 Sensor design methodology.

Sensor design requirements: the first consideration is the sensing range. The choices of the transducers are plenty, but without considering the sensing range, the sensor might not be able to give sufficient readings. The selection standard of this range depends on the tension at the breaking point of the yarn. The second consideration is that the yarns are flexible. Therefore, the measurement unit should keep the yarn in tension so that the reading reflects the real tension measurements. The third consideration is the direction of the winding process. It changes with time; therefore, the design should allow the operator to pull the yarns in multiple directions. The fourth consideration is the sensing elements. They should be as rigid as possible to assure the repeatability of the readings. The fifth consideration is the frictions, which should remain low in the system.

Transducer selection: according to the requirements, the sensor has to be in the right sensing range and applicable for tension measurements. A load cell can provide downward force readings on one end. The transducers on the load cell are the strain gauges with a Wheatstone bridge connection. Among the different bridge connections, a full bridge with 4 active strain gauge sensing elements can generate the measurements with the most sensitive reading and temperature compensations. A roller is essential to allow the yarns to run smoothly. Therefore, a full bridge load cell with a roller head is considered as the sensing bodies designs.

Roller selection: the main concern in this part of design is the size and the number of the rollers. A big roller is not necessary, but a small one might not be able to hold the yarns securely. The diameter of the roller should be slightly larger than the height of the load cell, and the groove depth should be enough for the yarns to run through without slipping. The number of the rollers used was 4. One was used for sensing; two were fixed as dummy sensors to assure the magnitude of the readings and the tension calculations. An additional dummy roller was also needed to change the direction of the yarns.

Substrate selection: the load cell and the sensing roller needed to be installed on a rigid body to assure the repeatable measurements. This required a rigid substrate. Additionally, the dummy rollers needed to be installed on a detachable wall because they should not generate weight to the load cell and the direction of the yarns should be fixed.

Sensor assembly and calibration: the selected parts were assembled, and the sensor readings were calibrated against with known force.

Calibration of the sensor can be done by pulling the yarn in one end using spring load. The reading from the spring load is the true tension for the current sensor measurements. By gradually pulling the yarn, the tension increases. At each time where the spring load stopped, the tension was recorded 5 times to estimate the actual value. The maximum tension should be within the breaking point and covered the full range of the yarn strength. After this process, the characteristic of this TMU was evaluated. The calculation of the tension is shown in equation (6-1) and the schematic plot is shown in Figure 6-8.

$$F3 + friction = F2$$

$$(F1 + F2) \times cos\theta + friction = F4 \quad \text{.....(6-1)}$$

where θ is the angle between $F1\ and\ F3$. If friction is ignored and $\theta = 180°$, $F2 = F3$. The advantage of having those dummy rollers is that the direction of $F3$ is allowed to change. Now, $F4 = F1 + F2$. Since $F1 = F2$, therefore, $F4 = 2 \times F3$.

This tension measurement setup allows the operator to perform winding on a dummy roller 3 while tension is measured at sensor roller, *F4*. The whole process

should keep the yarn in tension, assuming low friction the tension in *F3* can be calculated from *F4*.

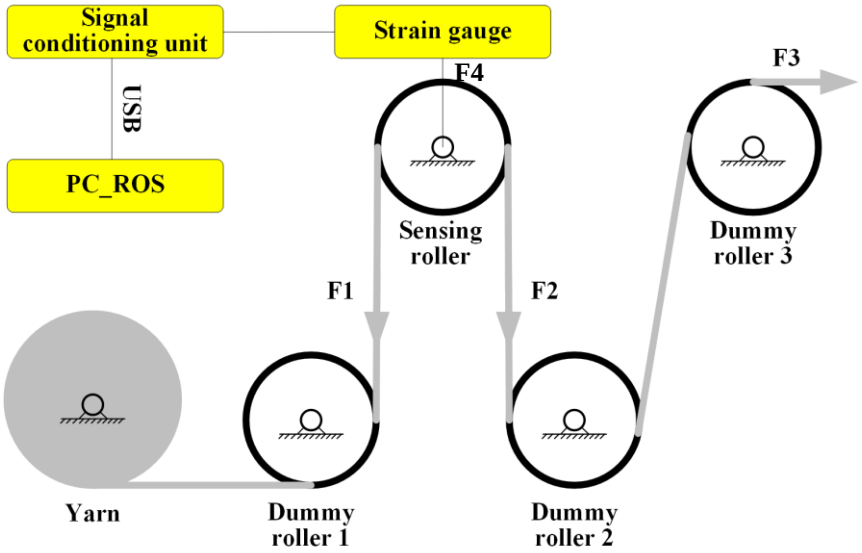


Figure 6-8 Schematic plot of the sensor design.

The sensor is shown in Figure 6-9. The load cell was rigidly bolted on the aluminum beam with the base. An aluminum adaptor plate was installed on one end of the load cell with a roller bolted. The other three dummy rollers were bolted on one piece of Medium-Density Fibreboard (MDF) which was also bolted to the base frame.

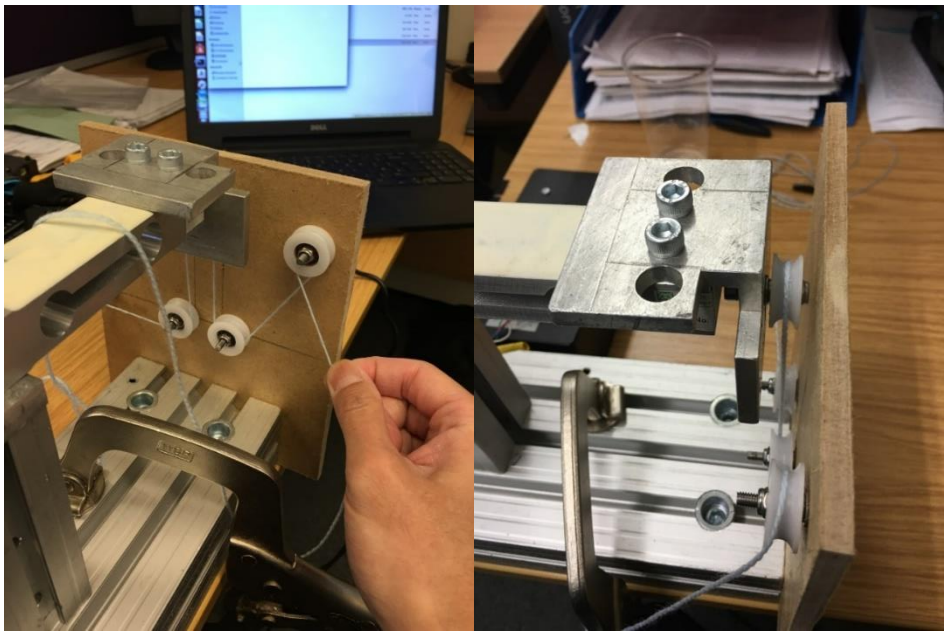


Figure 6-9 Sensor after manufacturing. View 1(left) and View 2(right).

The calibration results and fitted sensor-tension relations are shown in Appendix D. The fitting function is:

$$y = 4.73x + 8.38 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(6-2)}$$

After the design of the sensor, calibration data was collected to train the muscle-tension (sEMG-force) model. As a result, an adaptive control policy from multiple human demonstrations was derived by taking the inputs from haptic tension feedback and the hand motions, in order to reproduce the compliant movements in the manual beater winding process.

6.3.3 Off-line sEMG-force modelling

The sEMG-force modelling method contains three steps, involving the: 1) data rectification and filtering due to the noises sEMG signals, 2) principle component analysis (PCA) to reduce the variable dimensions and training time, and 3) employment of the time delayed neural network (TDNN) for non-linear function approximation. Details can be found in Chapter 5.2.2. Different from the PiH case study, the motion primitive for the winding process is cyclic movement which contains small variations in the absolute positions and large variations in the hand orientations. Henceforth, the input signals are $\{emg, q\}$ and the output signal is f_{raw}. 'Daubechies 6' (*db6*) basis wavelet and the 4th level approximation of the original signals were chosen heuristically as the data filtering strategies. Each armband has 8 channels of sEMG signals. PCA was applied to the 8 channels sEMG signals to reduce the dimensions. The Time Delayed Neural Network was then applied to find the relationship between the sEMG and the force measurement from TMU. After calibration on the test rig, the model needs to be validated on the actual winding process where the actual reference force measurement is given by the composed force vector, which will be explained in the experiment setup section.

6.3.4 Policy derivation and motion reproduction

The dataset was composed as a set of data points $x = (\dot{q}, q, f)$. Different from PiH task, the winding process is a cyclic process. It is possible to decompose the complete winding process into one unique cycle with many repetitions. In each cycle, the winding can be roughly classified into two distinct states: forward winding and backward winding. However human cannot generate a unique winding pattern over and over again. Therefore, it is important to model the state transition within each cycle of winding which requires more states to represent the different cycles. Also, it is important to model the state transitions in between the cycles because the end of the last

cycle means the upcoming of the next cycle. Therefore, the training dataset should contain more than one cycle and remain the nature of the state transitions in between the cycles.

The Hidden Markov Method (HMM) is suitable of learning the episodic behaviour by encoding the state transitions. It learns a joint probability model $P(\dot{q}, q, f)$ with *K* state, where the output was the Gaussian distribution of each state that representing the locally correlation between different variables. The parameters of the model is $\{\Pi,\ T,\ \mu, \Sigma\}$ learned through Baum-Welch algorithm [215], where h is the hidden component, $\Pi = \mathrm{p}(h_0)$, $T \sim p(h_t|h_{t-1}, \theta)$, and $(\mu, \Sigma) \sim p(x_t|h_t, \theta)$. $\{\Pi,\ T,\ \mu, \Sigma\}$ can be initialised by using *K*-mean algorithm followed by Gaussian Mixture Model. This step will accelerate the HMM training process and learn the inherent time dynamics in the dataset. As a remainder, the different variables of the dataset and associated model are labelled separately as:

$$\begin{bmatrix} x^I \\ x^O \end{bmatrix} = \begin{bmatrix} q \\ f \\ \dot{q} \end{bmatrix}, \begin{bmatrix} x^{I'} \\ x^{O'} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ f \\ q \end{bmatrix} \quad \text{……...……..……(6-3)}$$

$$\begin{bmatrix} \mu_i{}^I \\ \mu_i{}^O \end{bmatrix} = \begin{bmatrix} \mu_i{}^q \\ \mu_i{}^f \\ \mu_i{}^{\dot{q}} \end{bmatrix}, \begin{bmatrix} \Sigma_i{}^I & \Sigma_i{}^{IO} \\ \Sigma_i{}^{OI} & \Sigma_i{}^O \end{bmatrix} = \begin{bmatrix} \Sigma_i{}^q & \Sigma_i{}^{qf} & \Sigma_i{}^{q\dot{q}} \\ \Sigma_i{}^{fq} & \Sigma_i{}^f & \Sigma_i{}^{f\dot{q}} \\ \Sigma_i{}^{q\dot{q}} & \Sigma_i{}^{\dot{q}f} & \Sigma_i{}^{\dot{q}} \end{bmatrix} \quad \text{…….……(6-4)}$$

$$\begin{bmatrix} \mu_i{}^{I'} \\ \mu_i{}^{O'} \end{bmatrix} = \begin{bmatrix} \mu_i{}^{\dot{q}} \\ \mu_i{}^f \\ \mu_i{}^q \end{bmatrix}, \begin{bmatrix} \Sigma_i{}^{I'} & \Sigma_i{}^{IO'} \\ \Sigma_i{}^{OI'} & \Sigma_i{}^{O'} \end{bmatrix} = \begin{bmatrix} \Sigma_i{}^{\dot{q}} & \Sigma_i{}^{\dot{q}f} & \Sigma_i{}^{\dot{q}q} \\ \Sigma_i{}^{f\dot{q}} & \Sigma_i{}^f & \Sigma_i{}^{fq} \\ \Sigma_i{}^{\dot{q}q} & \Sigma_i{}^{qf} & \Sigma_i{}^q \end{bmatrix} \quad \text{……..(6-5)}$$

where the state variables are absolute angles $(q_{yaw}, q_{pitch}, q_{roll}) \in q$, angular velocity $\dot{q}$, and tension *f*. The upper case *I* and *O* represents input variable and output variables. In these two different set of variables, tension *f* is always used as one of the inputs, while the angle q and $\dot{q}$ are used interchangeably to estimate $\dot{q_d}$ and q_d.

During the reproductions, at each time step the current observations $x = (\dot{q}, q, f)$ is used to define a weight factor w_i, representing the impact of the *i-th* state:

$$\beta_i(x_t) = \frac{\alpha_{i,t}}{\sum_{k=1}^{K} \alpha_{k,t}} \quad \text{…….. ……..(6-6)}$$

with $\alpha_{i,t} = \left(\sum_{k=1}^{K} \alpha_{k,t-1}\alpha_{ki}\right)\mathcal{N}(x_t; \mu_i, \Sigma_i)$,

Where $\alpha_{i,t}$ is the forward variable (calculated recursively from HMM representation) corresponding to the probability of partial observations $x = \{x_1, x_2, \ldots, x_t\}$ of length *t* being in state *I* at time *t*.

A target angle $\hat{q}$ and angular velocity $\hat{\dot{q}}$ are estimated through GMR as:

$$q_d = \sum_{k=1}^{K} \beta_i(x)\left(\mu_i{}^{O} + \Sigma_i{}^{OI}\Sigma_i{}^{I}(x^I - \mu_i{}^{I})\right), \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(6\text{-}7)$$

$$\dot{q_d} = \sum_{k=1}^{K} \beta_i(x)\left(\mu_i{}^{O'} + \Sigma_i{}^{OI'}\Sigma_i{}^{I'}(x^{I'} - \mu_i{}^{I'})\right), \ldots\ldots\ldots\ldots\ldots\ldots\ldots..(6\text{-}8)$$

From the current position and velocity of the system, an impedance controller similar to a mass-damper system is computed to reach the desired angle $\hat{q}$ and angular velocity $\hat{\dot{q}}$. Then the acceleration control command in the task space is defined as:

$$\ddot{q} = \overbrace{\left(\hat{\dot{q}} - \dot{q}\right)k_v}^{\ddot{q}^v} + \overbrace{\left(\hat{q} - q\right)k_p}^{\ddot{q}^p} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(6\text{-}9)$$

$$\dot{q_t} = \dot{q}_{t-\Delta t} + \Delta t\ddot{q}_t, q_t = q_{t-\Delta t} + \Delta t\dot{q}_t \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots..(6\text{-}10)$$

Where k_v and k_p is the control gains. In equation (6-9), $\ddot{q}^v$ allows the learner to follow the dynamic, and $\ddot{q}^p$prevents the learner deviating from the unlearned situation and remaining in the existing context if any perturbation occurs.

6.4 Result analysis

6.4.1 Muscle tension model training

In this section, a muscle tension model is built based on the muscle activations, hand poses, and the tension measurements. The *Mean Square Error (MSE)* and regression coefficients were used as performance metrics. *MSE* is the error between the prediction and the sensor reading. Regression coefficient (*R*) indicates the similarity in between the prediction and the target.

a) Selection of the filtering and TDNN training parameters.

In this section, to address the filtering and training parameters, the dimension of the training data sets are not reduced. Both sEMG sensors were used for training. The wavelet functions were chosen from ['*db*4', '*db*5' and '*db*6']. The levels of decompositions were chosen from $D_l = \{4, 5\ and\ 6\}$. The N_d was chosen from $N_d =$

$\{0, 0{:}5, 0{:}10 \text{ and } 0{:}15\}$. N_h was chosen from $N_h = \{1,2\ and\ 3\}$. The results from subject A (Table 6-2 summarised the rest of the subjects) are used as an example and shown in Figure 6-10. In each plot, one pair of wavelet basis function and the level of decomposition were chosen. The plot shows the *MSE* performance by varying the number of the hidden units and the number of time delays for TDNN. The results indicate:

- The best choice of the basis function and the level of decomposition are '*db*5' and 4 respectively. $N_h = 2$, $N = [0{:}15]$, *MSE* = 0.046N, and *R* = 0.88.
- The worst choice of the basis function and the level of decomposition are '*db*4' and 6 respectively. $N_h = 2$, $N_d = [0{:}5]$, *MSE* = 0.059N, and *R* = 0.70.
- There is not much improvement by selecting a larger N_h. Even though the best results indicate $N_h = 2$, $N_h = 1$ seems acceptable across all the results. Also, all the plots indicate a worse performance when $N_h = 3$.
- The factor N_d has influence on all the results. Increasing N_d will improve the performance in most situations (e.g. when '*db*5' and $D_l = 4$, and '*db*6' and $D_l = 6$ are chosen).

From the results above, they indicate that to achieve a reasonable model, one need to select the filtering and training strategies properly. If the sEMG signals are over filtered with a larger decomposition level, it will remove too many essential details for learning the model. On the other hand, unnecessary details will corrupt the signal and lead to a degraded model. During the inspection of the benchmarking, the selection of the wavelet basis does not have major impacts on the model prediction performance. Henceforth, one can heuristically select the appropriate wavelet functions. The same strategy applies to the rest of the subjects. Summarised results are shown in Table 6-2 where the training parameters are selected for all the subjects after benchmarking. From the results:

- '*db*5' wavelet function is suitable for both subject A and B. '*db*6' wavelet function is appropriate for both subject C and D. '*db*4' is not an appropriate basis function.
- The 4th level decomposition applies on all the subjects. This indicates that further reduction of the level of details will degrade the model prediction accuracy.

- N_h is small in all cases. $N_h = 1$ seems a good choice. $N_d = [0{:}\,15]$ for both subject A and B, and $N_d = [0{:}\,10]$ for both subject C and D. Different from the standard neural network with $N_d = [0]$, a TDNN considers using a sliding window from the past to predict the current output value. The variations in N_d indicate that the data beyond the optimal sliding window size degrade the prediction performance.

In conclusion, the *MSE* and regression results in the best case (*MSE = 0.046N* and R = 0.88*)* suggest that the prediction is close to the target signal. The results indicate a simple network structure and the model is a weighted superposition of linear systems with a number of time delays. This might because the winding task is relatively simple, and the muscle activations provide sufficient information to build the reasonable muscle-tension model.

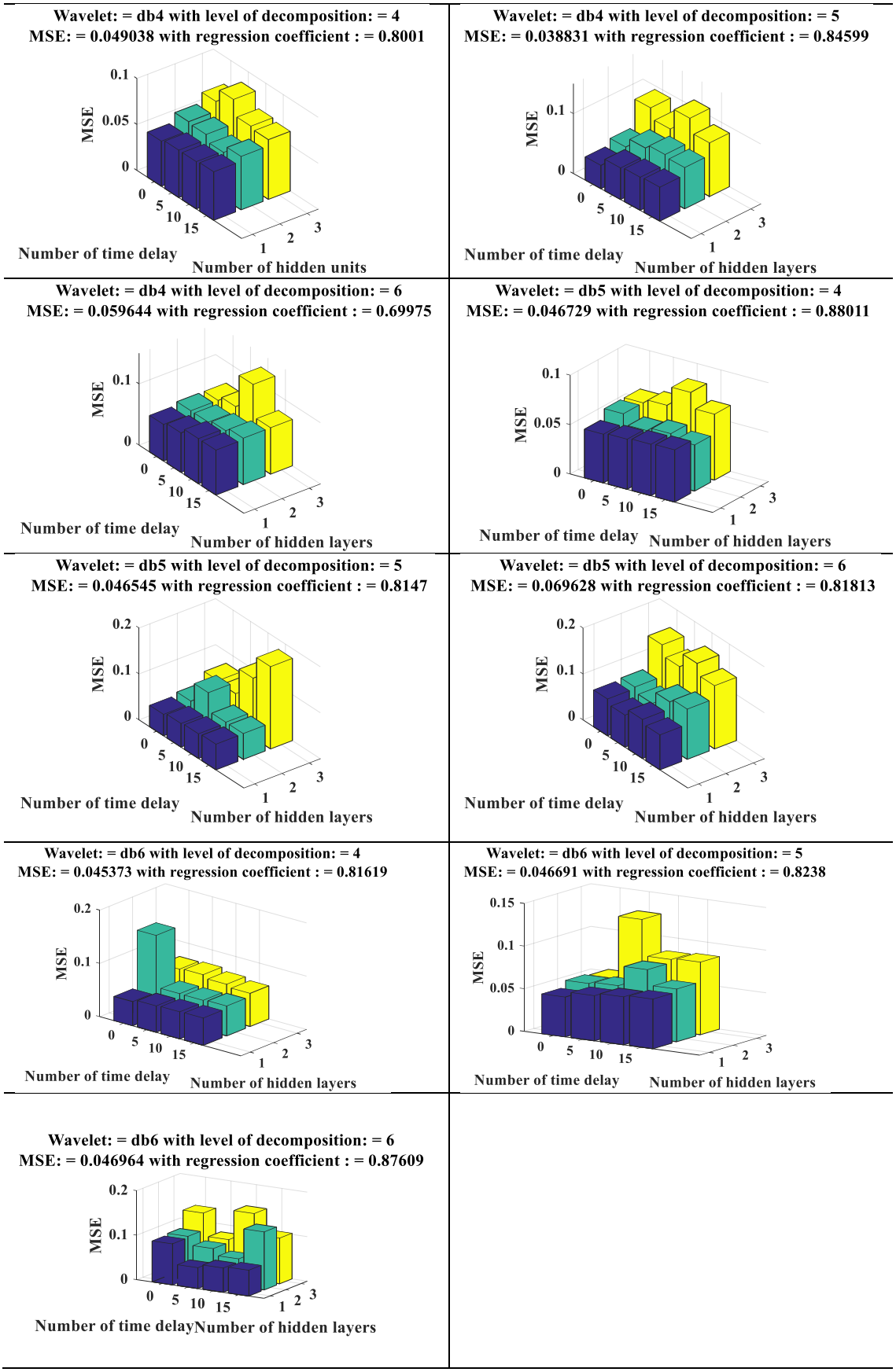


Figure 6-10 Benchmark for parameters selections for subject A.

Table 6-2 A summary of the benchmarking results across subjects.

	Wavelet	Decomposition level	Number of time delay	Number of hidden units
Subject A	**db5**	**4**	**15**	**2**
Subject B	**db5**	**4**	**15**	**1**
Subject C	**db6**	**4**	**10**	**1**
Subject D	**db6**	**4**	**10**	**1**

b) Influence of Principle Component Analysis (PCA)

PCA can reduce the dimension of the data set and potentially reduce the amount of training time. The filtering and training parameters were selected from the results in the last section, and they will be applied to the rest of the result discussions. In this section, two scenarios are tested where the raw training data was processed without PCA and with PCA (1% reduction). 1% is heuristically selected to show the influence of the data dimension reductions. Because training without PCA consumes much more time than using PCA, the training time when 100 iterations reached was used. In all cases, the model training with PCA took less than 100 steps. As shown in Table 6-3:

- PCA will improve training efficiency. It only took 2 seconds to complete training with PCA; but in the worst case, it took 25 seconds to finish 100 iterations for subject A.
- The *MSE* and regression coefficient results indicate that the predictions with PCA are less accurate than the predictions without PCA. But the MSE (0.03N, 0.01N, 0.02N and 0.02N for subject A, B, C and D) and R (0.04, 0.026, 0.013 and 0.033 for subject A, B, C and D) differences are small. The results indicate applying PCA will reduce the accuracy, but the effect is small.

Table 6-3 Model training performance with and without PCA.

	With PCA (99%)			Without PCA		
	MSE(N)	R	Time (s)	MSE(N)	R	Time(s)
Subject A	**0.08**	**0.840**	**2**	**0.05**	**0.880**	**25**
Subject B	**0.05**	**0.929**	**1**	**0.04**	**0.955**	**11**
Subject C	**0.09**	**0.943**	**1**	**0.07**	**0.956**	**14**
Subject D	**0.08**	**0.927**	**2**	**0.06**	**0.960**	**21**

c) Influence of muscle groups

Two groups of muscles are considered in the experiment: the biceps and triceps in the upper arm, and extensors and flexors in the lower arm. It is unclear which group of muscle contributes more to the model. In this section, models from the individual

muscle group data were trained. The same training parameters addressed from section 6.4.1.a) were used here. The data recorded are the hand motions and muscle activations for the whole arm. As shown in Table 6-4, the models are: Lower arm muscle with a hand motion (*L_M*), upper arm muscle with a hand motion (*U_M*), hand motion only (*M*), whole arm muscle with hand motion (*W_M*) and whole arm muscle only (*W*). From the results:

- In all cases, the prediction results from the *W_M* model show better performance. The most accurate model is built from subject B where the *MSE* is 0.04N, and *R* is 0.955. This indicates that the training model can accurately and reliably estimate the tensions measured by the tension measurement unit.
- The model built from the *L_M* is slightly less accurate than the model built from the whole arm with the hand motion. For instance, subject D has an *MSE* equal to 0.07N in the *L_M* while his *W_M* model has *MSE* = 0.06N; Subject C has the same performance in his *L_M* model.
- The *U_M* model does not have the same level of accuracy compared with the *L_M* model. For instance, subject D has *R* = 0.88 in his *U_M* model while the *R* value in the *L_M* model equals to 0.95. This means the *L_M* model provides more valuable force information than the *U_M* model.
- The motions only model *M* has the least accuracy and none of them are close to the best model in *W_M* which indicates that hand motion is insufficient to learn the model. The model prediction performances from the *W* model are slightly worse than the *W_M* model (in the worst case, the difference in *MSE* is 0.05N). This indicates that it is essential to have muscle activations to predict the tension force. However, hand motion data are also important since they help to improve the model performance, even though *M* alone cannot achieve reasonable accuracy.

From the above, the results indicate that the hand motion data are essential to building the tension model. This is because the motion data have clear features to differentiate the internal winding states. The muscle activations further improve these features and lead to a more accurate model. For this winding application, the muscle groups from the lower arm have more contributions to the final muscle-tension model. Therefore, the sEMG sensor installed on the upper arm can be potentially removed.

Table 6-4 Results summary on the influence of the muscle groups.

	L_M		U_M		M		W_M		W	
	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R	MSE(N)	R
Subject A	**0.08**	**0.840**	**0.11**	**0.813**	**0.12**	**0.790**	**0.05**	**0.880**	**0.10**	**0.85**
Subject B	**0.04**	**0.958**	**0.11**	**0.929**	**0.14**	**0.899**	**0.04**	**0.955**	**0.05**	**0.93**
Subject C	**0.09**	**0.956**	**0.15**	**0.880**	**0.22**	**0.820**	**0.09**	**0.956**	**0.08**	**0.950**
Subject D	**0.07**	**0.950**	**0.13**	**0.884**	**0.19**	**0.81**	**0.06**	**0.960**	**0.07**	**0.932**

d) Influence of the environment condition

Each subject repeated the experiment in two different times as described in the experiment setup with a fixed sensor location. The wearable sensors placements are assumed unchanged, and the operator is following similar pattern of winding. The discussion in this section aims to evaluate the effect of the environment condition (mainly when the experiment performed) on the built model. One trial of data is selected as testing data in *AM*, two trials of data are used as testing data in *PM*. As shown in Table 6-5, the *MSE* and *R* values from *PM* are slightly worse than *AM* in some of the cases. For instance, the testing data performed by subject C from trial 1 in the *PM* had *MSE* = 0.14N while in the *AM* he had *MSE* = 0.09N. In subject B and D, the *MSE* and *R* values are close. These results indicate that the model built from training data in the *AM* is still valid in the *PM*.

Table 6-5 Result summary on the influence of the different experiment time. AM represents a time in the morning. PM represents a time in the afternoon.

	AM		PM (Trial 1)		PM (Trial 2)	
	MSE(N)	R	MSE(N)	R	MSE	R
Subject A	**0.05**	**0.880**	**0.14**	**0.847**	**0.20**	**0.840**
Subject B	**0.04**	**0.955**	**0.09**	**0.937**	**0.04**	**0.961**
Subject C	**0.09**	**0.956**	**0.14**	**0.890**	**0.09**	**0.923**
Subject D	**0.06**	**0.960**	**0.23**	**0.951**	**0.24**	**0.954**

6.4.2 Model validation on actual beater winding

In this section, each subject is asked to perform the actual beater winding to evaluate the model built from the test rig. As shown in Figure 6-6, the actual drum beater was firmly attached to the F/T sensor which is installed on the robot end effector. The reason of doing this is that the pose of the beater head can be easily adjusted close to the roller winding experiment. The subject was asked to perform a similar pattern of the winding by keeping the arm pose steady. The target signal is $f^{F/T}$ after compositions as mentioned in experiment 4. In order to compare with the targets, the predictions need to project onto the actual force according to the linear relationship in

equation (6-2). Two trials with at least 15 windings were used for evaluation purpose, and the average *MSE* and *R* values of these two trials are shown in Table 6-6:

- Subject A has the smallest *MSE* = 0.28 N, while subject D has the larger *R* = 0.90 but largest *MSE* = 0.55N. This means that the shape of the predictions from subject C is linearly closer to the actual force signal, but they are less accurate. One reason of this is that the overall shapes of the predictions are slightly shifted upward or downward due to the fact that the actual winding process has misalignment with the roller winding. In another word, the subject was not exactly following the same pattern in the actual winding when he was performing the roller winding.

Table 6-6 Results summary for the actual beater winding process.

	MSE(N)	R
Subject A	**0.28**	**0.890**
Subject B	**0.31**	**0.834**
Subject C	**0.30**	**0.91**
Subject D	**0.55**	**0.90**

6.4.3 Skills encoding and motion reproduction

The yarn tension plays a major role in the winding process. Direct measurement of the tension is challenging because it is difficult and impractical to install a sensor to dynamically measure the tension in between the small gap between the beater head and the hand. In the previous section, the muscle-tension model was built to predict the tension from indirect test rig measurements. The model allows the operator to wear the sEMG sensors and the Vicon-IMU system to demonstrate the task freely without physical obstructions.

In this section, the winding skills are encoded by following the proposed method in section 6.3.4. Firstly, the HMM based approach was applied to capture the skills and decompose them into states representations. Since the winding process is mainly cyclic motion (as shown in Figure 6-1) which contains two main phases: forward and backward winding. Therefore, at least two states are expected. However, since the subject would not repeat exactly the same winding pattern in each cycle, some variations in each phase are expected. Due to the repetitive nature, the number of states is limited to five, which was heuristically derived from all the subjects' executions and fixed for all the subjects.

The main concern of this section is to use the probabilistic encoding of the skills and reproduce compliant hand motions from the predicted tension profiles. The quality of the reproduced trajectories is evaluated by additional testing winding trials from the same 4 subjects. The motions in these trials are considered as the optimal target motions. *MSE* and *R* are used as performance metrics.

To begin with, a benchmark of the control gains *kv* and *kp* (as introduced in section 6.3.4) were selected to control the reproduction process. These two parameters acted as a spring-damper system to smoothen the reproduced trajectories. They helped to trade off the control signals between the system dynamics and the system error.

From the results in Table 6-7, the values of *kv* and *kp*, and the corresponding best reproduction results from the motions in the test trials are listed. For conciseness, Figure 6-11-Figure 6-14 plot the tension-motion relationship for one complete winding cycle by subject A-D. A summary of model performances for all subjects is shown in Table 6-7. They indicate that:

- The optimal control gains were different in between subjects. The same pair of parameters does not repeat. This indicates that the reproduction strategy has to be changed according to these gains so that they adapt to the different input signals from the new testing trials.
- For subject A, the *MSE* for *yaw*, *pitch*, and *roll* are *1.14*, *0.57* and *0.86* in *degree*. The *R* are *0.75*, *0.77* and *0.87*. The reproduction of the tension-motion relations is shown in Figure 6-11. From the results, the new winding data is within the modeled Gaussian distributions. The reproduction attempts do not follow exactly the target trajectory which means the learner is trying to reproduce the motion in a different manner. However, the *MSE* error and *R* values indicate that the reproductions are following the dynamics of the winding process and respond to tension measurements in an adaptive manner.
- For subject B, the *MSE* for *yaw*, *pitch*, and *roll* are *11.46*, *8.60*, and *11.46* in *degree*. The *R* are *0.870*, *0.653* and *0.697*. The reproduction of the tension-motion relations is shown in Figure 6-12. From the results, the new winding data is not within the modeled Gaussian distributions. The performance of the reproduction attempts are not as good as subjects A, but the *R* value *(0.87)* in yaw is higher and the *MSE (0.20)* is smaller. This indicates that the training and testing sets have similar yaw

angle patterns. But the degraded *pitch* and *roll* angle reproductions indicate that the testing set contains larger data variations compared with the training set.

- In subject C, the *MSE* and *R* are better than subject B. The reproduction of the tension-motion relations is shown in Figure 6-13. The *roll* angle contains the largest error (*MSE = 5.73 in degree*) in the motions, but the overall reproductions are following the target in a good accuracy. The reproductions contain larger error in the *yaw* and *pitch* an angle, which indicates that the learned model is not capable to fully reproduce the motions given the new data inputs.
- For subject D, from the results in Figure 6-14, the reproductions are following the target motions with a small *MSE (8.02*, *3.44*, and *8.02 in degree)*. The small *R* value (0.54) in the *pitch* angle indicates that the shape of the reproductions has shifted from the target. The reason of this contradictory result is that the *pitch* variations in the hand motions are small.

Table 6-7 Skills encoding and reproduction results for the beater winding process.

			Yaw (degree)		Pitch (degree)		Roll (degree)	
	Kv	Kp	MSE(degree)	R	MSE(degree)	R	MSE(degree)	R
Subject A	**1.5**	**0.1**	**1.14**	**0.750**	**0.57**	**0.770**	**0.86**	**0.870**
Subject B	**1.0**	**0.03**	**17.19**	**0.870**	**8.59**	**0.653**	**11.46**	**0.697**
Subject C	**1.5**	**0.08**	**2.86**	**0.940**	**4.01**	**0.75**	**5.73**	**0.80**
Subject D	**1.7**	**0.1**	**8.02**	**0.772**	**3.44**	**0.54**	**8.02**	**0.72**

In conclusion, the learned models from the different human demonstrations show the capability to generate state-action policies to reproduce the compliant motions. But the reproduction performances of the individual axis of motions do have variations. The better axis of reproductions relates to good demonstration samples which generalises the unseen motions. For instance, if the state-action policies built from the demonstrations generalise across the training and actual winding processes like subject A, the reproductions will be better. One source of uncertainties during reproduction is the error of the sEMG-force model. Since tension is considered as an important variable in the input signals, the accuracy of the tension estimations will degrade the generalisation capability of the training model. Another source of variations is the inconsistency of the human demonstrations. This is because even though the operator demonstrates a general winding process but he/she cannot guarantee that he/she maintains a similar pattern of the winding motions in each cycle.

These variations were partially solved by HMM; however, some of the unwanted variations are not avoidable during demonstrated trials. This is one of the reasons why human executions are somewhat suboptimal which may lead to suboptimal state-action policies to the learner.

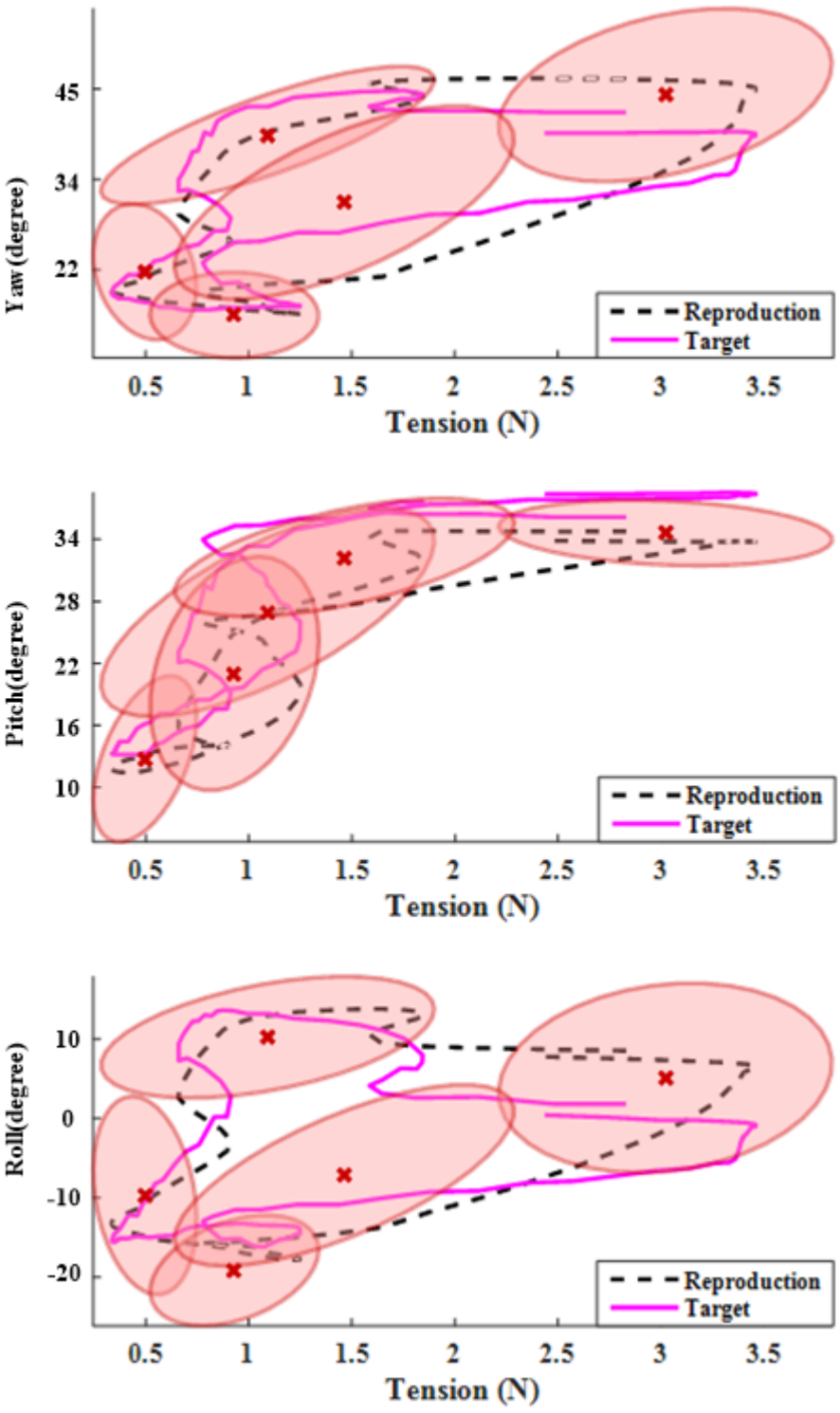


Figure 6-11 Motion reproduction results for subject A

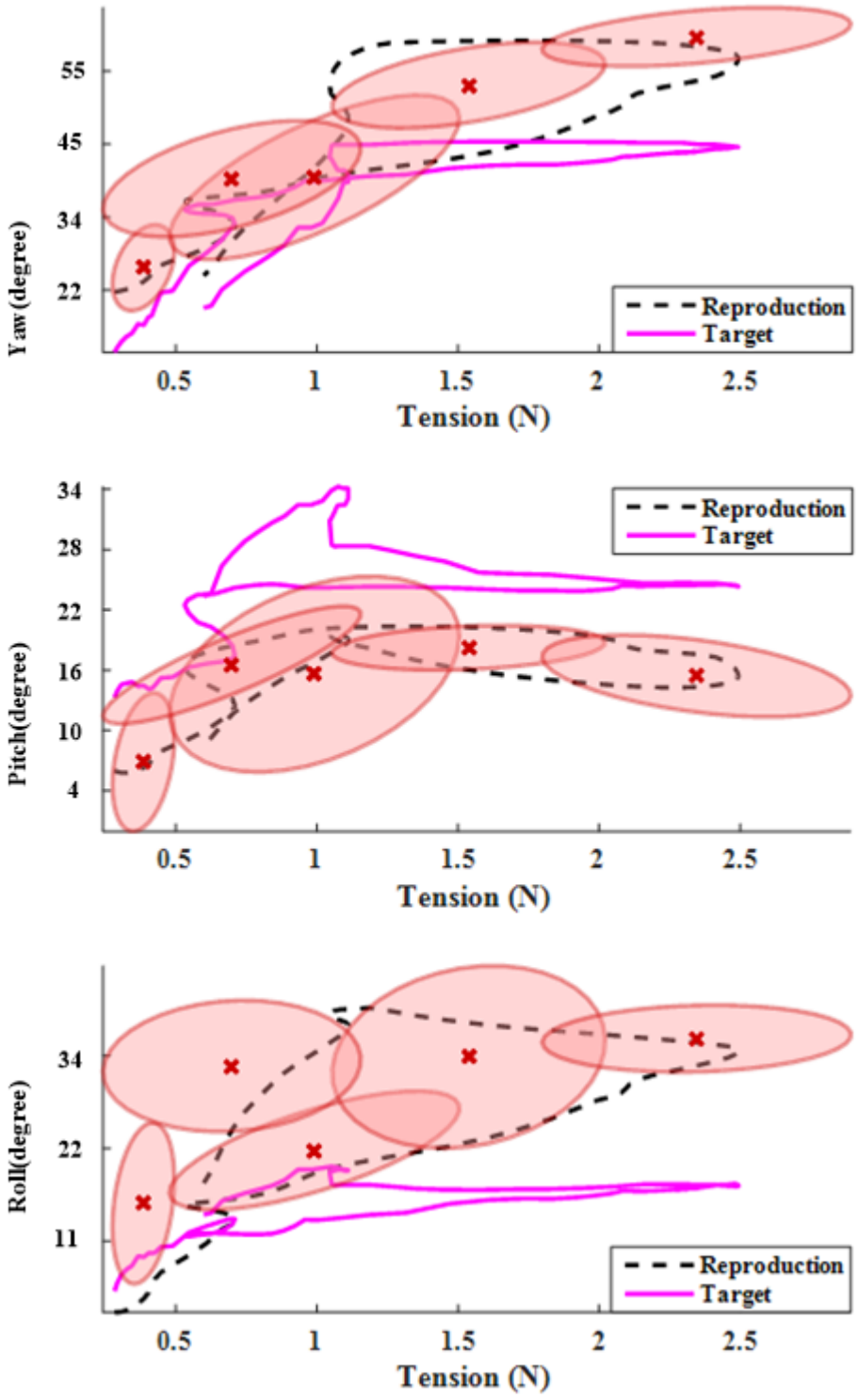


Figure 6-12 Motion reproduction results for subject B.

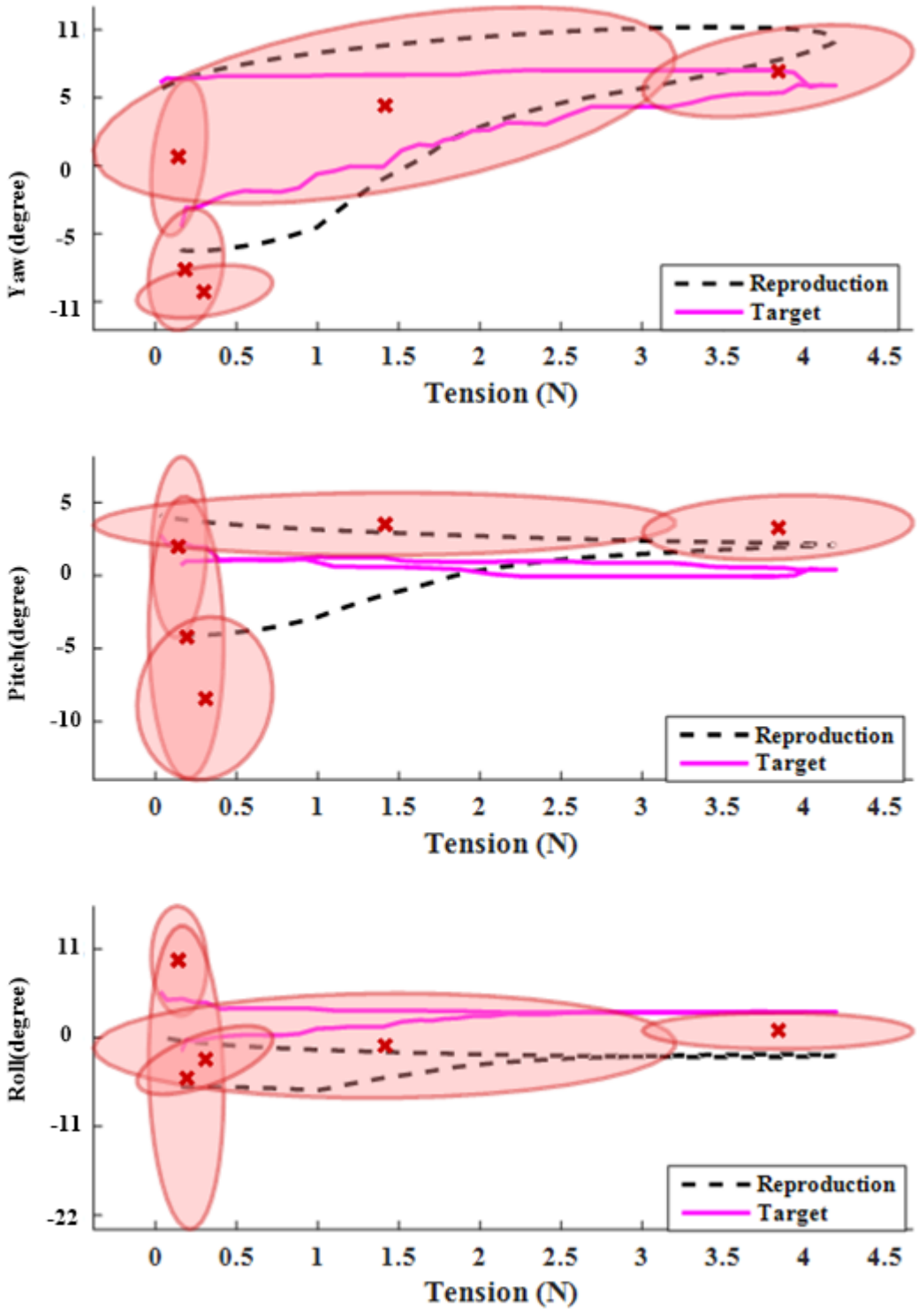


Figure 6-13 Motion reproduction results for subject C.

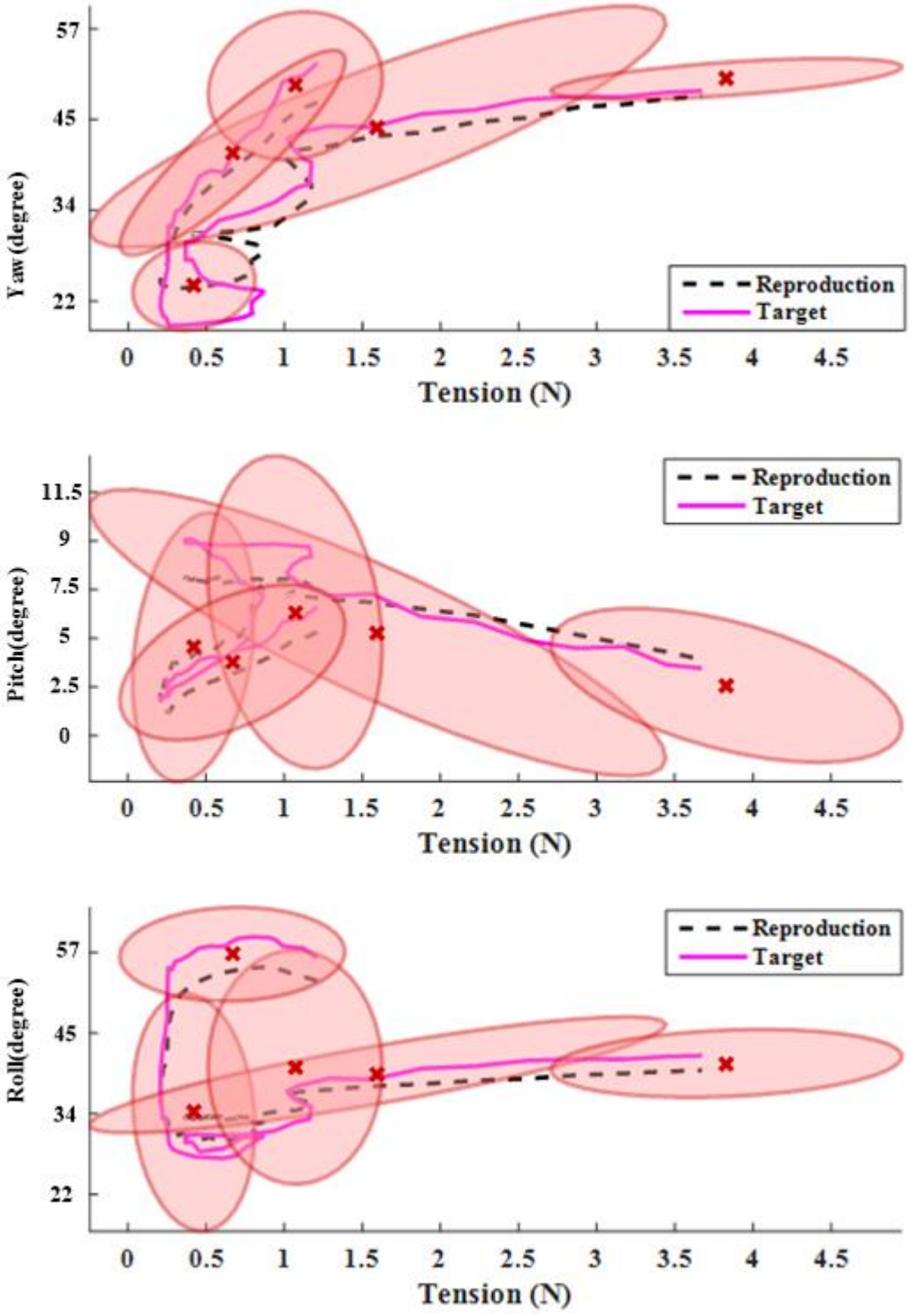


Figure 6-14 Motion reproduction results for subject D.

6.5 Conclusion

In this chapter, the research objective 4 *(to verify and evaluate the robustness of the proposed framework*) has been addressed. The aim is to measure the human motions and haptic feedbacks by using the wearable sensors and then extract and model the skills to allow motion reproductions in the new situations. The methodology for encoding the human skills has been evaluated on the beater winding task. Different from PiH task, the winding task is predominantly episodic cyclic movements. The states by nature come in sequences and are more complex than the PiH in hand motions which the modelling technique has to deal with. Also the primitive sEMG-force model needs specific design of the test rig to imitate the simplified winding process. it would be impossible to directly measure the yarn tension during demonstrations without altering the task executions.

To achieve an indirect modelling of sEMG and tension, first, the optimal parameters of the filtering strategy and TDNN were addressed by running a benchmark. The benchmark was running based on the test rig namely a tension measurement unit. The achieved best results for *MSE* and *R* are 0.04N and 0.955 respectively. The results of evaluations of the actual beater windings after building the muscle-tension model indicate that the built model is capable of estimating the actual winding process. The worst *MSE* is 0.55,N and the best regression coefficient is 0.90. Henceforth, the model enables the operator to demonstrate the whole task without using the dedicated F/T sensor or TMU once the muscle has been calibrated. The skills are then extracted and modeled by using the proposed method in Chapter 4, and the results indicate that a better state-action policy (with *MSE* for *yaw*, *pitch*, and *roll* equal to *1.14*, *0.57* and *0.86 in degree*) is achievable based on the current perception of the tension and the hand poses if the demonstrations generalise across the training and actual winding processes.

Chapter 7. Conclusions and Future Work

7.1 Summary

There is increasing demand for deploying robots in manufacturing industry beyond traditional repetitive high volume production. However, the conventional programming methods that require expert knowledge to solve a specific problem are difficult to reuse in another application and time consuming, which increases the cost of developing automation solutions. In modern applications, robots are required to deliver more complex skills such as compliant behaviours during human-robot interactions and fine motion control in assembly tasks. These skills are currently specific tacit domain knowledge of the human operator. This thesis aimed to reduce the robot programming efforts significantly by developing a methodology to reliably capture, model and transfer the tacit fine/dexterous manufacturing skills from a human demonstrator to the robot. To achieve this research aim four research challenges needed to be overcome.

The first challenge involves reliable tracking of human motion with minimal interference. To overcome the challenge, a hybrid wearable Vicon and IMU system has been developed to overcome the limitations of each system. The proposed system has been demonstrated to be able to compensate for the missing markers information and unsystematic drift in the IMUs. This work was covered in Chapter 3 of this thesis.

The second research challenge is the difficulties in deriving suitable and reusable policies after the skills capturing, encoding and generalisation. The control policy should be generated with a limited number of demonstrations with potentially degraded dataset to suit industrial applications. Thus, the second research objective was to build state-action policy models from human demonstrations that relate to manual industrial manipulations. This work was described in Chapter 4 and Chapter 6 of this thesis, namely for Peg in Hole (PiH) and drum beater winding tasks respectively.

The gross motions can be tracked by using the Vicon-IMU system; however, the force patterns are also important in most manipulations such as the case studies discussed in this thesis. It is often difficult, if not impossible to install a FT sensor in situ, to measure the force profile during demonstrations. This leads to the third research challenge where the haptic feedback were required to be recorded with the minimal interference of the task. The third research challenge was, therefore, to develop a

muscle-force model to predict the forces generated from forearm muscle activations using wearable sEMG devices. The muscle activations provided useful contact force information but were noisy. The signals had to be pre-processed and the effects of various factors that were essential to achieving good model predictions were investigated. This work was described in Chapter 5 and Chapter 6 of this thesis for the PiH and winding tasks respectively. The calibration primitives for muscle-force model were built prior the task demonstrations and validated on the actual PiH and winding tasks. Once the forces were recorded along with the motion trajectories, the skills model could be built.

The fourth research challenge related to the evaluation of the motion reproductions ability based on the learned state-action policy. To simplify the validation process, the motions were reproduced on new testing trials by assuming that the human and the robot systems were physically identical. This work was addressed in Chapter 5 and Chapter 6 of this thesis.

7.2 Contributions to knowledge

In this section, the main contributions from this research are summarised in Table 7-1 (pre-mentioned in Chapter 1.5).

Table 7-1 A summary of the research objectives, novelties, and achievements.

Research objectives and publications	Contributions and novelties	Achievements
To develop a wearable system that reliably tracks human motions. *Y.C. Zhao, Y.M. Goh, N. Lohse, L. Justham, M.R. Jackson, "A Robust Hybrid VICON and IMU System for Tracking Human Forearm Motions," submitted to IEEE Transactions on Sensors, under review.*	• *A quick and automatic data-driven approach to align the IMU and Vicon local frames.* • *A reliable forearm tracking system in free space movements (the actual working volume depends on the number of cameras available) without drifts and occlusion issues by using the proposed Vicon-IMU system.*	• *Systematically evaluated the Vicon system against the CMM, and identified a more accurate working volume (MSE<0.5mm).* • *Compensated the unsystematic error from the IMUs using the Vicon system and achieved unbiased orientation tracking with less than 2˚ accuracy.* • *Overcame the occlusion issue of the photometric based Vicon system using IMU.*
To develop a muscle-force model to predict the forces generated from the forearm	• *Design of test rigs to collect task-specific contact force measurements by using*	• *Model predictions (The worst model with MSE=11.16 N and R=0.979 for the PiH, and*

muscle activations using wearable sEMG devices. *Y.C. Zhao, A. Al-Yacoub, Y. M. Goh, L. Justham, N. Lohse and M.R. Jackson, "Surface EMG-based Force Torque Prediction in a Peg-in-hole Assembly Context for Human Tacit Knowledge Interpretation," submitted to IEEE Transactions on Systems, Man, and Cybernetics, under review.*	*sEMG driven models without installing the bulky F/T sensors during demonstrations.* • *A methodology that allows the operator to perform task demonstrations without interference by wearing sEMG sensors.* • *The design of test rigs to calibrate the sEMG signals to predict the contact and tension forces in the PiH and beater winding operation.*	*MSE=0.55N and R=0.90 for the winding task) were achieved and reported in the primitive calibration tasks. A simple model structure with multiple times delays was essential to build the muscle-force relations.* • *Verified the model performance by controlling the parameters involving the training strategies, the experiment time, subjects, data dimension reductions and the muscle groups.* • *Evaluated the proposed methodology in both PiH and winding tasks. Good force predictions were achieved using the selected primitive models, with some variabilities observed for different subjects and factors.*
To build state-action policy models from human demonstrations that relate to industrial manual manipulations. *Y.C. Zhao, A. Al-Yacoub, Y.M. Goh, L. Justham, N. Lohse, M.R. Jackson, "Human Assembly Skill Capture A Hidden Markov Model Analysis of Force and Torque Data in Peg-in-Hole Assembly," IEEE International Conference on Systems, Man, and Cybernetics (SMC), Budapest, 2016.*	• *Built the skill based GMM models for difficult-to-automate industrial manipulations for the robot to learn from.* • *The enhanced motion reproductions were achieved using encoded skills through simple probabilistic inference. The model is flexible to be reprogrammed if the skill changes.*	• *The probabilistic models have been built for both PiH and winding tasks. The strategies in the skills were encoded by the state transition and probability distribution models.* • *The generalised trajectories have been learned and generated across all the demonstrations with variations for all the subjects.* • *The force based motion reproductions were controlled by a PD controller and evaluated against new demonstration trials.* • *A good motion reproductions performance was achieved for the testing trials within the*

		demonstrated behaviours.
To verify and evaluate the robustness of the proposed framework. *Y.C. Zhao, T. Johnson, Y.M. Goh, L. Justham, N. Lohse, M.R. Jackson," A Sensor Design and Data Analysis for Automatic Drum Beater Winding (2014)", International Conference on Engineering Design (ICED), 2014.*	• *The framework is repeatable and transferrable to a different application.*	• *Both the sEMG-force modeling and the human skills encoding and reproductions methods were evaluated for the PiH and winding tasks.* • *The model was verified against a number of selection criteria and proven transferrable to different applications, with identification and evaluation of the suitable primitive motions to train the models.*

There are limitations in each part of the work to achieve the research objectives. Firstly, the Vicon-IMU system is promising in tracking free body movements without drifts and occlusions, but due to the limited number of cameras, the operator may be invisible from the vision system for a long time. This causes a problem since the IMU drifts cannot be compensated in time. Therefore, if long-term usage is of concern, the free space movements are actually limited in the working volume specified by the number of cameras.

Secondly, to mitigate the interference of the bulky F/T sensor during the task demonstrations, the calibration primitives were designed to simplify the sEMG-force model. However, the choice of this test rig needs to reflect the underlying skills of the actual manipulations. Therefore, the model is restricted by the specific task and careful design of the test rig is required.

Thirdly, the state-action policies derived from the human demonstrations reflect the individual skill's model, but the choice of a good trial is currently based on heuristics. Similarly, the generalisation of control policy by considering different subjects is currently achieved by adding another encoding layer based on the mean values from the Gaussian mixture regressions. However, this method is only well suited for the demonstrations without too many variations. In reality, this is not always the case since the different operator may demonstrate the skills differently. A rationalisation step could be introduced to assess which models are more

appropriate/optimal by defining the objective function. The case studies discussed in this work contain relatively simple trajectories, which reduced the model complexity.

Fourthly, the evaluations of the proposed methods for both case studies assumed the test data were generated from the real robot; however, the robot will have different kinematic embodiment. The assumption used in this work was because the focus of the thesis is to understand the human skills rather than implementation on the physical robot.

7.3 Future work

Much has been accomplished in this very challenging area of skills encoding and transferring to reduce the robot programming efforts, and some work remains to drive this topic forward. This section presents a number of possible topics arising from the research in this thesis.

7.3.1 Evaluations on the Vicon and IMU system

The Vicon system was evaluated using a CMM and the working volume had been identified from the results, however, a more systematic approach can be applied by plotting the error standard deviations along the diagonal distance. Thus the evaluation can be extended to include an error map [47], which is more informative than the current approach used for identification of the working volumes. Also the effects of lighting and different marker configurations need further discussions.

The proposed algorithm for the local frame calibration of the IMU and Vicon systems can be extended to other applications which is compatible to solve the $AX=XB$ problem such as the robot hand-eye calibration. One of the interesting topics is to estimate the transformation from the marker object frame to the robot end effector frame. This transformation is usually unknown if the CAD model is unavailable or the object tool has been reattached to the end effector. Furthermore, since the robot and the Vicon systems establish a close chain, the transformation ${}^{R}H_{V}$ (global Vicon frame in the robot base frame) can be determined accordingly. Due to the measurement and estimation uncertainties from both systems, an error map can then be plotted by exploring the full working volume. The values in the sub-map can be calculated using the *MSE* by comparing with the whole map. Another topic is to implement the anatomical alignment [235] by using the proposed method. This is particularly useful when the wearable sensors are not pre-aligned with the body segment frames. Several

standard human body gestures such as T pose are required to be performed by the human. Furthermore, the impact of the occlusion time for the proposed hybrid approach needs further discussions.

7.3.2 Generalising the sEMG-force model from multiple primitives

The model built in this work was from individual calibration tests (for dominant motion identified in the manual task). Different task related primitives can be performed by the human operator to better represent the actual task execution. The primitive with the best model prediction accuracy was selected to predict forces in the actual task. However, this method does not include the rest of the models, which maybe correlated with the task to some extent. Therefore, a potential way to improve these models is to generate weights to produce better predictive models. By doing this in each prediction time step, all the models will generate their current belief of the actual values of the contact force. These forces multiplied by the weighting parameters will produce the generalised force.

7.3.3 Learning human impedance behaviour using sEMG

When the skills are captured and learned from the demonstrations, the impedance behaviour of the human bodies can also be learned using the same data samples. One of the interesting topics is to teach the robot the compliant behaviour when human is collaborating with another human. A simple task is to lift some load from the work floor and deliver it to a target position [236]. Even though the task is simple, the impedance behaviour is difficult to transfer to the robot to perform at the same level. Part of the behaviour has already encoded in the demonstrations, but extra efforts can be done by designing spring-damper like controller. The sEMG signals can be directly used to map the desired end effector path alongside with the F/T measurements. They can also be indirectly used to indicate the feeling of the human operator. This is important since once the robot is operating online with the human, it needs to know whether the human is feeling comfortable with the current control policy or not. Then the robot can improve its performance by adjusting the control parameters by using the sEMG signals. To achieve this, a sEMG driven reward function needs to be designed. One candidate metric is to use the normalised intensity of all the muscle groups. This feature is invariant to the sensor placement but may not be invariant to different operation time.

Another research topic is to learn short-term impedance behaviour during manipulation and transfer it to the robot. The idea behind this is that the human body segments are not rigid and therefore are suitable for the task with compliance. As a starting point, a ball hitting task can be learned by building the control policy ($sEMG \rightarrow pose$) based on the model free approach [237] and transfer it to robot controller ($motor \rightarrow pose$).

7.3.4 Generalise the constraints from different demonstrations

One of the difficulties of deriving the control policy is the generalisation across multiple demonstrations. The demonstrations can be considered as skill sets for robots to learn a generalised control policy from the task specific constraints [196][238]. The approach proposed in this work was to use the mean values from the GMR to represent the generalised control policy for one subject. Another layer of encoding can be added if generalisations from the different subjects are required. This method is promising if the sample trajectories are close to each other which is the case in this work, when the subjects have been carefully selected and trained. But if the behaviour is too different, further research is required. One promising direction is to learn the potential based value functions, which represent the multiple objectives in a potential map with attractors and repellors. The learned models from the GMM-HMM-based approach can be used to generate such map and guide the derivations of the control policies.

Apart from the task related constraints, the physical constraints in the robot joint space needs to be considered. This is because human is demonstrating in the task space, but the robot has kinematic constraints encoded in the Jacobian matrix. Therefore, the reproduction needs extra care in sending commands in the robot joint space [204]. A pseudoinverse Jacobian method with optimization in the null space [205] is used to follow a desired path in Cartesian task space while keeping the motion in joint space as close as possible to the demonstrated trajectories. This method is useful for physically implementation of the learned model on the robot, which will be discussed next.

7.3.5 Implementation on the physical robots

This is a natural extension of this work. The skills have been well defined and captured from the task domain but in human embodiment. To evaluate the performance of the learned policy on the real robot is attractive from both understanding the human skills and extending the robot capabilities point of views. Before executions,

preparations are required. Firstly, the workspace has to be well defined. The transformation from the global Vicon frame to the robot base frame is important since the human is operating in the Vicon frames but the robot is required to reproduce the trajectories in the robot base frame. After all the frames have been fully defined, the robot can start testing the learned policy. It is interesting to test the skill model from the different subjects on the robot to evaluate its performance; also, it is appealing to test the generalised model across all subjects as well.

7.3.6 Continuous learning with reinforcement learning

Reinforcement learning (RL) is a promising approach to learn complex tasks. Although it is not feasible to run large amount of simulations (such as playing the Atari games [239] or playing the game of the Go [240]) in the robotic applications, RL has been extensively used in the various applications [241]. One critical requirement is defining the exploration strategies for the robot. This is important since learning from simple trial-and-error from scratch is time-consuming. Also, random exploration is often not preferred in the real implementations for safety and space constraints considerations. These are undesired for the manufacturing applications. However, by learning and defining some task constraints using the human demonstrations will solve this problem. On one hand, this can help to reduce the learning time by defining a smaller exploration area. On the other hand, the robot behaviour is predictable during the learning period. Therefore, the results from this work can be used as an initialisation program for a robot to execute continuous learning policy in a more complex environment.

7.3.7 Learning contextual knowledge from a sequence of tasks

This is an important extension of this work since the focus was to learn one task at a time. In reality, the robot is required to execute a sequence of tasks. For instance, it may pick up a workpiece from the table, assemble it to another workpiece and move to the next work piece. The decision on which task should be taken can be manually coded by the programmer if the sequence is mostly procedural. However, if the sequence needs soft decisions and the order of the executions may change dynamically, the task level control is desired. The RL based approach is also promising for learning such contextual knowledge [242], therefore this is another direction of this work when all the skills for the specific tasks have been learned.

References

[1] C. S. G. Lee, “Robot Arm Kinematics, Dynamics, and Control,” *Computer (Long. Beach. Calif).*, vol. 15, no. 12, pp. 62–80, 1982.

[2] G. Michalos, S. Makris, N. Papakostas, D. Mourtzis, and G. Chryssolouris, “Automotive assembly technologies review: challenges and outlook for a flexible and adaptive approach,” *CIRP J. Manuf. Sci. Technol.*, vol. 2, no. 2, pp. 81–91, 2010.

[3] T. Ito and K. Morishige, “Automation of Polishing Process by Industrial Robots with Judgment of Polished Surface Quality based on Imaging Processing,” vol. 2, no. August, pp. 483–487, 2008.

[4] T. Brogårdh, “Present and future robot control development-An industrial perspective,” *Annu. Rev. Control*, vol. 31, no. 1, pp. 69–79, Jan. 2007.

[5] EEF, “EEF skills report 2016: An Up-skill Battle,” 2016.

[6] A. Albu-Schäffer, S. Haddadin, C. Ott, A. Stemmer, T. Wimböck, and G. Hirzinger, “The DLR lightweight robot: design and control concepts for robots in human environments,” *Ind. Robot An Int. J.*, vol. 34, no. 5, pp. 376–385, 2007.

[7] A. Jeremy, E. R. Elena, B. Brian, V. Wyk, and L. J. L. Jean, “Tools for Robotics in SME Workcells: Challenges and Approaches for Calibration and Registration,” *NIST U.S. Dep. Commer.*, 2015.

[8] J. Rasmussen, “Skills, Rules, and Knowledge; Signals, Signs, and Symbols, and Other Distinctions in Human Performance Models,” *IEEE Trans. Syst. Man Cybern.*, vol. SMC-13, no. 3, pp. 257–266, 1983.

[9] C. G. Atkeson and S. Schaal, “Robot learning from demonstration,” in *ICML*, 1997, vol. 97, pp. 12–20.

[10] Y. Yang *et al.*, “Fast programming of Peg-in-hole Actions by human demonstration,” *Proc. - 2014 Int. Conf. Mechatronics Control. ICMC 2014*, no. Icmc, pp. 990–995, 2015.

[11] S. Ekvall and D. Kragic, “Robot learning from demonstration: A task-level planning approach,” *Int. J. Adv. Robot. Syst.*, vol. 5, no. 3, pp. 223–234, 2008.

[12] B. Soediono, “Robot Learning From Demonstration of Force-Based Manipulation Tasks,” *J. Chem. Inf. Model.*, vol. 53, no. May, p. 160, 1989.

[13] P. Bonato, “Wearable sensors and systems,” *IEEE Eng. Med. Biol. Mag.*, vol. 29, no. 3, pp. 25–36, 2010.

[14] S. K. Kim, S. Hong, and D. Kim, “A walking motion imitation framework of a humanoid robot by human walking recognition from IMU motion data,” *9th IEEE-RAS Int. Conf. Humanoid Robot. HUMANOIDS09*, pp. 343–348, 2009.

[15] D. Hutchison and J. C. Mitchell, *Lecture Notes in Computer Science*. 2001.

[16] D. Sueaseenak, T. Chanwimalueang, C. Pintavirooj, and M. Sangworasil, “An accurate forearm EMG signal classification method using two-channel electrode,” *IEEJ Trans. Electr. Electron. Eng.*, vol. 8, no. 4, pp. 328–338, 2013.

[17] B. Karlık, “Machine Learning Algorithms for Characterization of EMG Signals,” *Int. J. Inf. Electron. Eng.*, vol. 4, no. 3, pp. 189–194, 2014.

[18] R. D. Crowninshield and R. A. Brand, “A physiologically based criterion of muscle force prediction in locomotion,” *J. Biomech.*, vol. 14, no. 11, pp. 793–801, 1981.

[19] Y. Geng, P. Zhou, and G. Li, “Toward attenuating the impact of arm positions on electromyography pattern-recognition based motion classification in transradial amputees,” *J. Neuroeng. Rehabil.*, vol. 9, no. 1, p. 74, 2012.

[20] J. Ma, N. V. Thakor, and F. Matsuno, “Hand and wrist movement control of myoelectric prosthesis based on synergy,” *IEEE Trans. Human-Machine Syst.*, vol. 45, no. 1, pp. 74–83, 2015.

[21] L. Sigal, A. O. Balan, and M. J. Black, “HUMAN EVA : Synchronized Video and Motion Capture Dataset Human Motion,” *Int. J. Comput. Vis.*, pp. 4–27, 2010.

[22] M. Sartori, M. Reggiani, D. Farina, and D. G. Lloyd, “EMG-Driven Forward-Dynamic Estimation of Muscle Force and Joint Moment about Multiple Degrees of Freedom in the Human Lower Extremity,” *PLoS One*, vol. 7, no. 12, 2012.

[23] K. R. S. Holzbaur, W. M. Murray, and S. L. Delp, “A model of the upper extremity for simulating musculoskeletal surgery and analyzing neuromuscular control,” *Ann. Biomed. Eng.*, vol. 33, no. 6, pp. 829–840, 2005.

[24] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, “A survey of robot learning from demonstration,” *Rob. Auton. Syst.*, vol. 57, no. 5, pp. 469–483, 2009.

[25] C. L. Nehaniv and K. Dautenhahn, “The Correspondence Problem in Social Learning: What Does it Mean for Behaviors to ‘Match’ Anyway?,” *Perspect. Imitation From Cogn. Neurosci.*, pp. 1–10, 2002.

[26] M. Kaiser, M. Kaiser, H. Friedrich, and R. Dillmann, “Obtaining Good Performance From A Bad Teacher,” *Int. Conf. Mach. Learn. Work. Program. By Demonstr. Tahoe*, 1995.

[27] N. Delson and H. West, “Robot programming by human demonstration: adaptation and inconsistency in constrained motion,” *Int. Conf. Robot. Autom.*, no. April, pp. 30–36, 1996.

[28] P. Pook and D. Ballard, “Recognizing teleoperated manipulations,” *Int. Conf. Robot. Autom.*, pp. 578–585, 1993.

[29] K. Nyomen, M. Romarheim Haugen, and A. R. Jensenius, “MuMYO — Evaluating and Exploring the MYO Armband for Musical Interaction,” *Proc. Int. Conf. New Interfaces Music. Expr.*, pp. 215–218, 2015.

[30] S. L. Delp, P. Loan, M. G. Hoy, F. E. Zajac, E. L. Topp, and J. M. Rosen, “An interactive graphics based model of the lower extremity to study orthopaedic surgical procedures,” *IEEE Trans. Biomed. Eng.*, vol. 37, no. 8, pp. 757–767, 1990.

[31] K. Mountjoy, E. Morin, and K. Hashtrudi-Zaad, “Contraction-based variations in upper limb EMG-force models under isometric conditions,” *Proc. 31st Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. Eng. Futur. Biomed. EMBC 2009*, pp. 2955–2959, 2009.

[32] L. W. O’Sullivan and T. J. Gallwey, “Upper-limb surface electro-myography at maximum supination and pronation torques: The effect of elbow and forearm angle,” *J. Electromyogr. Kinesiol.*, vol. 12, no. 4, pp. 275–285, 2002.

[33] K. Kiguchi and Y. Hayashi, “An EMG-based control for an upper-limb power-assist exoskeleton robot,” *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 42, no. 4, pp. 1064–1071, 2012.

[34] J. Hashemi, E. Morin, P. Mousavi, and K. Hashtrudi-Zaad, “Surface EMG force modeling with joint angle based calibration,” *J. Electromyogr. Kinesiol.*, vol. 23, no. 2, pp. 416–424, 2013.

[35] S. Calinon and A. G. Billard, “What is the teacher’s role in robot programming by demonstration?: Toward benchmarks for improved learning,” *Interact. Stud.*, vol. 8, no. 3, pp. 441–464, Jan. 2007.

[36] S. K. Yun, “Compliant manipulation for peg-in-hole: Is passive compliance a key to learn contact motion?,” in *Proceedings - IEEE International Conference on Robotics and Automation*, 2008, pp. 1647–1652.

[37] F. J. Abu-Dakka, B. Nemec, A. Kramberger, A. G. Buch, N. Krüger, and A. Ude, “Solving peg-in-hole tasks by human demonstration and exception strategies,” *Ind. Robot An Int. J.*, vol. 41, no. 6, pp. 575–584, 2014.

[38] T. Johnson, “Percussion plus Human Factors report,” 2014.

[39] L. N. Smith and M. L. Smith, “Automatic machine vision calibration using statistical and neural network methods,” *Image Vis. Comput.*, vol. 23, no. 10, pp. 887–899, Sep. 2005.

[40] a. Agrawal, J. Barnwell, and R. Raskar, “Vision-guided Robot System for Picking Objects by Casting Shadows,” *Int. J. Rob. Res.*, vol. 29, no. 2–3, pp. 155–173, Nov. 2009.

[41] Y. Zhang, H. Hu, and H. Zhou, “Study on adaptive kalman filter algorithms in human movement tracking,” in *IEEE International Conference on Information Acquisition*, 2005.

[42] R. Poppe, “Vision-based human motion analysis: An overview,” *Comput. Vis. Image Underst.*, vol. 108, no. 1–2, pp. 4–18, 2007.

[43] Qualisys, “Qualisys.” [Online]. Available: http://www.qualisys.se/.

[44] J. Lee, J. Chai, P. S. a. Reitsma, J. K. Hodgins, and N. Pollard, “Interactive control of avatars animated with human motion data,” *ACM Trans. Graph.*, vol. 21, no. 3, pp. 491–500, 2002.

[45] Charndyn, “Charndyn.” [Online]. Available: http://www.charndyn.com/.

[46] Ndigital, “Ndigital.” [Online]. Available: http://www.ndigital.com/.

[47] R. Summan *et al.*, “Spatial calibration of large volume photogrammetry based metrology systems,” *Meas. J. Int. Meas. Confed.*, vol. 68, pp. 189–200, 2015.

[48] P.-F. Yang, M. Sanno, G.-P. Bruggemann, and J. Rittweger, “Evaluation of the performance of a motion capture system for small displacement recording and a discussion for its application potential in bone deformation in vivo measurements,” in *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, 2012, vol. 226, no. 11, pp. 838–847.

[49] H. Liu, C. Holt, and S. Evans, “Accuracy and repeatability of an optical motion analysis system for measuring small deformations of biological tissues,” *Biomech*, pp. 210–214, 2007.

[50] M. Windolf, N. Gotzen, and M. Morlock, “Systematic accuracy and precision analysis of video motion capturing systems – exemplified on the Vicon-460 system,” *Biomech*, pp. 2776– 2780, 2008.

[51] A. Cappozzo, “Gait Analysis Methodology,” *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2013.

[52] C. Yang, A. Kerr, V. Stankovic, L. Stankovic, P. Rowe, and S. Cheng, “Human Upper Limb Motion Analysis for Post-Stroke Impairment Assessment Using Video Analytics,” *IEEE Access*, vol. 4, pp. 650–659, 2016.

[53] C. Bregler, “Motion Capture Technology for Entertainment,” *Spotlight Mag.*, pp. 158–160, 2007.

[54] D. M. Kwartowitz, M. I. Miga, S. D. Herrell, and R. L. Galloway, “Towards image guided robotic surgery: Multi-arm tracking through hybrid localization,” *Int. J. Comput. Assist. Radiol. Surg.*, vol. 4, no. 3, pp. 281–286, 2009.

[55] R. Schmidt, C. Disselhorst-Klug, J. Silny, and G. Rau, “A marker-based measurement procedure for unconstrained wrist and elbow motions,” *J. Biomech.*, vol. 32, no. 6, pp. 615–621, 1999.

[56] M. Damsgaard, J. Rasmussen, S. T. Christensen, E. Surma, and M. de Zee, “Analysis of musculoskeletal systems in the AnyBody Modeling System,” *Simul. Model. Pract. Theory*, vol. 14, no. 8, pp. 1100–1111, 2006.

[57] D. K. Bhatnagar, “Position trackers for Head Mounted Display systems: A survey,” *Univ. North Carolina, Chapel Hill TR93-010*, pp. 1–22, 1993.

[58] C. Wren, A. Azarbayjani, T. Darrell, and A. Pentland, “Pfinder: Real-Time Tracking of the Human Body,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 780–785, 1997.

[59] A. M. Baumberg and D. C. Hogg, “An efficient method for contour tracking using active shape models,” *Proc. 1994 IEEE Work. Motion Non-rigid Articul. Objects*, pp. 194–199, 1994.

[60] E. Huber and A. Group, “3-D Real-Time Gesture Recognition Using Proximity Spaces,” *3rd IEEE Work. Appl. Comput. Vision, 196. WACV ’96*, pp. 136–141, 1996.

[61] D. Marr and H. K. Nishihara, “Representation and recognition of the spatial organization of three dimensional shapes,” *Proc. R. Soc. London B Biol. Sci.*, no. October, pp. 26–28, 1978.

[62] J. Davis, “Hierarchical Motion Hidtory Images for Recognizing Human Motion,” in *IEEE Workshop on Detection and Recognition of Events in Video*, 2001, pp. 39–46.

[63] R. Polana and R. Nelson, “Low Level Recognition of Human Motion,” *Proc. IEEE Work. Motion Non-Rigid Articul. Objects*, pp. 77–82, 1994.

[64] B. Coifman, D. Beymer, P. McLauchlan, and J. Malik, “A real-time computer vision system for vehicle tracking and traffic surveillance,” *Transp. Res. Part C Emerg. Technol.*, vol. 6, no. 4, pp. 271–288, 1998.

[65] L. Ren, G. Shakhnarovich, J. K. Hodgins, H. Pfister, and P. Viola, “Learning silhouette features for control of human motion,” *ACM Trans. Graph.*, vol. 24, no. 4, pp. 1303–1331, 2005.

[66] L. Zhang, J. Sturm, D. Cremers, and D. Lee, “Real-time human motion tracking using multiple depth cameras,” *Intell. Robot. Syst. (IROS), 2012 IEEE/RSJ Int. Conf.*, pp. 2389–2395, 2012.

[67] R. Shotton, J Fitzgibbon, A Cook, M Sharp, T Finocchio, M Moore, “Real- Time Human Pose Recognition in Parts from Single Depth Images. In: Computer Vision and Pattern Recognition,” in *(CVPR), 2011 IEEE Conference; 20–25 June*, 2011, pp. 1297–1304.

[68] V. Ganaphathi, C. Plagemann, D. Koller, and S. Thrun, “Real time motion capture using a single time-of-flight camera,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 755–762.

[69] C. Sminchisescu and B. Triggs, “Covariance-scaled sampling for Monocular 3D body tracking,” in *IEEE International Conference on Computer Vision and Pattern Recognition*, 2001, vol. 1, pp. 447–454.

[70] A. Agarwal and B. Triggs, “Recovering 3D human pose from monocular images,” in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2006, vol. 28, no. 1, pp. 44–58.

[71] Y. Uno *et al.*, “Formation and control of optimal trajectory in human multijoint arm movement,” *Biol. Cybern.*, vol. 61, no. 2, pp. 89–101, 1989.

[72] H. Zhou and H. Hu, “Human motion tracking for rehabilitation-A survey,” *Biomed. Signal Process. Control*, vol. 3, no. 1, pp. 1–18, 2008.

[73] D. Roetenberg, H. Luinge, and P. Slycke, “Xsens MVN : Full 6DOF Human Motion Tracking Using Miniature Inertial Sensors,” *Tech. Rep.*, pp. 1–7, 2013.

[74] H. Zhou and H. Hu, “Inertial motion tracking of human arm movements in stroke rehabilitation,” *IEEE Int. Conf. Mechatronics Autom.*, pp. 1306–1311, 2005.

[75] H. Zhou, H. Hu, and N. Harris, “Wearable inertial sensors for arm motion tracking in home-based rehabilitation,” *Proc. 9th Int. Conf. Intell. Auton. Syst.*, pp. 930–937, 2006.

[76] H. Zhou, H. Hu, and A. Motivation, “Kinematic model aided inertial motion tracking of human upper limb,” *Inf. Acquis. 2005 IEEE Int. Conf.*, no. January, pp. 150–155, 2005.

[77] G. Yu *et al.*, “Potential of IMU sensors in performance analysis of professional alpine skiers,” *Sensors (Switzerland)*, vol. 16, no. 4, pp. 1–21, 2016.

[78] S. Sessa, M. Zecca, Z. H. Lin, L. Bartolomeo, H. Ishii, and A. Takanishi, “A Methodology for the Performance Evaluation of Inertial Measurement Units,” *J. Intell. Robot. Syst.*, vol. 71, no. 2, pp. 143–157, 2013.

[79] D. Kang, Y. Jung, A. Park, and J. Kim, “Human Body Motion Capture System using Magnetic and Inertial Sensor Modules,” no. October, pp. 1–4, 2011.

[80] S. Alavi, D. Arsenault, and A. Whitehead, “Quaternion-Based Gesture Recognition Using Wireless Wearable Motion Capture Sensors,” *Sensors*, vol. 16, no. 5, p. 605, 2016.

[81] S. Qiu, Z. Wang, H. Zhao, and H. Hu, “Using Distributed Wearable Sensors to Measure and Evaluate Human Lower Limb Motions,” *IEEE Trans. Instrum. Meas.*, vol. 65, no. 4, pp. 939–950, 2016.

[82] Ascension Tech, “Nest of birds.” [Online]. Available: http://www.ascensiontech.com/products/motionstar.pdf.

[83] Polhemus, “Polhemus.” [Online]. Available: http://www.polhemus.com/.

[84] T. Molet, R. Boulic, and D. Thalmann, “A real time anatomical converter for human motion capture,” *Eurographics Work. Comput. Animat. Simul.*, vol. 96, no. September, pp. 79–94, 1996.

[85] Vrealities, “vrealities.” [Online]. Available: http://www.vrealities.com/cyber.html.

[86] A. Pascual-Leone, “Design considerations for a wearable monitor to measure finger posture,” *J. Neuroeng. Rehabil.*, vol. 2, p. 5, 2005.

[87] A. Taylor, “Design of an exoskeletal arm for use in long term stroke rehabilitation,” *Univ. Bath*, 1997.

[88] R. S. Mosher, “Industrial Manipulators,” *Sci. Am.*, vol. 211, no. 4, pp. 88–96, 1964.

[89] W. R. Corliss and E. G. Johnson, *Teleoperator Controls*. Washington DC, 1968.

[90] K. B. Wilson, “Servoarm - A Water Hydraulic Master-Slave,” in *23rd Conference on Remote Systems Technology*, 1975, pp. 233–240.

[91] C. Loconsole, S. Dettori, A. Frisoli, C. A. Avizzano, and M. Bergamasco, “An EMG-based approach for on-line predicted torque control in robotic-assisted rehabilitation,” *IEEE Haptics Symp. HAPTICS*, pp. 181–186, 2014.

[92] G. Ligorio and A. M. Sabatini, “A Novel Kalman Filter for Human Motion Tracking With an Inertial-Based Dynamic Inclinometer.,” *IEEE Trans. Biomed. Eng.*, vol. 62, no. 8, pp. 2033–43, 2015.

[93] K. Dautenhahn and I. Werry, “Issues of Robot-Human Interaction Dynamics in the Rehabilitation of Children with Autism,” *Soc. Dyn.*, vol. 6, pp. 519–528, 2000.

[94] X. Pennec, P. Cachier, and N. Ayache, “Tracking brain deformations in time sequences of 3D US images,” *Pattern Recognit. Lett.*, vol. 24, no. 4–5, pp. 801–813, 2003.

[95] A. Roche, X. Pennec, G. Malandain, and N. Ayache, “Rigid registration of 3-D ultrasound with MR images: a new approach combining intensity and gradient information.,” *IEEE Trans. Med. Imaging*, vol. 20, no. 10, pp. 1038–1049, 2001.

[96] V. Technologies, “CyberTouch,” Company brochure, Palo Alto, CA, 1998.

[97] J. G. Andrews and Y. Youm, “A biomechanical investigation of wrist kinematics,” *Biomechanics*, vol. 12, pp. 83–89, 1979.

[98] D. Lee and Y. Nakamura, “Motion Capturing from Monocular Vision by Statistical Inference Based on Motion Database : Vector Field Approach,” in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2007, pp. 617–623.

[99] D. M. Gavrila and L. S. Davis, “3-D model-based tracking of humans in action: a multi-view approach,” in *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1996, pp. 73–80.

[100] R. Kehl and L. Van Gool, “Markerless Tracking of Complex Human Motions from Multiple Views.,” *Comput. Vis. Image Underst.*, vol. 104, no. 2, pp. 190–209, 2006.

[101] R. H. Chowdhury, M. B. I. Reaz, M. A. B. M. Ali, A. A. A. Bakar, K. Chellappan, and T. G. Chang, “Surface electromyography signal processing and classification techniques.,” *Sensors (Basel).*, vol. 13, no. 9, pp. 12431–12466, 2013.

[102] R. Merletti and C. J. De Luca, “New techniques in surface electromyography,” *Comput. Aided EMG Expert Syst.*, pp. 115–124, 1989.

[103] D. Zazula, D. Korosec, and A. Sostaric, “Computer-assisted decomposition of the electromyograms,” in *IEEE Symp. Computer-Based Medical System*, 1998, pp. 26–31.

[104] H. M. Pereira, “Influence of contraction intensity on electromyographic activity and perceived exertion during arm abduction,” in *Proc. Congress of the International Society of Electrophysiology and Kinesiology (ISEK)*, 2010.

[105] C. J. De Luca, C. De Luca, and C. J. De Luca, “The use of surface electromyography in biomechanics,” *J. Appl. Biomech.*, vol. 13, no. July 1993, pp. 1–38, 1997.

[106] A. Alkan and M. Günay, “Identification of EMG signals using discriminant analysis and SVM classifier,” *Expert Syst. Appl*, vol. 39, pp. 44–47, 2012.

[107] R. H. Chowdhury, M. B. I. Reaz, M. A. B. Ali, A. A. A. Bakar, K. Chellappan, and T. G. Chang, “Surface electromyography signal processing and classification techniques,” *Sensors*, vol. 13, no. 9, pp. 12431–12466, 2013.

[108] E. A. Clancy, E. L. Morin, and R. Merletti, “Sampling, noise-reduction and amplitude estimation issues in surface electromyography,” *J. Electromyogr. Kinesiol.*, vol. 12, pp. 1–16, 2002.

[109] D. P. Burbank and J. G. Webster, “Reducing skin potential motion artifact by skin abrasion,” *Med. Biol. Eng. Comput.*, vol. 16, pp. 31–38, 1978.

[110] M. Malboubi, F. Razzazi, and S. M. Aliyari, “Elimination of Power Line Noise from EMG Signals Using an Efficient Adaptive Laguerre Filter,” in *In Proceedings of the International Conference on Signals and Electronic Systems*, 2010, pp. 49–52.

[111] M. B. I. Raez, M. S. Hussain, F. Mohd-Yasin, M. Reaz, M. S. Hussain, and F. Mohd-Yasin, “Techniques of EMG signal analysis: detection, processing, classification and applications.,” *Biol. Proced. Online*, vol. 8, no. 1, pp. 11–35, 2006.

[112] F. Hug, “Can muscle coordination be precisely studied by surface electromyography?,” *J. Electromyogr. Kinesiol.*, vol. 21, no. 1, pp. 1–12, 2011.

[113] A. L. Ricamato, R. G. Absher, M. T. Moffroid, and J. P. Tranowski, “A Time-Frequency Approach to Evaluate Electromyographic Recordings,” in *Proceedings of the Fifth Annual IEEE Symposium on Computer-Based Medical Systems*, 1992, pp. 520–527.

[114] J. U. Chu, I. Moon, Y. J. Lee, S. K. Kim, and M. S. Mun, “A supervised feature-projection-based real-time EMG pattern recognition for multifunction myoelectric hand control,” *IEEE/ASME Trans. Mechatron.*, vol. 12, pp. 282–290, 2007.

[115] I. Daubechies, “The wavelet transform, time-frequency localization and signal analysis,” *IEEE Trans. Inf. Theory*, vol. 36, pp. 961–1005, 1990.

[116] M. Farge, “Wavelet transforms and their applications to Turbulence,” *Annu. Rev. Fluid Mech.*, vol. 24, pp. 395–457, 1992.

[117] P. Guglielminotti and R. Merletti, “Effect of electrode location on surface myoelectric signal variables: a simulation study,” *9th Int Congr ISEK*, p. 106, 1992.

[118] F. Laterza and G. Olmo, “Analysis of EMG signals by means of the matched wavelet transform,” *Electron. Lett.*, vol. 33, no. 5, p. 357, 1997.

[119] M. Khezri and M. Jahed, “Surface electromyogram signal estimation based on wavelet thresholding technique.,” *Conf. Proc. IEEE Eng. Med. Biol. Soc.*, vol. 2008, pp. 4752–5, 2008.

[120] A. Phinyomark, C. Limsakul, and P. Phukpattaranont, “A comparative study of wavelet denoising for multifunction myoelectric control,” *Proc. - 2009 Int. Conf. Comput. Autom. Eng. ICCAE 2009*, pp. 21–25, 2009.

[121] A. Phinyomark, C. Limsakul, and P. Phukpattaranont, “Optimal wavelet functions in wavelet denoising for multifunction myoelectric control,” *Proc. Electr. Eng. Comput. Telecommun. Inf. Technol.*, pp. 1098–1101, 2009.

[122] C. F. Jiang and S. L. Kuo, “A comparative study of wavelet denoising of surface electromyographic signals,” *Annu. Int. Conf. IEEE Eng. Med. Biol. - Proc.*, pp. 1868–1871, 2007.

[123] D. K. Kumar, N. D. Pah, and A. Bradley, “Wavelet analysis of surface electromyography to determine muscle fatigue,” *IEEE Trans Neural Syst Rehabil Eng*, vol. 11, no. 4, pp. 400–406, 2003.

[124] M. S. Hussain and M. Mamun, “Effectiveness of the Wavelet Transform on the Surface EMG to Understand the Muscle Fatigue During Walk,” *Meas. Sci. Rev.*, vol. 12, no. 1, pp. 28–33, 2012.

[125] K. C. Chua, V. Chandran, U. R. Acharya, and C. M. Lim, “Application of higher order statistics/spectra in biomedical signals-A review,” *Med. Eng. Phys.*, vol. 32, no. 7, pp. 679–689, 2010.

[126] G. B. Giannakis and M. K. Tsatsanis, “HOS or SOS for Parametric Modeling?,” in *In Proceedings of the IEEEthe International Conference on Acoustics, Speech, and Signal Processing*, 1991, pp. 3097–3100.

[127] S. Shahid, “Higher Order Statistics Techniques Applied to EMG Signal Analysis and Characterization,” University of Limerick, 2004.

[128] P. A. Kaplanis, C. S. Pattichis, L. J. Hadjileontiadis, and S. M. Panas, “Bispectral analysis of surface EMG,” *Proc. Mediterr. Electrotech. Conf. - MELECON*, vol. 2, pp. 770–773, 2000.

[129] K. Kanosue, M. Yoshida, K. Akazawa, and K. Fujii, “The number of active motor units and their firing rates in voluntary contraction of human brachialis muscle.,” *Jpn. J. Physiol.*, vol. 29, pp. 427–443, 1979.

[130] S. Karlsson and L. Nystrom, “Real-time system for EMG signal analysis of static and dynamic contractions,” *Med. Biol. Soc.*, pp. 1347–1348, 1995.

[131] N. E. Huang *et al.*, “The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis,” *Proc. R. Soc. Lond.*, vol. A 495, pp. 903–995, 1998.

[132] A. O. Andrade, S. Nasuto, P. Kyberd, C. M. Sweeney-Reed, and F. R. Van Kanijn, “EMG signal filtering based on Empirical Mode Decomposition,” *Biomed. Signal Process. Control*, vol. 1, no. 1, pp. 44–55, 2006.

[133] K.-M. Chang, “Ensemble Empirical Mode Decomposition: a Noise-Assisted Data Analysis Method,” *Adv. Adapt. Data Anal.*, vol. 55, no. 4, pp. 193–201, 2010.

[134] P. Comon, “Independent Component Analysis---a new concept?,” *Signal Processing*, vol. 36, pp. 287–314, 1994.

[135] H. Nakamura, M. Yoshida, M. Kotani, K. Akazawa, and T. Moritani, “The application of independent component analysis to the multi-channel surface electromyographic signals for separation of motor unit action potential trains: Part II - Modelling interpretation,” *J. Electromyogr. Kinesiol.*, vol. 14, no. 4, pp. 433–441, 2004.

[136] D. P. Acharya and G. Panda, “A Review of Independent Component Analysis Techniques and their Applications,” *Iete Tech. Rev.*, vol. 25, no. 6, pp. 320–332, 2009.

[137] G. A. Garcia, R. Okuno, and K. Akazawa, “A decomposition algorithm for surface electrode-array electromyogram. A noninvasive, three-step approach to analyze surface EMG signals,” *IEEE Eng Med Biol Mag*, vol. 24, no. 4, pp. 63–72, 2005.

[138] M. Zardoshti-Kermani, B. C. Wheeler, K. Badie, and R. M. Hashemi, “EMG feature evaluation for movement control of upper extremity prostheses,” *IEEE Trans. Rehabil. Eng.*, vol. 3, no. 4, pp. 324–333, 1995.

[139] B. Hudgins, P. Parker, and R. N. Scott, “A New Strategy for Multifunction Myoelectric Control,” *IEEE Trans. Biomed. Eng.*, vol. 40, no. 1, pp. 82–94, 1993.

[140] P. P. Angkoon Phinyomark and C. Limsakul, “Feature reduction and selection for EMG signal classification,” *Expert Syst. Appl.*, vol. 39, no. 8, pp. 7420–7431, 2012.

[141] C. Disselhorst-Klug, T. Schmitz-Rode, and G. Rau, “Surface electromyography and muscle force: Limits in sEMG-force relationship and new approaches for applications,” *Clin. Biomech.*, vol. 24, no. 3, pp. 225–235, 2009.

[142] K. Englehart, B. Hudgins, P. A. Parker, and M. Stevenson, “Classification of the myoelectric signal using time-frequency based representations,” *Med. Eng. Phys.*, vol. 21, no. 6–7, pp. 431–438, 1999.

[143] L. J. Hargrove, G. Li, K. B. Englehart, and B. S. Hudgins, “Principal components analysis preprocessing for improved classification accuracies in pattern-recognition-based myoelectric control,” *IEEE Trans. Biomed. Eng.*, vol. 56, no. 5, pp. 1407–1414, 2009.

[144] A. Phinyomark, C. Limsakul, and P. Phukpattaranont, “A Novel Feature Extraction for Robust EMG Pattern Recognition,” *J. Comput.*, vol. 1, no. 1, pp. 71–80, 2009.

[145] C. Disselhorst-Klug, T. Schmitz-Rode, and G. Rau, “Surface electromyography and muscle force: Limits in sEMG-force relationship and new approaches for applications,” *Clin. Biomech. (Bristol, Avon)*, vol. 24, no. 3, pp. 225–35, 2008.

[146] D. Staudenmann, K. Roeleveld, D. F. Stegeman, and J. H. Van Dieën, "Methodological aspects of SEMG recordings for force estimation--a tutorial and review.," *J. Electromyogr. Kinesiol.*, vol. 20, no. 3, pp. 375–87, 2010.

[147] H. H. C. M. Savelberg and W. Herzog, "Prediction of dynamic tendon forces from electromyographic signals: An artificial neural network approach," *J. Neurosci. Methods*, vol. 78, no. 1–2, pp. 65–74, 1997.

[148] M. H. Jali, T. A. Izzuddin, Z. H. Bohari, M. F. Sulaima, and H. Sarkawi, "Predicting EMG based elbow joint torque model using multiple input ANN neurons for arm rehabilitation," *Proc. - UKSim-AMSS 16th Int. Conf. Comput. Model. Simulation, UKSim 2014*, pp. 189–194, 2014.

[149] S. Kwon and J. Kim, "Real-time upper limb motion estimation from surface electromyography and joint angular velocities using an artificial neural network for human-machine cooperation," *IEEE Trans. Inf. Technol. Biomed.*, vol. 15, no. 4, pp. 522–530, 2011.

[150] Q. Zhang, R. Hosoda, and G. Venture, "Human joint motion estimation for electromyography (EMG)-based dynamic motion control," *Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS*, vol. 1, no. 2, pp. 21–24, 2013.

[151] M. Sartori, L. Gizzi, D. G. Lloyd, and D. Farina, "A musculoskeletal model of human locomotion driven by a low dimensional set of impulsive excitation primitives.," *Front. Comput. Neurosci.*, vol. 7, no. June, p. 79, 2013.

[152] S. L. Delp, F. C. Anderson, A. S. Arnold, P. Loan, A. Habib, and C. T. John, "OpenSim: open- source software to create and ana- lyze dynamic simulations of move- ment," *IEEE Trans. Biomed. Eng.*, vol. 54, pp. 1940–1950, 2007.

[153] R. Heine, K. Manal, and T. S. Buchanan, "Using Hill-Type Muscle Models and EMG Data in a Forward Dynamic Analysis of Joint Moment," *J. Mech. Med. Biol.*, vol. 3, no. 2, pp. 169–186, 2003.

[154] F. E. Zajac, "Muscle and tendon: properties, models, scaling, and application to biomechanics and motor control.," *Crit. Rev. Biomed. Eng.*, vol. 17, no. 4, pp. 359–411, 1989.

[155] M. Sartori, M. Reggiani, D. Farina, and D. G. Lloyd, “EMG- driven forward-dynamic estimation of muscle force and joint moment about multiple degrees of freedom in the human lower extremity,” *PLoS One*, 2012.

[156] D. G. Lloyd and T. F. Besier, “An EMG-driven musculoskeletal model to estimate muscle forces and knee joint moments in vivo,” *J. Biomech.*, vol. 36, no. 6, pp. 765–776, 2003.

[157] R. Sutton and A. Barto, “Reinforcement Learning: An Introduction,” *IEEE Trans. Neural Netw.*, vol. 9, no. 5, p. 1054, 1998.

[158] A. Billard, S. Callinon, R. Dillmann, and S. Schaal, “Robot programming by demonstration,” in *Handbook of Robotics*, New York, NY, USA: Springer, 2008.

[159] S. Schaal, A. Ijspeert, and A. Billard, “Computational approaches to motor learning by imitation.,” *Philos. Trans. R. Soc. Lond. B. Biol. Sci.*, vol. 358, no. 1431, pp. 537–47, 2003.

[160] I. Benenson, “Imitation in Animals and Artifacts,” *J. Artif. Soc. Soc. Simul.*, vol. 7, no. 4, 2004.

[161] S. Calinon, P. Evrard, E. Gribovskaya, A. Billard, and A. Kheddar, “Learning collaborative manipulation tasks by demonstration using a haptic interface,” *Int. Conf. Adv. Robot.*, pp. 1–6, 2009.

[162] P. Kormushev, S. Calinon, and D. G. Caldwell, “Imitation Learning of Positional and Force Skills Demonstrated via Kinesthetic Teaching and Haptic Input,” *Adv. Robot.*, vol. 25, no. 5, pp. 581–603, 2011.

[163] G. Al-Naymat, S. Chawla, and J. Taheri, “SparseDTW: A Novel Approach to Speed up Dynamic Time Warping,” no. 2007, 2012.

[164] Z. Jakovljevic, P. B. Petrovic, and J. Hodolic, “Contact states recognition in robotic part mating based on support vector machines,” *Int. J. Adv. Manuf. Technol.*, vol. 59, no. 1–4, pp. 377–395, 2012.

[165] C. L. Nehaniv, K. Dautenhahn, K. Dautenhahn, and C. L. Nehaniv, “The Correspondence Problem,” Cambridge, MA, USA: Press, MIT, 2002.

[166] A. Ng *et al.*, “Inverted autonomous helicopter flight via reinforcement learning,” in *International Symposium on Experimental Robotics*, 2004.

[167] P. K. Pook and D. H. Ballard, “Recognizing teleoperated manipulations,” in *IEEE International Conference on Robotics and Automation, ICRA*, 2007.

[168] J. D. Sweeney and R. A. Grupen, “A model of shared grasp affordances from demonstration,” in *IEEE-RAS International Conference on Humanoids Robots*, 2007.

[169] T. Inamura and M. Inaba, “Acquisition of probabilistic behavior decision model based on the interactive teaching method,” in *International Conference on Advanced Robotics*, 1999.

[170] W. D. Smart, “Making Reinforcement Learning Work on Real Robots,” Borwn University, 2002.

[171] T. Tang, H.-C. Lin, Y. Zhao, Y. Fan, W. Chen, and M. Tomizuka, “Teach industrial robots peg-hole-insertion by human demonstration,” *2016 IEEE Int. Conf. Adv. Intell. Mechatronics*, pp. 488–494, 2016.

[172] S. Calinon, D. Florent, E. L. Sauser, D. G. Caldwell, and A. G. Billard, “An approach based on Hidden Markov Model and Gaussian Mixture Regression,” *IEEE Robot. Autom. Mag.*, vol. 17, pp. 44–45, 2010.

[173] J. Demiris and G. M. Hayes, “Imitation as a Dual-Route Process Featuring Predictive and Learning Components: A Biologically Plausible Computational Model,” *Imitation Anim. Artifacts*, vol. 21, no. 1, pp. 327–361, 2002.

[174] M. N. Nicolescu and M. J. Matarić, “Experience-based Representation Construction Learning From Human and Robot Teachers,” in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001.

[175] U. Nehmzow, O. Akanyeti, C. Weinrich, T. Kyriacou, and S. A. Billings, “Robot programming by demonstration through system identification,” in *IEEE International Conference on Intelligent Robots and Systems*, 2007, pp. 801–806.

[176] A. J. Ijspeert, J. Nakanishi, and S. Schaal, “Movement imitation with nonlinear dynamical systems in humanoid robots,” *IEEE Int. Conf. Robot. Autom.*, no. May, pp. 1398–1403, 2002.

[177] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato, "Learning from demonstration and adaptation of biped locomotion," *Rob. Auton. Syst.*, vol. 47, no. 2–3, pp. 79–91, 2004.

[178] S. Calinon and A. Billard, "Learning of Gestures by Imitation in a Humanoid Robot," *Imitation Soc. Learn. Robot. Humans Anim. Behav. Soc. Commun. Dimens.*, pp. 153–177, 2007.

[179] J. Aleotti and S. Caselli, "Robust trajectory learning and approximation for robot programming by demonstration," *Rob. Auton. Syst.*, vol. 54, no. 5, pp. 409–413, 2006.

[180] A. Billard and R. Siegward, "Robot Learning from Demonstration," in *Robotics and Autonomous Systems*, 2004, vol. 47, no. 2–3, pp. 65–67.

[181] R. Amit and M. Matarić, "Learning movement sequences from demonstration," in *Proceedings - 2nd International Conference on Development and Learning, ICDL 2002*, 2002, pp. 203–208.

[182] A. Billard and M. J. Mataric, "Learning human arm movements by imitation: Evaluation of a biologically inspiredconnectionist architecture," *Rob. Auton. Syst.*, vol. 37, no. 2–3, pp. 145–160, 2001.

[183]N. S. Pollard and J. K. Hodgins, "Generalizing demonstrated manipulation tasks," in *Springer Tracts in Advanced Robotics*, 2004, vol. 7 STAR, pp. 523–539.

[184] S. Calinon and A. Billard, "Recognition and reproduction of gestures using a probabilistic framework combining PCA, ICA and HMM," *Proc. 22nd Int. Conf. Mach. Learn. ICML 05*, pp. 105–112, 2005.

[185] M. Ogino, H. Toichi, Y. Yoshikawa, and M. Asada, "Interaction rule learning with a human partner based on an imitation faculty with a simple visuo- motor mapping," *Soc. Mech. Robot Program. by Demonstr. Robot. Auton. Syst.*, vol. 54, no. 5, pp. 414–418, 2006.

[186] M. Lopes and J. Santos-Victor, "Visual learning by imitation with motor representations," *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 35, no. 3, pp. 438–449, 2005.

[187] S. Chernova and M. Veloso, “Confidence-based policy learning from demonstration using Gaussian mixture models,” *Proc. 6th Int. Jt. Conf. Auton. agents multiagent Syst.*, p. 1, 2007.

[188] T. Inamura, “Acquisition of Probabilistic Behavior Decision Model based on the Interactive Teaching Method,” *Internatial Conf. Adv. Robot.*, pp. 523–528, 1999.

[189] J. Saunders, M. Imitation, P. Demonstration, C. L. Nehaniv, and K. Dautenhahn, “Teaching robots by moulding behavior and scaffolding the environment,” *Proc. 1st ACM SIGCHI/SIGART Conf. Human-robot Interact.*, pp. 118–125, 2006.

[190] G. E. Hovland, P. Sikka, B. J. McCarragher, and Ieee, “Skill acquisition from human demonstration using a hidden markov model,” *Robot. Autom. 1996. Proceedings., 1996 IEEE Int. Conf.*, vol. 3, pp. 2706–2711, 1996.

[191] K. R. Dixon and P. K. Khosla, “Learning by observation with mobile robots: a computational approach,” *IEEE Int. Conf. Robot. Autom. 2004. Proceedings. ICRA ’04. 2004*, vol. 1, pp. 1–6, 2004.

[192] P. E. Rybski and R. M. Voyles, “Interactive task training of a mobile robot through human gesture recognition,” in *IEEE International Conference on Robotics and Automation*, 1999, vol. 1, pp. 664–669.

[193] A. Lockerd and C. Breazeal, “Tutelage and socially guided robot learning,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008.

[194] S. Chernova and M. Veloso, “Teaching multi-robot coordination using demonstra- tion of communication and state sharing,” in *International Conference on Autonomous Agents and Multiagent Systems*, 2008.

[195] A. Ude, “Trajectory generation from noisy positions of object features for teaching robot paths,” *Rob. Auton. Syst.*, vol. 11, no. 2, pp. 113–127, 1993.

[196] S. Calinon and A. Billard, “Statistical Learning by Imitation of Competing Constraints in Joint Space and Task Space,” *Adv. Robot.*, vol. 23, no. 15, pp. 2059–2076, 2009.

[197] A. J. Ijspeert, J. Nakanishi, and S. Schaal, “Trajectory formation for imitation with nonlinear dynamical systems,” *Proc. 2001 IEEE/RSJ Int. Conf. Intell. Robot. Syst. Expand. Soc. Role Robot. Next Millenn. (Cat. No.01CH37180)*, vol. 2, pp. 752–757, 2001.

[198] T. Inamura, I. Toshima, and Y. Nakamura, “Acquiring motion elements for bidirectional computation of motion recognition and generation,” *Exp. Robot. VIII*, pp. 357–366, 2003.

[199] D. Grimes, R. Chalodhorn, and R. Rao, “Dynamic Imitation in a Humanoid Robot through Nonparametric Probabilistic Inference,” *Robot. Sci. Syst.*, pp. 1–6, 2006.

[200] S. Schaal, “Constructive Incremental Learning From Only Local Information,” *Neural Comput.*, vol. 10, no. 8, pp. 2047–2084, 1998.

[201] S. Vijayakumar, S. Schaal, and L. Angeles, “Incremental Online Learning in High Dimensions,” *Science (80-.).*, no. i, pp. 1–34, 2005.

[202] D. Nguyen-Tuong and J. Peters, “Local Gaussian process regression for real-time model-based robot control,” *2008 IEEE/RSJ Int. Conf. Intell. Robot. Syst. IROS*, pp. 380–385, 2008.

[203] A. J. Ijspeert, J. Nakanishi, and S. Schaal, “Trajectory formation for imitation with nonlinear dynamical systems,” in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, 2001, vol. 2, pp. 752–757.

[204] S. Calinon, F. Guenter, and A. Billard, “On learning the statistical representation of a task and generalizing it to various contexts,” *Proc. - IEEE Int. Conf. Robot. Autom.*, vol. 2006, no. May, pp. 2978–2983, 2006.

[205] S. Calinon and A. Billard, “A probabilistic programming by demonstration framework handling constraints in joint space and task space,” *2008 IEEE/RSJ Int. Conf. Intell. Robot. Syst. IROS*, pp. 367–372, 2008.

[206] J. Kober and J. Peter, “Policy search for motor primitives in robotics,” *Springer Tracts Adv. Robot.*, vol. 97, pp. 83–117, 2014.

[207] J. Peters and S. Schaal, “Policy gradient methods for robotics,” *IEEE Int. Conf. Intell. Robot. Syst.*, pp. 2219–2225, 2006.

[208] A. Gipsman, L. Rauschert, M. Daneshvar, and P. Knott, “Evaluating the reproducibility of motion analysis scanning of the spine during walking,” *Adv. Med.*, vol. 2014, no. 2014, p. 9, 2014.

[209] R. A. Siston and S. L. Delp, “Evaluation of a new algorithm to determine the hip joint center,” *J. Biomech.*, vol. 39, no. 1, pp. 125–130, 2006.

[210] “BONITA Size.Performance.Reliability.” [Online]. Available: http://www.prophysics-sol.se/assets/Uploads/Bonita-Kamera.pdf.

[211] F. C. Park and B. J. Martin, “Robot sensor calibration: solving AX=XB on the Euclidean group - Robotics and Automation, IEEE Transactions on,” *IEEE Trans. Robot.*, vol. 10, no. 5, pp. 717–721, 1994.

[212] F. Boochs, R. Schütze, C. Simon, F. Marzani, H. Wirth, and J. Meier, “Increasing the accuracy of untaught robot positions by means of a multi-camera system,” *2010 Int. Conf. Indoor Position. Indoor Navig. IPIN 2010 - Conf. Proc.*, no. September, pp. 15–17, 2010.

[213] F. Dietrich *et al.*, “On contact models for assembly tasks: Experimental investigation beyond the peg-in-hole problem on the example of force-torque maps,” *IEEE/RSJ 2010 Int. Conf. Intell. Robot. Syst. IROS 2010 - Conf. Proc.*, pp. 2313–2318, Oct. 2010.

[214] L. R. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition,” *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[215] L. R. Welch, “Hidden Markov Models and the Baum-Welch Algorithm,” *IEEE Inf. Theory Soc. Newsl.*, vol. 53, no. 4, p. 1,10-13, 2003.

[216] R. A. DeVore, B. Jawerth, and B. J. . Lucier, “Image Compression Through Wavelet Transform Coding,” *IEEE Trans. Inf. Technol.*, vol. 38, no. 2, pp. 719–746, 1992.

[217] D. L. Donoho, “De-Noising by Soft-Thresholding,” *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, 1995.

[218] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, “A Symbolic Representation of Time Series, with Implications for Streaming Algorithms,” *SIGMOD Work. Res. Issues Data Min. Knowl. Discov.*, pp. 2–11, 2003.

[219] D. Posada and T. R. Buckley, “Model selection and model averaging in phylogenetics: advantages of akaike information criterion and bayesian approaches over likelihood ratio tests.,” *Syst. Biol.*, vol. 53, no. 5, pp. 793–808, 2004.

[220] M. Halaki and K. a Ginn, “Normalization of EMG Signals: To Normalize or Not to Normalize and What to Normalize to?,” *Comput. Intell. Electromyogr. Anal. - A Perspect. Curr. Appl. Futur. Challenges*, pp. 175–194, 2012.

[221] V. Dzhafarov and A. Barkana, “Principal Component Analysis,” *Audio*, vol. 9, no. 6, pp. 655–662, 2001.

[222] W. Krzanowski, *Principles of multivariate analysis*. OUP Oxford, 2000.

[223] G. A. F. Seber, *Multivariate observation*. John Wiley & Sons, 1984.

[224] R. Battiti, “First and second order methods for learning: Between steepest descent and Newton’s method,” *Neural Comput.*, vol. 4, no. 2, pp. 141–166, 1992.

[225] M. Caudill and C. Butler, “Understanding Neural Net-works: Computer Explorations,” 1992.

[226] Percussion Plus, “Percussion Plus.” [Online]. Available: https://percussionplus.co.uk/.

[227] S. Woods and M. West, “The Psychology of Work and Organisations,” *Psychol. Work Organ.*, pp. 429–465, 2014.

[228] J. Van Asselt, “Ball Game Apparatus,” 7134975.

[229] T. Lefebvre, J. Xiao, H. Bruyninckx, and G. De Gersem, “Active compliant motion : a survey,” *Adv. Robot.*, vol. 19, no. 5, pp. 479–499, 2015.

[230] A. Caird-Daley, S. R. Fletcher, and W. Baker, “Automating Human Skills: Preliminary Development of a Human Factors Methodology to Capture Tacit Cognitive Skills.,” *Proc. 11th Int. Conf. Manuf. Res.*, no. October, pp. 177–186, 2013.

[231] B. Kirwan and L. K. Ainsworth, “A guide to task analysis,” *Univ. Michigan*, vol. 16, no. 2, p. 417, 1992.

[232] J. Rasmussen, “Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human performance models,” *Syst. Des. Hum. Interact.*, pp. 291–300, 1987.

[233] J. Rasmussen, “Information Processing and Human-Machine Interaction: An Approach to Cognitive Engineering,” *Elsevier Sci.*, vol. 12, p. 104, 1986.

[234] Y. . Zhao, T. Johson, Y. . Goh, and J. Mike, “A SENSOR DESIGN AND DATA ANALYSIS FOR AUTOMATIC DRUM BEATER WINDING,” in *ICED*, 2015, no. July, pp. 1–10.

[235] W. Kong, S. Sessa, M. Zecca, and A. Takanishi, “Anatomical Calibration through Post-Processing of Standard Motion Tests Data.,” *Sensors (Basel).*, vol. 16, no. 12, pp. 0–21, 2016.

[236] L. Rozo, S. Calinon, and D. G. Caldwell, “Learning force and position constraints in human-robot cooperative transportation,” *IEEE RO-MAN 2014 - 23rd IEEE Int. Symp. Robot Hum. Interact. Commun. Human-Robot Co-Existence Adapt. Interfaces Syst. Dly. Life, Ther. Assist. Soc. Engag. Interact.*, pp. 619–624, 2014.

[237] T. Mori, M. Howard, and S. Vijayakumar, “Model-free apprenticeship learning for transfer of human impedance behaviour,” *IEEE-RAS Int. Conf. Humanoid Robot.*, pp. 239–246, 2011.

[238] M. Howard, S. Klanke, M. Gienger, C. Goerick, and S. Vijayakumar, “A novel method for learning policies from constrained motion,” *2009 IEEE Int. Conf. Robot. Autom.*, no. November 2016, pp. 1717–1723, 2009.

[239] V. Mnih *et al.*, “Playing Atari with Deep Reinforcement Learning,” *arXiv Prepr. arXiv1312.5602*, pp. 1–9, 2013.

[240] D. Silver *et al.*, “Mastering the game of Go with deep neural networks and tree search,” *Nature*, vol. 529, no. 7587, pp. 484–489, 2016.

[241] S. Gu, E. Holly, T. Lillicrap, and S. Levine, “Deep Reinforcement Learning for Robotic Manipulation,” *ArXiv e-prints*, 2016.

[242] A. Kupcsik, M. P. Deisenroth, J. Peters, A. P. Loh, P. Vadakkepat, and G. Neumann, “Model-based contextual policy search for data-efficient generalization of robot skills,” *Artif. Intell.*, 2014.

Appendix A - Motion Reproduction results for PiH Experiment

This Appendix shows the additional comparison results for the motion reproductions in chapter 4. Results from subject A, C, D, E and F are summarised in the tables below:

Table A-1 Summarised results for subject A. a) implicit time reproduction b) explicit time reproduction

a)								
Trial	**Kv**	**Kp**	**Yaw (rad)**		**Pitch (rad)**		**Roll (rad)**	
			MSE(rad)	**R**	**MSE(rad)**	**R**	**MSE(rad)**	**R**
1	**0.3**	**0.03**	**0.0032**	**0.9958**	**0.0005**	**0.9977**	**0.0023**	**0.7767**
2	**0.3**	**0.1**	**0.002**	**0.9964**	**0.0004**	**0.9947**	**0.0007**	**0.8543**
3	**0.7**	**0.1**	**0.0005**	**0.9965**	**0.0011**	**0.9819**	**0.0033**	**0.6981**
4	**0.5**	**0.1**	**0.0006**	**0.9942**	**0.001**	**0.9870**	**0.0037**	**0.4835**
5	**0.3**	**0.08**	**0.0006**	**0.9942**	**0.001**	**0.9870**	**0.0037**	**0.4835**
All	**0.42**	**0.082**	**0.0015±0.001**	**0.9963±0.001**	**6e-04±0.001**	**0.9905±0.006**	**0.0024±0.001**	**0.5954±0.277**

b)							
Trial	**Input Variable**	**Yaw (rad)**		**Pitch (rad)**		**Roll (rad)**	
		MSE(rad)	**R**	**MSE(rad)**	**R**	**MSE(rad)**	**R**
1	**[t,f]**	**0.0081**	**0.9792**	**0.0010**	**0.9823**	**0.0007**	**0.9106**
2	**[t,f]**	**0.0066**	**0.9880**	**0.0015**	**0.9875**	**0.0001**	**0.9778**
3	**[t,f]**	**0.0027**	**0.9936**	**0.0007**	**0.9894**	**0.0007**	**0.8396**
4	**[t,f]**	**0.0033**	**0.9853**	**0.0019**	**0.9730**	**0.0016**	**0.6771**
5	**[t,f]**	**0.0027**	**0.9779**	**0.0018**	**0.9843**	**0.0013**	**0.4033**
All	**[t,f]**	**0.0047±0.002**	**0.9848±0.006**	**0.0014±0.001**	**0.9833±0.0064**	**0.00008±0.001**	**0.7617±0.229**

Table A-2 Summarised results for subject C. a) implicit time reproduction b) explicit time reproduction

a)								
Trial	**Kv**	**Kp**	**Yaw (rad)**		**Pitch (rad)**		**Roll (rad)**	
			MSE(rad)	**R**	**MSE(rad)**	**R**	**MSE(rad)**	**R**
1	**0.3**	**0.1**	**0.0045**	**0.9959**	**0.0002**	**0.9912**	**0.0029**	**0.9919**
2	**1.7**	**0.1**	**0.0032**	**0.9853**	**0.0006**	**0.9859**	**0.0024**	**0.9834**
3	**0.3**	**0.1**	**0.0049**	**0.9671**	**0.0060**	**0.9858**	**0.0035**	**0.9716**
4	**0.3**	**0.1**	**0.0014**	**0.9885**	**0.0031**	**0.9949**	**0.0023**	**0.9786**
5	**0.7**	**0.1**	**0.0018**	**0.9788**	**0.0011**	**0.9854**	**0.0021**	**0.9584**
All	**0.66**	**0.1**	**0.0031±0.001**	**0.9831±0.010**	**0.0022±0.002**	**0.9887±0.004**	**0.0026±6e-4**	**0.9768±0.012**

b)							
Trial	Input Variable	Yaw (rad)		Pitch (rad)		Roll (rad)	
		MSE(rad)	R	MSE(rad)	R	MSE(rad)	R
1	[t,f]	0.0011	0.9802	0.0013	0.9848	0.0006	0.9919
2	[t,f]	0.0005	0.9956	0.0006	0.9948	0.0008	0.9905
3	[t,f]	0.001	0.9821	0.0090	0.9813	0.0041	0.9810
4	[t,f]	0.0005	0.9914	0.0058	0.9906	0.0004	0.9944
5	[t,f]	0.0008	0.9847	0.0041	0.9713	0.0011	0.9844
All	[t,f]	0.0001±0.001	0.9868±0.006	0.0042±0.003	0.9846±0.009	0.0014±0.002	0.9885±0.006

Table A-3 Summarised results for subject D. a) implicit time reproduction b) explicit time reproduction

a)								
Trial	Kv	Kp	Yaw (rad)		Pitch (rad)		Roll (rad)	
			MSE(rad)	R	MSE(rad)	R	MSE(rad)	R
1	1.5	0.1	0.0002	0.9974	0.0007	0.9961	0.0002	0.9980
2	0.7	0.1	0.0001	0.9988	0.0009	0.9979	0.0001	0.9988
3	1.7	0.03	0.0007	0.9927	0.0005	0.9815	0.0024	0.9868
4	1.7	0.1	0.0022	0.9924	0.0019	0.9912	0.0003	0.9892
5	0.3	0.03	0.0004	0.9964	0.0011	0.9985	0.0001	0.9962
All	1.18	0.072	7e-4±8e-4	0.9955±0.0029	0.001±5e-4	0.993±0.0071	6e-4±0.001	0.9938±0.0054

b)							
Trial	Input Variable	Yaw (rad)		Pitch (rad)		Roll (rad)	
		MSE(rad)	R	MSE(rad)	R	MSE(rad)	R
1	[t,f]	0.0002	0.9966	0.0007	0.9903	0.0004	0.9850
2	[t,f]	0.0004	0.9976	0.0007	0.9957	0.0004	0.9945
3	[t,f]	0.0049	0.9828	0.0015	0.9604	0.0006	0.9858
4	[t,f]	0.0006	0.9988	0.0027	0.9960	0.0001	0.9945
5	[t,f]	0.0007	0.9950	0.0025	0.9875	0.0017	0.9697
All	[t,f]	0.0014±0.002	0.9941±0.0065	0.0016±9e-4	0.9860±0.0148	6e-4±6e-4	0.9859±0.0102

Table A-4 Summarised results for subject E. a) implicit time reproduction b) explicit time reproduction

a)								
Trial	Kv	Kp	Yaw (rad)		Pitch (rad)		Roll (rad)	
			MSE(rad)	R	MSE(rad)	R	MSE(rad)	R
1	0.3	0.1	0.0008	0.9504	0.0067	0.9766	0.0048	0.9474
2	0.3	0.1	0.0007	0.9695	0.0111	0.9307	0.0009	0.9695
3	0.3	0.06	0.0002	0.9901	0.0004	0.9978	0.0003	0.9537
4	0.5	0.1	0.0043	0.9168	0.0276	0.8435	0.0015	0.9788
5	1.7	0.1	0.0001	0.9966	0.0003	0.9985	0.0016	0.9345
All	0.62	0.092	0.0012±0.002	0.9647±0.032	0.0092±0.011	0.9494±0.065	0.0018±0.002	0.956±0.017

b)							
Trial	Input Variable	Yaw (rad)		Pitch (rad)		Roll (rad)	
		MSE(rad)	R	MSE(rad)	R	MSE(rad)	R
1	[t,f]	0.0011	0.8716	0.0026	0.9631	0.0072	0.7554
2	[t,f]	0.0002	0.9881	0.0006	0.9933	0.001	0.9223
3	[t,f]	0.0004	0.9884	0.0006	0.9916	0.0006	0.9605
4	[t,f]	0.0014	0.919	0.0084	0.9195	0.0008	0.8781
5	[t,f]	0.0004	0.9944	0.0008	0.9893	0.0008	0.9477
All	[t,f]	6e-4±4e-4	0.9523±0.0547	0.0026±0.0033	0.9714±0.0315	0.0021±0.0028	0.8928±0.083

Table A-5 Summarised results for subject F. a) implicit time reproduction b) explicit time reproduction

a)								
Trial	Kv	Kp	Yaw (rad)		Pitch (rad)		Roll (rad)	
			MSE(rad)	R	MSE(rad)	R	MSE(rad)	R
1	0.3	0.1	0.0005	0.9902	0.0001	0.9830	0.0005	0.9866
2	0.7	0.1	0.0002	0.9987	0.0005	0.9540	0.0041	0.8511
3	0.3	0.1	0.0013	0.9974	0.0002	0.9946	0.0003	0.9921
4	0.3	0.1	0.0004	0.9962	0.0001	0.9939	0.0001	0.9936
5	1.7	0.1	0.002	0.9870	0.0007	0.9739	0.0018	0.9260
All	0.62	0.092	8e-4±0.003	0.9939±0.0323	3e-4±0.021	0.9799±0.0653	0.0014±0.0027	0.9499±0.0156

b)							
Trial	Input Variable	Yaw (rad)		Pitch (rad)		Roll (rad)	
		MSE(rad)	R	MSE(rad)	R	MSE(rad)	R
1	[t,f]	0.0019	0.9857	0.0001	0.9766	0.0009	0.9670
2	[t,f]	0.0028	0.9751	0.0002	0.9773	0.0008	0.9631
3	[t,f]	0.002	0.9897	0.0004	0.9914	0.0009	0.9759
4	[t,f]	0.0025	0.9855	0.0003	0.9768	0.0016	0.9490
5	[t,f]	0.0009	0.9769	0.0005	0.9838	0.0004	0.9783
All	[t,f]	0.002±7e-4	0.9826±0.0063	3e-4±1e-4	0.9812±0.0065	9e-4±4e-4	0.9667±0.0117

Appendix B - Matlab Program (Human skills encoding using GMM-HMM approach)

```
%%% Time based HMM GMR encoding demo for chapter 4 -- force without
time
close all;
clear all;
clc;
%% parameters
model = [];
model.nbStates = 5; %Number of components in the GMM
model.nbVarPos = 3; %Dimension of position data (here: x1,x2)
model.nbVarOrient = 0;
model.nbVarforce = 1; %Dimension of force data (here: f1,f2)
model.nbDeriv = 2; %Number of static&dynamic features (D=2 for [x,dx],
D=3 for [x,dx,ddx], etc.)
% model.nbVar = model.nbVarPos *
model.nbDeriv+model.nbVarforce; %Dimension of state vector
model.dt = 1; %Time step (without rescaling, large values such as 1
has the advantage of creating clusers based on position information)
nbSamples = 5; %Number of demonstrations
nbData = 200; %Number of datapoints in a trajectory
model.dt = 0.1; %Time step
model.sub_name = 'subject B';
time_flag = 1;
%% load marked data in
curser %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
load('curser_B.mat');
[input,target] =
prepare_Data_PiH('sEMG_FT_PiH_subjectB_ft_message.csv','','','','sEMG_
FT_PiH_subjectB_orange_imu.csv',1);
%% optional smooth the input
signal %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% for i = 1:3
% input_smoothed(i,:) =
smooth([1:length(input)]',input(i,:),0.05,'rloess');
% end
input_smoothed = input;
data = [PCA_plus(target(:,:),0,1);input_smoothed];
%% normalization or scaling
data %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
normalization_flag = 0;
%%  training data prepare from curser
selection %%%%%%%%%%%%%%%%%%%%%%%%%%%%
[tmp_data]=training_data_prepare_PiH(data,pih_curser_B(1:10),normaliza
tion_flag);
%% resampling
data %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Data = [];
s = [];
Data_hmm = [];
for n = 1:nbSamples
    s(n).Data = spline(1:size(tmp_data.data_cell{n},2),
tmp_data.data_cell{n},
linspace(1,size(tmp_data.data_cell{n},2),nbData)); %Resampling
    s(n).Data = [s(n).Data;
gradient(s(n).Data(2:end,:))/model.dt]; %Velocity computation
    s(n).nbData = size(s(n).Data,2);
```

```
    Data = [Data s(n).Data];
    Data_hmm(:,:,n) = s(n).Data;
end
%%%%% GMM training and generalization across
demons %%%%%%%%%%%%%%%%%%%%%%
loglik = [];
for m = 1:nbSamples
    idx = [1:nbData*(m-1)+nbData];
    model = init_GMM_kbins(Data(:,idx), model,m);
    model.Trans = zeros(model.nbStates);
    for i=1:model.nbStates-1
        model.Trans(i,i) = 1 - (model.nbStates / nbData);
        model.Trans(i,i+1) = model.nbStates / nbData;
    end
    model.Trans(model.nbStates,model.nbStates) = 1.0;
    model.StatesPriors = zeros(model.nbStates,1);
    model.StatesPriors(1) = 1;
    %Parameters refinement with EM
    model = EM_HMM(s(1:m), model);
    for j = 1:nbSamples
        [F,obslik] = mhmm_logprob({s(j).Data},  model.StatesPriors,
model.Trans, model.Mu, model.Sigma);
        loglik(m,j) = F;
    end
end
figure;bar3(loglik');
%% Nonlinear force profile retrieval
currF = s(2).Data(1,:);
currPos = s(2).Data(2:4,1); %Current position (initialization)
currVel = [0; 0;0]; %Current velocity (initialization)
currAcc = [0; 0;0]; %Current acceleration (initialization)
kP_max = 10;
kv = 1/model.dt;
%% Reproduction loop
[reprData.mu,reprData.sigma] =
impedence_control(currPos,currVel,currF,kP_max,kv,model);
%%%%% plot traning data results  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
figure('position',[10,10,1300,500]);
%% Plot GMM
subplot(1,3,1); hold on; axis off; title('GMM');
plot(Data(2,:),Data(3,:),'.','markersize',8,'color',[.5 .5 .5]);
plotGMM(model.Mu(2:3,:), model.Sigma(2:3,2:3,:), [.8 0 0],.5);
set(gca,'Xtick',[]); set(gca,'Ytick',[]);
subplot(1,3,2); hold on; axis off; title('GMM');
plot(Data(1,:),Data(2,:),'.','markersize',8,'color',[.5 .5 .5]);
plotGMM(model.Mu(1:2,:), model.Sigma(1:2,1:2,:), [.8 0 0],.5);
set(gca,'Xtick',[]); set(gca,'Ytick',[]);
%% Plot GMR
subplot(1,3,3); hold on; axis off; title('GMR');
plot(Data(2,:),Data(3,:),'.','markersize',8,'color',[.5 .5 .5]);
plotGMM(reprData.mu(1:2,:), reprData.sigma(1:2,1:2,:), [0 .8 0], .1);
plot(reprData.mu(1,:),reprData.mu(2,:),'-','linewidth',2,'color',[0 .4
0]);
set(gca,'Xtick',[]); set(gca,'Ytick',[]);
%% reproduction on new
trajectories %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[GMM_data_new]=training_data_prepare_PiH(data,pih_curser_B(11:20),norm
alization_flag);
nbSamples = 5;
Data_new = [];
s_new = [];
```

```
Data_hmm_new = [];
for n = 1:nbSamples
    s_new(n).Data = spline(1:size(GMM_data_new.data_cell{n},2),
GMM_data_new.data_cell{n},
linspace(1,size(GMM_data_new.data_cell{n},2),nbData)); %Resampling
    s_new(n).Data = [s_new(n).Data;
gradient(s_new(n).Data(2:end,:))/model.dt]; %Velocity computation
    s_new(n).nbData = size(s_new(n).Data,2);
    Data_new = [Data_new s_new(n).Data];
    Data_hmm_new(:,:,n) = s_new(n).Data;
end
for i = 1:nbSamples
    currF = s_new(i).Data(1,:);
    currPos = s_new(i).Data(2:4,1); %Current position (initialization)
    currVel = [0; 0;0]; %Current velocity (initialization)
    currAcc = [0; 0;0]; %Current acceleration (initialization)
    [reprData_new.mu(:,:,i),reprData_new.sigma(:,:,:,i)] =
impedence_control(currPos,currVel,currF,kP_max,kv,model);
end
%% generate attractor path from learned GMM
plotregression %%%%%%%%%%%%%%%%
colorlist = [[0 0 0];[1 0 1];[0 1 1];[1,0,0];[0 0 1]];
for i = 1:nbSamples
    figure;
    for kk =1:3
        h2 = [];
        subplot(3,1,kk);hold on
        %     xlabel('Time(s)','fontsize',16);
ylabel(y_label{i},'fontsize',16);
        p1 = plot(s_new(i).Data([1],:),reprData_new.mu([kk],:,i), '--
','LineWidth', 2, 'color', [colorlist(1,:)]);
        p2 = plot(s_new(i).Data([1],:),s_new(i).Data([kk+1],:), '-
','LineWidth', 2, 'color', [colorlist(2,:)]);
        h2 = [h2,p1,p2];
        legend([p1,p2],'Reproduction','Target','Location','NorthEast');
        plotGMM(model.Mu([1,kk+1],:),  model.Sigma([1,kk+1],[1,kk+1],:),
[.8 0.0 0],0.4);
        ax = gca;
        ax.XLabel.FontName = 'Times New Roman';
        ax.XLabel.FontWeight = 'bold';
        ax.XLabel.FontSize = 16;
        ax.XAxis.FontWeight = 'bold';
        ax.XAxis.FontName = 'Times New Roman';
        ax.YLabel.FontName = 'Times New Roman';
        ax.YLabel.FontWeight = 'bold';
        ax.YLabel.FontSize = 16;
        ax.YAxis.FontWeight = 'bold';
        ax.YAxis.FontName = 'Times New Roman';
        set(gca,'FontSize',12,'FontName','Times New
Roman','FontWeight','bold');
        axis tight;
    end
    if i==1
        title(['Skills encoding for',
model.sub_name],'FontSize',16,'FontName','Times New
Roman','FontWeight','bold')
    end
end
figure;
p1 = plot(s_new(1).Data([1],:),'-','LineWidth', 2, 'color',
[colorlist(1,:)]);hold on
```

```
p2 = plot(s_new(2).Data([1],:),'--','LineWidth', 2, 'color',
[colorlist(2,:)]);
p3 = plot(s_new(3).Data([1],:),'.','LineWidth', 2, 'color',
[colorlist(3,:)]);
p4 = plot(s_new(4).Data([1],:),':','LineWidth', 2, 'color',
[colorlist(4,:)]);
p5 = plot(s_new(5).Data([1],:),'-.','LineWidth', 2, 'color',
[colorlist(5,:)]);
legend([p1,p2,p3],'Force 1','Force 2','Force
3','Location','NorthEast');

end

function [reprData,expSigma] =
impedence_control(currPos,currVel,currF,kp_max,kv,model)
%% Reproduction loop
nbVarOut = 3;
diagRegularizationFactor = 1E-8; %Optional regularization term
expSigma = zeros(nbVarOut,nbVarOut,200);
for t=1:200
    %% Keep trace of the motion
    reprData(1:3,t) = currPos;
    reprData(4:6,t) = currVel;
    %% Compute the influence of each Gaussian
    for j=1:model.nbStates
        B(j,t) = gaussPDF([currF(t);currPos], model.Mu(1:4,j),
model.Sigma(1:4,1:4,j));
    end
    if t ==1
        h(:,1) = model.StatesPriors(:).*B(:,t);
        [h(:,1)] = normalise(h(:,t));
    else
        m = model.Trans' * h(:,t-1);
        h(:,t) = m(:) .* B(:,t);
        [h(:,t)] = normalise(h(:,t));
    end
    %% Compute the desired position and desired velocity through GMR
    targetPos=[0;0;0]; targetVel=[0;0;0];
    for j=1:model.nbStates
        MuTmp_pos(:,j)=(model.Mu(2:4,j) + ...

model.Sigma(2:4,[1,5:7],j)*inv(model.Sigma([1,5:7],[1,5:7],j)) *
([currF(t);currVel]-model.Mu([1,5:7],j)));
        targetPos = targetPos + h(j,t) .* MuTmp_pos(:,j);
        targetVel = targetVel + h(j,t) .* (model.Mu(5:7,j) + ...
            model.Sigma(5:7,[1:4],j)*inv(model.Sigma([1:4],[1:4],j)) *
([currF(t);currPos]-model.Mu([1:4],j)));
    end
    pos_target(:,t) = targetPos;
    vel_target(:,t) = targetVel;
    for i=1:model.nbStates
        SigmaTmp = model.Sigma(2:4,2:4,i) -
model.Sigma(2:4,[1,5:7],i)/model.Sigma([1,5:7],[1,5:7],i) *
model.Sigma([1,5:7],2:4,i);
        expSigma(:,:,t) = expSigma(:,:,t) + h(i,t) * (SigmaTmp +
MuTmp_pos(:,i)*MuTmp_pos(:,i)');
    end
    expSigma(:,:,t) = expSigma(:,:,t) - targetPos*targetPos' +
eye(nbVarOut) * diagRegularizationFactor;
    %% Acceleration defined by mass-spring-damper system (impedance
controller)
```

```
    for j=1:model.nbStates
        LL_max(j) = gaussPDF([model.Mu(1,j);model.Mu(2:4,j)],
model.Mu(1:4,j), model.Sigma(1:4,1:4,j));
        LL_min(j) = gaussPDF([currF(t);currPos], model.Mu(1:4,j),
model.Sigma(1:4,1:4,j));
    end
    LL_max = log(LL_max);
    LL_min_ = log(LL_min);
    ind = [];
    for k = 1:length(LL_min_)
        if LL_min_(k)==-Inf
            ind = [ind,k];
        end
    end
    LL_min_(ind) = [];
    kp(:,t) = kp_max*(max(LL_max)-log(sum(LL_min)))/(max(LL_max)-
min(LL_min_));
    currAcc = (targetVel-currVel).*kv + (targetPos-currPos).*kp(:,t);
    %%Update velocity
    currVel = currVel + model.dt*currAcc;
    %% Update position
    currPos = currPos + model.dt*currVel;
end
end
```

Appendix C – Complete Human Factor Analysis Results for Beater Winding

This Appendix provides a complete human factor analysis (Task decomposition) for the beater winding process:

HTA Level		Purpose	Beater Variation				Cues	Decision	Action / Response	Performance level	Potential errors	Potential to correct errors	Critical values	Process variations
0	Wind a Percussion Plus beater. Plan 0: 1 (077 only) – 2 – 3 – 4 – 5 (All beaters)		077 soft hard beater	076 medium beater	071 soft beater	075 hard beater								
1	Prepare for winding. 077 Soft hard beater only: Plan 1: 1.1 – 1.3		Only for 077 soft hard 1 - 1.1.3.1.2	Not applicable for theses beaters	Not applicable for theses beaters	Not applicable for theses beaters								
1.1	Acquire appropriate beater	There are different sticks with different heads for each beater	Soft Hard	N/a	N/a	N/a	VISUAL: look for the designation on the side of the beater stick	Are there any soft hard beaters available?	Acquire soft hard beaters	**Rule based**: relies on a knowledge of which beater is required for each variation	The wrong beater is chosen	At this stage a new beater can be picked up, at a later stage the winding and sewing will need to be taken apart	Beater type	
1.2	Place rubber band over the beater head	Increase the grip of the beater head for the felt		N/a	N/a	N/a	VISUAL: check that the tubing is on the beater head evenly with a similar amount at the top as there is at the bottom	Is the band on the beater head?	Move the tube ensuring it is secure and evenly placed on the head of the beater	**Rule based**			Bicycle inner tube cut to 3/4"	
1.3	Secure the felt to the head of the beater	Ensures that the felt does not slip off during winding stage (2)		N/a	N/a	N/a								

HTA Level		Purpose	Beater Variation				Cues	Decision	Action / Response	Performance level	Potential errors	Potential to correct errors	Critical values	Process variations
			077 soft hard beater	076 medium beater	071 soft beater	075 hard beater								
1.3.5.1 .2	Thread the needle through loop created	To create a locking point for the thread preventing the need for knotting		N/a	N/a	N/a	VISUAL: check that the thread goes through the loop	Is the thread passing through the loop?	If not try again	**Rule based**				
1.3.5.1 .3	Pull the thread tight, pulling the loop flat	Locking off the thread and ensuring that the loop is pulled flat		N/a	N/a	N/a	VISUAL: loop lies flat on the felt	The loop does not lie flat	Pull the thread tighter, or loosen it off and pull the thread again	**Rule based**				
1.3.5.2	Run the needle and thread up through the middle of the felt to the top	Create a tail so that the tie off is less likely to come undone before winding		N/a	N/a	N/a	VISUAL: ensure that the thread runs through the middle of the felt	Is the thread caught and runs through the middle of the felt?	Reattempt if it is not.	**Rule based**				
1.3.5.3	Pull thread tight and cut the thread close to the felt so that no thread is found trailing	So that the thread doesn't interfere with the shape created during winding		N/a	N/a	N/a	VISUAL: check whether the thread is trailing out of the felt	The thread is not cut close enough to the felt	Trim the thread until it does not trail from the felt	**Rule based**				
2	**Winding beater. Plan 2: 2.1 (all variations except 077 soft hard) 2.2 – 2.4 (all variations)**													
2.1	**Get appropriate beater**	There are different sticks with different heads for each beater	Soft hard	Medium	Longer beater and no designation on side of stick	Hard	VISUAL: look for the designation on the side of the beater stick	Are there any soft hard beaters available?	Acquire soft hard beaters	**Rule based**: relies on a knowledge of which beater is required for each variation	The wrong beater is chosen	At this stage a new beater can be picked up for all but the soft hard beater, at a later stage the winding and sewing will need to be taken apart	See variation columns	

HTA Level		Purpose	Beater Variation				Cues	Decision	Action / Response	Performance level	Potential errors	Potential to correct errors	Critical values	Process variations
			077 soft hard beater	076 medium beater	071 soft beater	075 hard beater								
2.2	**Get appropriate colour of thread on spool**	Each type of beater has a different colour designated to it	Light blue thread	Red thread	Red	Blue	VISUAL: colour of the thread	Has the right colour been acquired?	Identify which colour is needed for current beater and get it.	**Rule based**: relies on a knowledge of which thread is required for each variation	The wrong thread is chosen	At this stage a new thread can be picked up, at a later stage the winding and sewing will need to be taken apart and the process will need to be recompleted	See variation columns. Length is not required because thread comes straight from a spool: ensure enough thread on the spool to complete task (min: 15 meters)	
2.3	**Wind appropriate number of horizontal winds on beater**	Cover the head of the beater	60	70	70	60	VISUAL: check that there is a circular area on the top platform of the beater head that is not covered by thread.						See variation columns. Beater held horizontally with winds at 60° to the beater stick. Each wind is from the bottom to the top of the beater head	
2.3.1	Secure thread to the head of the beater	To ensure a successful wind at the right tension												
2.3.1.1	Secure end of thread along the stick of the beater using thumb ensuring that the thread is flush with the underside of the beater	Thread length small enough to be held in place to begin the wind and to then be hidden in the winds					TACTILE: feel that the thread is held securely by the thumb against the beater stick	Is the thread being held? Is it held securely enough?	Increase pressure	**Rule based**	The thread is too long it will cause a disturbance to the wrap.	This will become apparent later and will require starting over	Enough to hold. Approx.: 2 cm	

HTA Level		Purpose	Beater Variation				Cues	Decision	Action / Response	Performance level	Potential errors	Potential to correct errors	Critical values	Process variations
			077 soft hard beater	076 medium beater	071 soft beater	075 hard beater								
2.3.1.2	Wind first three winds securing the thread around the beater then let go of the tail and push it up against the underside of the beater so that it is caught in the next winds	Prevent the end from coming loose and consequently all of the winding					Visual: look to check that the winds overlap to ensure that the tail is caught	Has the end been caught? Or has it slipped out from under the thumb?	While winding make sure that there is some overlap while se	**Rule based**	The tail is not caught in the winds	Catch the tail in subsequent winds		
2.3.2	Horizontal winding process													
2.3.2.1	Wind one wind after another shifting the head around using the index finger of the left hand as the thread is wound	Ensure an even coverage					VISUAL: identify any gaps and look at how the winds are lying	Is there an area left uncovered? Are the winds to loose?	Fill as many gaps as possible. Increase tension if winds feel or look to loose.	**Rule based:** act of winding is guided by rules and procedures stored in the operators memory	The coverage is uneven	Rewind ensuring that the winds are more even		
							TACTILE: ensure that tension is high but feeling the give in the thread to judge how much more tension it can manage.	Do the winds slip off of the beater head? Does the thread feel like it is straining?	Wind further away from the top of the beater head. Decrease the tension if the thread feels like it is under too much tension and may break.	**Skill Based:** level of tension to apply and where to wind to prevent winds from slipping off of the beater head and to ensure the right appearance	The wrong amount of tension is placed on the thread.	If there is not enough tension then the threads will slip off the beater: tension should be increased. Too much tension and the thread will snap requiring rewinding		
2.3.2.2	Hold thumb of left hand against the underside of the beater applying pressure	Prevents the thread from winding down the beater stick and creating a teardrop shaped beater head					VISUAL: identify whether the shape of the beater head is beginning to distort	Is the thread winding down the stick or staying close to the beater head?	Increase the tension of the winds and the pressure applied to the underside of the beater head	**Rule based**	Not enough pressure is applied and shape starts distorting	Start over winding and increase pressure		

HTA Level		Purpose	Beater Variation				Cues	Decision	Action / Response	Performance level	Potential errors	Potential to correct errors	Critical values	Process variations
			077 soft hard beater	076 medium beater	071 soft beater	075 hard beater								
2.4	**Wind appropriate number of vertical winds**	Tidy up the beater head. At top point of the beater, go over the central point of the beater head with the thread.	60. Slower more conscious winding due to less possible winds	70	70	60	VISUAL: look for any gaps in the wind. TACTILE: feel for the right level of tension	Are there any gaps? Are the winds slipping off of the beater?	Cover any gaps with winds. Increase the tension	**Knowledge based:** monitoring the accuracy of the winds **Skill Based:** level of tension to apply and where to wind to prevent winds from slipping off of the beater head	Gaps are visible in the wind	Look to fill in while wrapping	See variation columns. Beater held vertically. Each wind is wound from the bottom to the top of the beater head.	
2.5	**Cover any gaps by dragging the needle down the beater head to shift around winds**	Ensure even coverage and a neat appearance					VISUAL: identify any gaps	has the entire beater been checked	check entire beater and perform action	**Knowledge:** consciously check the lie of the threads	Gaps may be missed	During stitching these can be covered		
3	**Top and bottom securing stitches**	Prevent the winds from slipping off with handling and use												
3.1	**Cut thread from spool maintaining tension on the head of the beater**	Thread needs to be cut from the spool so that it can be threaded onto a needle					Visual: check that the thread is long enough	Is the thread long enough?	If it is not long enough complete as many stitches as possible and cut a new length of thread and using the locking stitch (4.2.1 – 4.2.2) to attach thread, then continue with steps	**Rule based**	The thread is not held tight on the beater head and winds come undone	If possible rewind number of winds that have come undone or start winding process again	Approx.: Length of beater	

Figure C-1 Full task decomposition analysis for beater winding process

Appendix D – Sensor Calibration Table for Tension Measurement Unit

This Appendix provides the experiment records for the Tension Measurement Unit calibrations. Full Wheatstone bridge is used to generate load cell readings.

Table D-1 Sensor calibration table for TMU

	Sensor reading						
Spring load (N)	1	2	3	4	5	Mean	Std
0	8389938	8389871	8389951	8389931	8389904	8389919	31.85122
0.5	8393124	8392797	8393487	8393084	8393417	8393182	278.1379
1	8395314	8395177	8395406	8395770	8395564	8395446	229.2841
1.5	8397983	8397274	8397535	8397385	8397843	8397604	300.9917
2	8399632	8399908	8399424	8399226	8399940	8399626	307.7986
2.5	8401408	8401885	8401478	8401357	8401419	8401509	214.3252
3	8404504	8404243	8404342	8404632	8403862	8404317	294.669
3.5	8407065	8406766	8407271	8406830	8406899	8406966	203.5994
4	8408386	8408534	8408927	8409111	8409321	8408856	390.7924
4.5	8411258	8411667	8411035	8411256	8411461	8411335	238.8876
5	8414536	8414669	8414938	8414865	8414863	8414774	166.4082
5.5	8416908	8416948	8417600	8417091	8417630	8417235	353.2999
6	8419566	8419942	8419830	8420625	8419986	8419990	390.8071
6.5	8422385	8422175	8422502	8421920	8422476	8422292	244.273
7	8424581	8424447	8423921	8424521	8424808	8424456	327.8945
7.5	8426205	8426632	8426388	8426255	8426475	8426391	171.9288
8	8428433	8428192	8428087	8428770	8428288	8428354	265.1349
8.5	8430672	8430879	8430369	8430035	8430404	8430472	320.878
9	8432784	8433050	8433022	8433175	8432649	8432936	213.8843
9.5	8434805	8435385	8435636	8435075	8435344	8435249	318.081
10	8436733	8436488	8436389	8436424	8437122	8436631	305.5106
10.5	8440799	8439715	8440133	8440052	8440132	8440166	393.4879
11	8443272	8442992	8442542	8443340	8443489	8443127	373.4595
11.5	8444489	8444485	8444731	8444716	8444415	8444567	145.7813
12	8446441	8446015	8446091	8446847	8446380	8446355	329.917
12.5	8449047	8449060	8449076	8449183	8449025	8449078	61.4874
13	8451067	8451362	8451851	8451544	8451492	8451463	285.0275

The mean values are plotted in Figure D- (y-axis is scaled sensor readings). From the results, the raw data contains some noise but follows a linear trend. Therefore, a linear curve is fitted, and it indicates a strong linear relationship between the sensor reading and the tension.

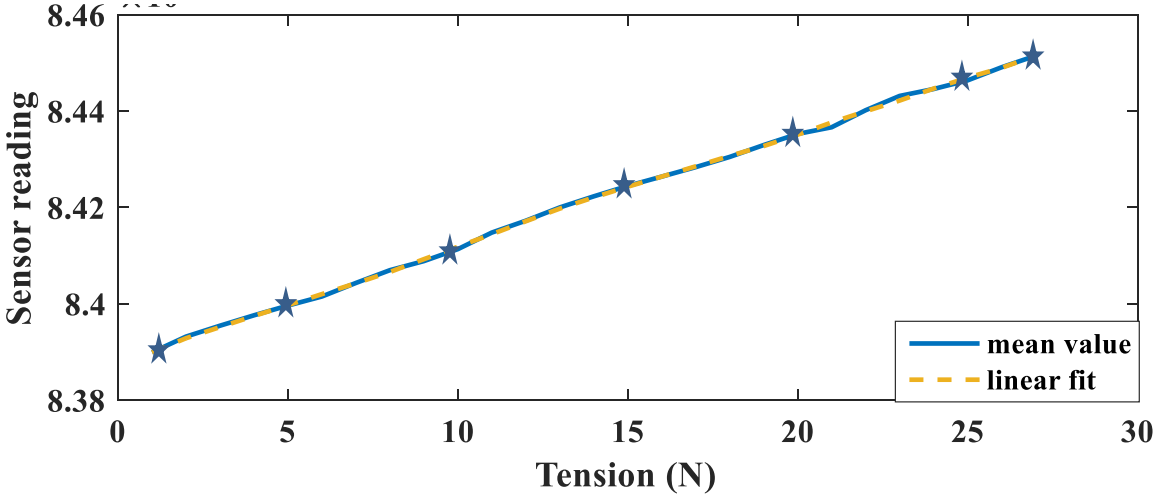


Figure D-1 Sensor reading plots against reference tension.